Performance-Based Classifier Combination in Atlas-Based Image Segmentation Using Expectation-Maximization Parameter Estimation

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

***Abstract*—It is well known in the pattern recognition community that the accuracy of classifications obtained by combining decisions made by independent classifiers can be substantially higher than the accuracy of the individual classifiers. We have previously shown this to be true for atlas-based segmentation of biomedical images. The conventional method for combining individual classifiers weights each classifier equally (vote or sum rule fusion). In this paper, we propose two methods that estimate the performances of the individual classifiers and combine the individual classifiers by weighting them according to their estimated performance. The two methods are multiclass extensions of an expectation-maximization (EM) algorithm for ground truth estimation of binary classification based on decisions of multiple experts (Warfield *et al.*, 2004). The first method performs parameter estimation independently for each class with a subsequent integration step. The second method considers all classes simultaneously. We demonstrate the efficacy of these performance-based fusion methods by applying them to atlas-based segmentations of three-dimensional confocal microscopy images of bee brains. In atlas-based image segmentation, multiple classifiers arise naturally by applying different registration methods to the same atlas, or the same registration method to different atlases, or both. We perform a validation study designed to quantify the success of classifier combination methods in atlas-based segmentation. By applying random deformations, a given ground truth atlas is transformed into multiple segmentations that could result from imperfect registrations of an image to multiple atlas images. In a second evaluation study, multiple actual atlas-based segmentations are combined and their accuracies computed by comparing them to a manual segmentation. We demonstrate in both evaluation studies that segmentations produced by combining multiple individual registration-based segmentations are more accurate for the two classifier fusion methods we propose, which weight the individual classifiers according to their EM-based performance estimates, than for simple sum rule fusion, which weights each classifier equally.**

Manuscript received January 5, 2004; revised April 26, 2004. The work of T. Rohlfing was supported in part by the National Science Foundation (NSF) under Grant EIA-0 104 114. This work is an extended version of a paper presented at the 18th International Conference on Information Processing in Medical Imaging (IPMI 2003), Ambleside, U.K., July 20–25, 2003 [31]. The Guest Editors responsible for coordinating the review of this paper and recommending its publication were C. J. Taylor and J. A. Noble. *Asterisk indicates corresponding author.*

*T. Rohlfing is with the Image Guidance Laboratories, Department of Neurosurgery, Stanford University, 300 Pasteur Drive, Room S-012, MC 5327, Stanford, CA 94305-5327 USA (e-mail: rohlfing@stanford.edu).

D. B. Russakoff is with the Computer Science Department, Stanford University, Stanford, CA 94305-9025 USA and with the Image Guidance Laboratories, Department of Neurosurgery, Stanford University, Stanford, CA 94305-5327 USA (e-mail: dbrussak@stanford.edu).

C. R. Maurer, Jr. is with the Image Guidance Laboratories, Department of Neurosurgery, Stanford University, Stanford, CA 94305-5327 USA (e-mail: calvin.maurer@igl.stanford.edu).

Digital Object Identifier 10.1109/TMI.2004.830803

***Index Terms*—Atlas-based segmentation, classifier performance, expectation-maximization (EM) parameter estimation, mixture of experts, multiclassifier decision fusion.**

I. Introduction

ONE WAY to automatically segment an image is to perform a nonrigid registration of the image to a labeled atlas image; the labels associated with the atlas image are mapped to the image being segmented using the resulting nonrigid transformation [1]–[8]. This approach has two important components that determine the quality of the segmentations, namely the registration method and the atlas. Just as human experts typically differ slightly in their labeling decisions, different registration methods produce different segmentations when applied to the same raw image and the same atlas. Likewise, different segmentations typically result from using different atlases. Therefore, each combination of a registration algorithm with an atlas effectively represents a unique classifier for the voxels in the raw image [9].

The atlas can be an image of an individual or an average image of multiple individuals. Our group recently showed that the choice of the atlas image has a substantial influence on the quality of a registration-based segmentation [10], [11]. Moreover, we demonstrated that by using multiple atlases, the segmentation accuracy can be improved over that obtained using a single atlas (either an image of an individual or an average image of multiple individuals). Specifically we showed that a segmentation produced by combining multiple individual segmentations is more accurate than the individual segmentations.[1] This finding is consistent with the observation that a combination of classifiers is generally more accurate than an individual classifier in many pattern recognition applications [12]–[17].

Typically among the individual segmentations there are more accurate ones as well as less accurate ones. This is true for human experts, due to different levels of experience, as well as for automatic classifiers, due, for example, to differences in similarities between the image to be segmented and different atlases. The conventional method for combining individual classifiers weights each classifier equally (vote or sum rule fusion [12]). More sophisticated techniques quantify classifier performance on a set of preclassified training samples used during the supervised classifier learning phase. The classifiers are then either weighted in the combination according to their performance

[1] Each individual registration was produced by nonrigid registration of an image to a different atlas that is a labeled image of a reference individual. The combination was performed by simple label averaging (sum rule fusion).

on the training set [18], [19], or the output of the most accurate classifier for a sample is selected [20]. A hierarchical multiclassifier model was described by Jordan and Jacobs [21]. Here, the outputs of elementary classifiers are propagated through a network of additional "gating networks" that assign weights to the classifier outputs based on their estimated reliabilities. The reliabilities are adjusted by an expectation-maximization (EM) algorithm [22], [23] to maximize the posterior probability of a training set.

In this paper, we propose to estimate the performances of the individual classifiers without a training set, thus eliminating the requirement of a supervised training stage. The individual classifiers are combined by weighting them according to their estimated performance. For binary (i.e., object versus background) image segmentation, Warfield *et al.* [24], [25] recently introduced an EM algorithm that derives estimates of segmentation quality parameters (sensitivity and specificity) from segmentations of the same image performed by several experts. Their method also enables the generation of an estimate of the unknown ground truth segmentation. This ground truth estimate provides a way of defining a combined segmentation that takes into account all experts, weighted by their individual accuracies.

We develop two generalizations of Warfield's algorithm to segmentations with multiple labels. The first method performs parameter estimation independently for each class with a subsequent integration step. The second method considers all classes simultaneously; by modeling explicitly the interactions between the classes in this multiclass method, the classification properties of each classifier are described comprehensively. Using the two estimation methods as classifier combination techniques, we estimate the performance parameters of multiple atlas-based segmentations and compute a maximum likelihood estimate of the correct segmentation. We apply our methods to atlas-based segmentations of confocal microscopy images of bee brains generated by registering each unsegmented image to multiple atlases, each derived from a different subject.

After introducing the fundamental concepts of atlas-based classifiers and classifier fusion in Section II, the two generalizations of the Warfield method to segmentations with arbitrary numbers of labels are described in Section III. Next, two atlas-based segmentation evaluation studies are described, which quantitatively compare different methods of combining multiple segmentations into one. First, in Section IV-B a numerical simulation study with known ground truth and random errors of known magnitudes is specifically designed to model situations where the segmentations are generated by nonrigid registration of an image to atlas images. Second, in Section IV-C we evaluate the combinations of multiple actual atlas-based segmentations.

II. Concepts and Notation

A. Atlas-Based Classifiers

Consider a segmentation that distinguishes L different classes, or labels, in a label set $\Lambda = \{1, \ldots, L\}$. A three-dimensional (3-D) atlas image $\mathcal{A}$ is a mapping from coordinates to labels $\mathcal{A} : \mathbb{R}^3 \rightarrow \Lambda$. An atlas-based classifier for a target image $\mathcal{I}$ is defined by an atlas image $\mathcal{A}$ and a coordinate transformation $\mathbf{T} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that maps the target coordinates to the atlas coordinates. From a machine learning perspective, the process of registering the atlas to the target image can be considered as training the classifier [9].

The "trained" classifier takes as its input a pixel coordinate $\mathbf{x}$ and retrieves its classification decision by looking up the label at the corresponding location in the atlas, i.e.,

$$e(\mathbf{x}) = \mathcal{A}(\mathbf{T}(\mathbf{x})). \tag{1}$$

For the present paper, we take a slightly different approach to the segmentation problem, reflected in a modified notation. For the purpose of the algorithms presented below, the spatial arrangement of voxels in a 3-D image is not relevant. In a more general way we, therefore, consider classifications of samples x that are essentially scalar, for example indexes of an event vector. When the kth classifier assigns sample x to class i we, therefore, write

$$e_k(x) = i. \tag{2}$$

The set of all samples that truly belong to class i is denoted by C_i, so the *a priori* ground truth "x is truly in class i" is written as

$$x \in C_i. \tag{3}$$

B. Classifier Performance Models

The performance of a classifier can be described by the probabilistic dependencies between its decisions and actual class memberships, written as conditional probabilities such as

$$P(e_k(x) = j \,|\, x \in C_i) \tag{4}$$

to express, for example, the probability that classifier k assigns a sample x to class j, when in fact x is in class i.

For a given test set of samples with known classifications, the classification behavior of classifier k can be expressed by its *confusion matrix* $\mathbf{N}_k$ (see [12]). The number of rows of this matrix is equal to L, the number of classes. The number of columns of the confusion matrix is equal to $L+1$. Each row corresponds to one class that a sample can be in. Each column represents a classifier decision, with an additional column for "rejected" classifications. In atlas-based segmentation, rejected classifications can, for example, be generated for spatial coordinates outside the domain of the atlas image, although it usually makes more sense to classify such locations as "background," if the atlas image is known to fully cover the object of interest.

The entries of $\mathbf{N}_k$ are the co-occurrences of classifier decisions and actual class memberships

$$n_{k,i,j} = \#\{x \,|\, x \in C_i, e_k(x) = j\}. \tag{5}$$

In other words, each of the entries $n_{k,i,j}$ of $\mathbf{N}_k$ is the number of samples x from class i which classifier k assigned to class j. Given these values, one can easily compute the conditional probabilities that describe the classification behavior of a classifier. From its confusion matrix, the probability that a sample x from class i is classified by classifier k as belonging to class j is computed as

$$\begin{aligned} P(e_k(x) = j \,|\, x \in C_i, \mathbf{N}_k) &= \frac{\#\{x \,|\, e_k(x) = j \wedge x \in C_i\}}{\#\{x \,|\, x \in C_i\}} \\ &= \frac{n_{k,i,j}}{n_{k,i,*}} \end{aligned} \tag{6}$$

where the denominator is the row sum of the ith row of $\mathbf{N}_k$, i.e.,

$$n_{k,i,*} = \sum_j n_{k,i,j}. \tag{7}$$

C. Multiclassifier Decision Fusion

The combined classifier output $E(x)$ for a sample x should be the class that maximizes the probability, given all classifier decisions $e_1(x)$ through $e_K(x)$, where K is the number of individual classifiers, and some arbitrary classifier performance model $\mathbf{P}$

$$E(x) = \arg\max_i P(x \in C_i \,|\, e_1(x), \ldots, e_K(x), \mathbf{P}). \tag{8}$$

We will take a closer look at two possible performance models in the next section. In the simplest case, there is no prior knowledge of the classifiers' performances, and all classifiers are, therefore, considered equally accurate for all classes. Xu *et al.* [12] derive several combination rules for this case, the simplest of which is "Vote Rule" decision fusion with the combination rule

$$E_{\text{vote}}(x) = \arg\max_i \sum_k \begin{cases} 1, & \text{if } i = e_k(x) \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

We note that one can apply a more general classifier model that assigns to each label a confidence value between zero and one, which expresses the strength of belief of the classifier that a given sample is from a particular class. In atlas-based segmentation in particular, these confidence values can be computed in a straight-forward way by partial volume interpolation from the atlas [9], [26], whereas binary confidences would result from using nearest neighbor interpolation within the atlas. With nonbinary confidences, we can combine classifier decisions using the so-called sum rule, which selects the class with the largest total confidence over all classifiers as their combined output. This is generally considered to be more accurate than the vote rule [27]. It is also much less likely to fail due to equal numbers of classifier votes.

Suppose now that there is some classifier performance model that quantifies how accurately each classifier recognizes samples from each of the L classes. In particular, let us consider a performance model that is specified by means of conditional classification probabilities like (6). Then Bayes' rule

$$p(A_i \,|\, B) = \frac{p(B \,|\, A_i) p(A_i)}{\sum_i p(B \,|\, A_i) p(A_i)} \tag{10}$$

yields the following class probabilities

$$\begin{aligned} &P(x \in C_i \,|\, e_1(x), \ldots, e_K(x), \mathbf{P}) \\ &= \frac{P(x \in C_i \,|\, \mathbf{P}) P(e_1(x), \ldots, e_K(x) \,|\, x \in C_i, \mathbf{P})}{\sum_{i'} P(x \in C_{i'} \,|\, \mathbf{P}) P(e_1(x), \ldots, e_K(x) \,|\, x \in C_{i'}, \mathbf{P})}. \end{aligned} \tag{11}$$

Assuming independence of the individual classifiers, this expands to

$$\begin{aligned} &P(x \in C_i \,|\, e_1(x), \ldots, e_K(x), \mathbf{P}) \\ &\quad = \frac{P(x \in C_i \,|\, \mathbf{P}) \prod_k P(e_k(x) \,|\, x \in C_i, \mathbf{P})}{\sum_j P(x \in C_j \,|\, \mathbf{P}) \prod_k P(e_k(x) \,|\, x \in C_j, \mathbf{P})} \end{aligned} \tag{12}$$

which can be incorporated into a combined classification according to (8).

III. Performance Parameter Estimation

In order to compute the performance parameters of a classifier, its outputs need to be compared to a ground truth classification of the given inputs. The ground truth is only available in a supervised training stage, which is not possible in atlas-based segmentation without first solving the entire segmentation problem.

With the ground truth unknown, the performance parameters can only be estimated by means of some approximation. This section presents two such approximation algorithms. Each is based on a different model of classifier performance. Section III-A reviews a binary performance model, whereas Section III-B introduces a multiclass performance model. From each model, an EM algorithm is derived that simultaneously estimates both the performance parameters and an approximation to the unknown ground truth classifications. This ground truth estimate also serves as the combined classifier output given the estimated performance parameters.

A. Binary Performance Model

We review below an algorithm that models classifier performance in binary segmentation, recently presented by Warfield *et al.* [24], [25]. From the performance model, an EM parameter estimation algorithm is derived, which Warfield called "simultaneous truth and performance level evaluation" (STAPLE). We omit here most of the derivation and refer the interested reader to either of Warfield's original papers instead.

Although originally formulated for binary segmentations, STAPLE is easily applied to multiclass problems by performing the parameter estimation independently for each class. Note that although independent, the parameter estimation for all classes can be executed in parallel. This means that the EM algorithm does not need to be repeated L times, where L is the number of classes, thus reducing computation time.

In the binary model, the performance of each classifier k is described by two coefficients, sensitivity p and specificity q. In binary segmentation, these are, respectively, the true positive and true negative fractions of the classifier decisions. Accordingly, the coefficients for each class i can be defined as the following conditional probabilities:

$$p_{k,i} = P(e_k(x) = i \,|\, x \in C_i) \tag{13}$$

$$q_{k,i} = P(e_k(x) \neq i \,|\, x \notin C_i). \tag{14}$$

1) Expectation Step: Using an estimate of the above performance parameters, the classifier decisions e_k for one sample x can be combined, and weights W for all classes i that represent the posterior probability of x belonging to the respective class can be computed. These weights are defined as

$$\begin{aligned} W_i(x) &\equiv P(x \in C_i \,|\, \mathbf{e}, \mathbf{p}, \mathbf{q}) \\ &= \frac{P(x \in C_i \,|\, \mathbf{p}, \mathbf{q}) \alpha_i}{P(x \in C_i \,|\, \mathbf{p}, \mathbf{q}) \alpha_i + P(x \notin C_i \,|\, \mathbf{p}, \mathbf{q}) \beta_i} \end{aligned} \tag{15}$$

where

$$\begin{aligned} \alpha_i &= P(e_1(x) = i, \ldots, e_K(x) = i \,|\, x \in C_i, \mathbf{p}, \mathbf{q}) \\ &= \left(\prod_{k: e_k(x) = i} p_{k,i} \right) \left(\prod_{k: e_k(x) \neq i} (1 - p_{k,i}) \right) \end{aligned} \tag{16}$$

and

$$\begin{aligned}\beta_i &= P(e_1(x) \neq i, \ldots, e_K(x) \neq i \,|\, x \notin C_i, \mathbf{p}, \mathbf{q}) \\ &= \left(\prod_{k:e_k(x)\neq i} q_{k,i} \right) \left(\prod_{k:e_k(x)=i} (1 - q_{k,i}) \right). \end{aligned} \tag{17}$$

The above definition of both α_i and β_i assumes independence of the individual classifiers.

Equation (15) constitutes the expectation step in the original STAPLE algorithm. In our implementation, we estimate the *a priori* class probabilities $P(x \in C_i \,|\, \mathbf{p}, \mathbf{q})$, which are independent of the performance parameters, from the classifier decisions as

$$\begin{aligned} P(x \in C_i \,|\, \mathbf{p}, \mathbf{q}) &= P(x \in C_i) \\ &\approx \frac{\sum_k \#\{x \,|\, e_k(x) = i\}}{\sum_j \sum_k \#\{x \,|\, e_k(x) = j\}}. \end{aligned} \tag{18}$$

In future work, this definition may be replaced by a map of spatially varying class probabilities, which could be derived from a large number of subjects in the form of a probabilistic atlas [28]. For now, we have found this approximation to work very well and effectively compensate for large differences in the relative frequencies of classes in a segmentation.

2) Maximization Step: From the previously calculated weights $W_i(x)$, the maximization step of the EM algorithm computes new estimates of p and q for each classifier k and each class i as follows:

$$p_{k,i}^{(t+1)} = \frac{\sum_{x:e_k(x)=i} W_i(x)}{\sum_x W_i(x)} \tag{19}$$

and

$$q_{k,i}^{(t+1)} = \frac{\sum_{x:e_k(x)\neq i} (1 - W_i(x))}{\sum_x (1 - W_i(x))}. \tag{20}$$

B. Multiclass Performance Model

Rather than performing the binary STAPLE algorithm for each class, it is possible to treat all classes simultaneously and quantify precisely the misclassification behavior of each classifier. For that, we employ a Bayesian classifier model [12], where the decisions of classifier k are described by its confusion matrix $\mathbf{N}_k$. For ease of notation and in order to achieve independence from the actual number of samples, we define row-normalized coefficients

$$\lambda_{k,i,j} = \frac{n_{k,i,j}}{n_{k,i,*}} = P(e_k(x) = j \,|\, x \in C_i, \mathbf{N}_k) \tag{21}$$

to directly represent the conditional probabilities in (6). With these, the expectation step of our algorithm is straight forward.

1) Expectation Step: As with the binary performance model, we compute weights $W_i(x)$ that represent the posterior probability of sample x belonging to class i. By inserting the definition of the performance parameters from (21) into (12), these weights can be computed as

$$\begin{aligned} W_i(x) &\equiv P(x \in C_i \,|\, \mathbf{e}, \mathbf{N}) \\ &= \frac{P(x \in C_i \,|\, \mathbf{N}) \prod_k \lambda_{k,i,e_k(x)}}{\sum_j P(x \in C_j \,|\, \mathbf{N}) \prod_k \lambda_{k,j,e_k(x)}}. \end{aligned} \tag{22}$$

Again, the classifiers are assumed to be mutually independent, and the *a priori* probabilities of the classes $P(x \in C_i \,|\, \mathbf{N})$ are estimated from the classifier decisions [see (18)].

2) Maximization Step: We write the maximization step of our algorithm in terms of the row-normalized coefficients λ rather than the entries $n_{k,i,j}$ of the confusion matrices. Recall that the expectation step (22) is easily written using these coefficients, so that their new values are all that is needed for the continuation of the algorithm. Based on the weights W computed in the previous estimation step, the updated coefficients $\lambda^{(t+1)}$ are determined as

$$\lambda_{k,i,j}^{(t+1)} = \frac{\sum_{x:e_k(x)=j} W_i(x)}{\sum_x W_i(x)}. \tag{23}$$

The derivation of this update rule is shown in the Appendix. Note the similarity of this definition with the definition of the sensitivity in the binary algorithm in (19). In the multiclass algorithm, the conditional probability $P(e_k(x) \,|\, x \in C_i, \mathbf{N})$ represented by $\lambda_{k,i,i}$ is in fact the sensitivity of classifier k for label i, i.e., $p_{k,i} \equiv \lambda_{k,i,i}$. The specificities $q_{k,i}$ are spread over the off-diagonal elements of $\mathbf{N}_k$ and can be expressed as

$$q_{k,i} = P(e_k(x) \neq i \,|\, x \notin C_i) \equiv 1 - \frac{\sum_{i' \neq i} n_{k,i',i}}{\sum_{i' \neq i} \sum_j n_{k,i',j}}. \tag{24}$$

Note that the $q_{k,i}$ can *not* be expressed in terms of the normalized coefficients λ, since normalization within a matrix row in (21) removes all information across rows. The sum over all i' in the denominator of the fraction in (24) can, therefore, not be computed using only the row-normalized $\lambda_{k,i,j}$.

C. Memory and Time-Efficient Implementation

1) One-Step Computation: As mentioned by Moon [23], the E-step and the M-step can be combined into a single, one-step update rule in order to eliminate intermediate storage. In our case, for example using the binary performance model, the update rules (19) for $p_{k,i}$ and (20) for $q_{k,i}$ depend only on the sums of the weights $W_i(x)$ either over all samples x, or over partitions thereof that are separated from each other by the classifier decisions. We can, therefore, avoid storing all weights, which would result in $\mathcal{O}(LN)$ memory. Instead, the sums required to evaluate (19) and (20) can be computed on the fly, with a single iteration over all x. For each x, we compute the weights $W_i(x)$ for all i and add them incrementally, based on the classifier decisions. By collapsing the expectation step and the maximization step into one simultaneous operation, the algorithm, therefore, has an implementation that is memory efficient as it estimates the classifier parameters in $\mathcal{O}(K) + \mathcal{O}(L)$ memory. There is virtually no penalty in computational performance. In fact, this procedure actually saves some processing time since it only requires one iteration over the classifier decisions for all samples x, rather than two iterations (one to create and one to combine) over the weights for all samples. The exact same strategy as described above for the binary model estimation can be applied completely analogously to the multiclass parameter estimation.

2) Grouping and Undisputed Samples: The second important observation is that for samples with identical classifier decisions the same weights W are computed. That is, if for samples x and x' we have $e_k(x) = e_k(x')$ for all k, then $W_i(x) = W_i(x')$, as can easily be seen from the conditional probability in (15) when $\mathbf{e}$ is considered constant. Therefore, in principle, we can compute the sums of the weights over all samples by summing over all possible classifier decisions, appropriately weighted with the frequencies of their combinations

$$\sum_x W_i(x) = \sum_{j_1} \cdots \sum_{j_K} S(j_1, \ldots, j_K) \tag{25}$$

where

$$S(j_1, \ldots, j_K) = n(j_1, \ldots, j_K) P(x \in C_i \,|\, e_k(x) = j_k, k = 1, \ldots, K) \tag{26}$$

are the appropriately weighted conditional probabilities and

$$n(j_1, \ldots, j_K) = \#\{x \,|\, e_k(x) = j_k, k = 1, \ldots, K)\} \tag{27}$$

are the multiplicities of the respective combinations of classifier decisions among all samples. The conditional probability in the former equation is again computed according to (15) in the binary model, and according to (22) in the multiclass model.

The partial sum in the numerator of (19) can be computed analogously by restricting summations to all combinations for which $e_k(x) = i$ for a given k and i

$$\sum_{x:e_k(x)=i} W_i(x) = \sum_{j_1} \cdots \sum_{j_{k-1}} \sum_{j_{k+1}} \cdots \sum_{j_K} S(j_1, \ldots, j_{k-1}, i, j_{k+1}, \ldots, j_K). \tag{28}$$

The sums of $1 - W_i(x)$, which are needed for computing the updated specificity coefficients $q_{k,i}^{(t+1)}$, can be evaluated in the same manner.

If the number of addends on the right-hand side of (25) and (28) is substantially smaller than the number of samples, then its evaluation is potentially computationally faster than that of the left-hand side of both equations. Unfortunately, the number of addends in the right-hand side expressions is exponential in the number of classifiers, i.e., its asymptotic computation time is $\mathcal{O}(L^K)$. Obviously, unless L is small, there is no gain in trading an expression that can be computed in linear time $\mathcal{O}(N)$ for one that requires exponential time, regardless of how large N may be.

Instead of the method outlined above, we pursue a mixed approach. Assuming that the individual classifiers are reasonably accurate, one can expect that they provide identical classifications for a substantial fraction of all samples. We, therefore, separate the sum over all samples into a part summing over all those samples that the classifiers disagree on, and a second part that sums the respective unanimous decisions over all classes

$$\sum_x W_i(x) = \sum_{x \in D} W_i(x) + \sum_j S(j, \ldots, j) \tag{29}$$

where

$$D = \{x \,|\, \exists k, k' : e_k(x) \neq e_{k'}(x)\} \tag{30}$$

is the set of *disputed samples*, i.e., the set of samples for which at least one classifier disagrees with the others. Again, sums over subsets of all samples and summations of $1 - W_i(x)$ are analogous. As a result, only the disputed samples need to be considered individually, while all undisputed samples are covered by L addends, one per class unanimously assigned by all classifiers.

The second term on the right-hand side of (29) can now be computed efficiently in time $\mathcal{O}(L)$. The first term strictly still requires time $\mathcal{O}(N)$, but with a substantially smaller constant than a sum over all samples. In practice, we have experienced ratios $|D|/N$ of about ten percent, resulting in a speedup factor of 10. Since L is usually small, and because the undisputed samples can be precomputed, this speedup through split-sum computation comes with virtually no penalty.

IV. Evaluation Study

In order to quantify the accuracy of combined as well as individual classifiers, their outputs need to be compared to the actual class memberships, the ground truth classification. Warfield's STAPLE algorithm [24] was originally intended to simultaneously estimate both the unknown ground truth segmentation and performance level parameters of the segmentation methods or human experts. Warfield validated this assessment by comparison to digital phantoms and demonstrated its application to assessing rater and algorithm performance in some clinical applications of segmentation. Warfield proposed the estimated ground truth segmentation was an appropriate reference standard that could be used for selecting between segmentation algorithms, or for fine tuning them. In this paper, we have evaluated generalizations of Warfield's method for the task of deriving an improved segmentation estimate from a collection of atlas-based segmentations. In this section, we describe validation experiments comparing fusion strategies for deriving the optimal segmentation, using both synthetic specified ground truth and manual segmentations.

A. Image Data

We evaluate the methods described in this paper by segmenting 3-D confocal microscopy images of the brains of 20 adult foraging honeybees (see [10] for details). Each volume contained 84–114 slices with thickness 8 μm, and each slice had 610–749 pixels in x direction and 379–496 pixels in y direction with pixel size 3.8 μm. In each individual image, 22 anatomical structures were distinguished and manually labeled by a human expert. The manual segmentation for each image can serve both as the ground truth to quantify the accuracy of an automatic segmentation, and as an atlas for atlas-based segmentation of another image. An example slice from a microscopy image and the corresponding label image are shown in Fig. 1. For a detailed description of the imaging process and a complete list of the anatomical structures, the interested reader is referred to [10] and [11].

B. Numerical Simulation Study

1) Independent Random Classifiers With Known Ground Truth: Imperfect segmentations with known error magnitudes are simulated by applying random deformations to a given atlas. Each randomly deformed atlas serves as a model of an imperfect segmentation of the image that the original, undeformed atlas was derived from. Several of these deformed atlases are combined into one segmentation using the methods described

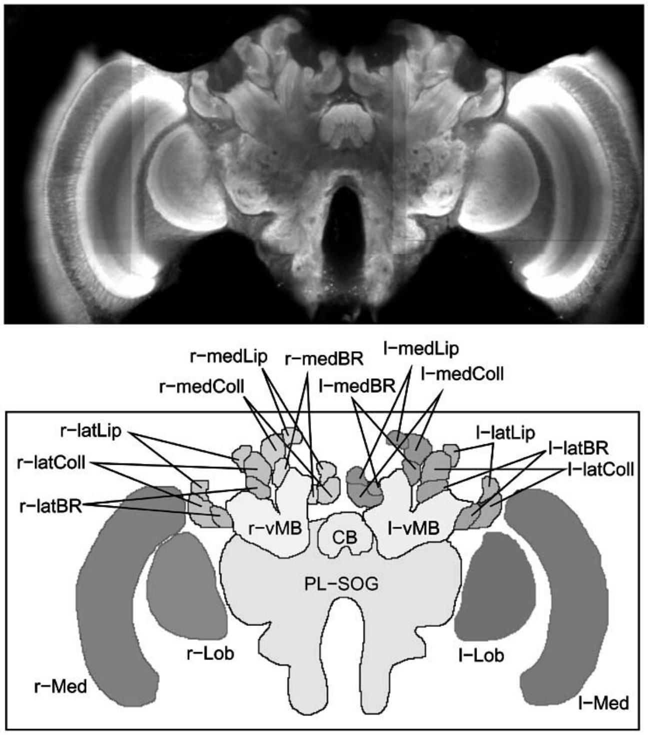


Fig. 1. Example of bee brain confocal microscopy (*top*) and corresponding label image as defined by manual segmentation (*bottom*). Every gray level in the label image represents a different anatomical structure. Due to limitations of reproduction, different gray levels may look alike. From Rohlfing *et al.* [10].

in the previous sections. Since the original (undeformed) atlas is known, it provides the ground truth for evaluating the results of the classifier combination methods.

An increasingly popular nonrigid registration method originally introduced by Rueckert *et al.* [29] applies free-form deformations [30] based on B-spline interpolation between uniform control points. We implemented this transformation model and generate random transformations by adding normally distributed random numbers to the control point coordinates. The standard deviation of the normal distribution controls the magnitude of the random deformation, and consequently the error rate of the simulated segmentation.

2) Evaluation Study Design: For each ground truth (i.e., atlas), random B-spline-based free-form deformations were generated by adding independent Gaussian-distributed random numbers to the coordinates of all control points. The control point spacing was 120 μm, corresponding to approximately 30 voxels in x and y direction and 15 voxels in z direction. The standard deviations of the Gaussian distributions were $\sigma = 10$, 20, and 30 μm, corresponding to approximately 2, 4, and 8 voxels in x and y direction (1, 2, and 4 voxels in z direction). Fig. 2 shows examples of an atlas after application of several random deformations of different magnitudes. A total of 20 random deformations were generated for each individual and each σ. The randomly deformed atlases were combined into a final atlas once by label averaging (sum rule fusion), and once using each of our novel algorithms.

3) Results: First, we investigated the combined classification accuracy of individual segmentations with identical error levels. Fig. 3 shows a plot of the mean recognition rates over all 20 individuals versus the number of segmentations. Both EM algorithms performed consistently better, i.e., produced more accurate combined segmentations, than simple label averaging. The improvement achieved using the EM strategies was larger for greater magnitudes of the random atlas deformations. Between the two EM methods, repeated application of the binary algorithm outperformed the multiclass method. For all algorithms, adding additional segmentations increased the accuracy of the combined segmentation. The incremental improvement obtained by adding an additional segmentation decreased as the number of atlases increased. The figure also illustrates the superiority of using multiple atlases over using just one: in all cases, the individual recognition rates are substantially lower than any of the combined results. Again, the difference increases as the magnitude of the random deformations is increased.

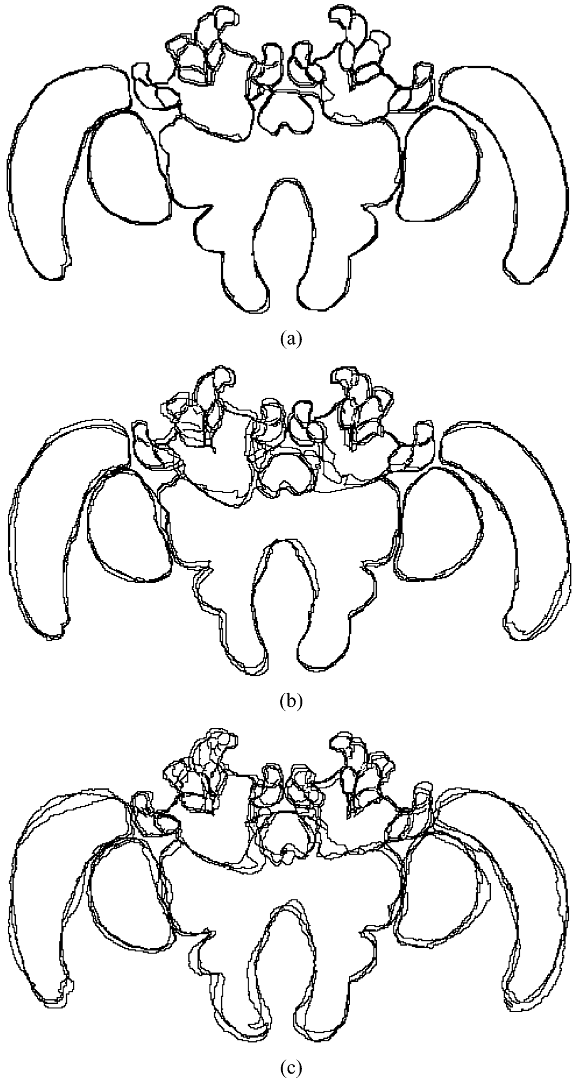


Fig. 2. Examples of a randomly deformed atlas. Each image shows overlays of the contours from the same atlas deformed by three random transformations of equal magnitudes (a) $\sigma = 10$ μm, (b) $\sigma = 20$ μm, and (c) $\sigma = 30$ μm.

Next, we considered the performance of the classifier combination methods when the input segmentations have different error levels. For each of the three deformation magnitudes (10,

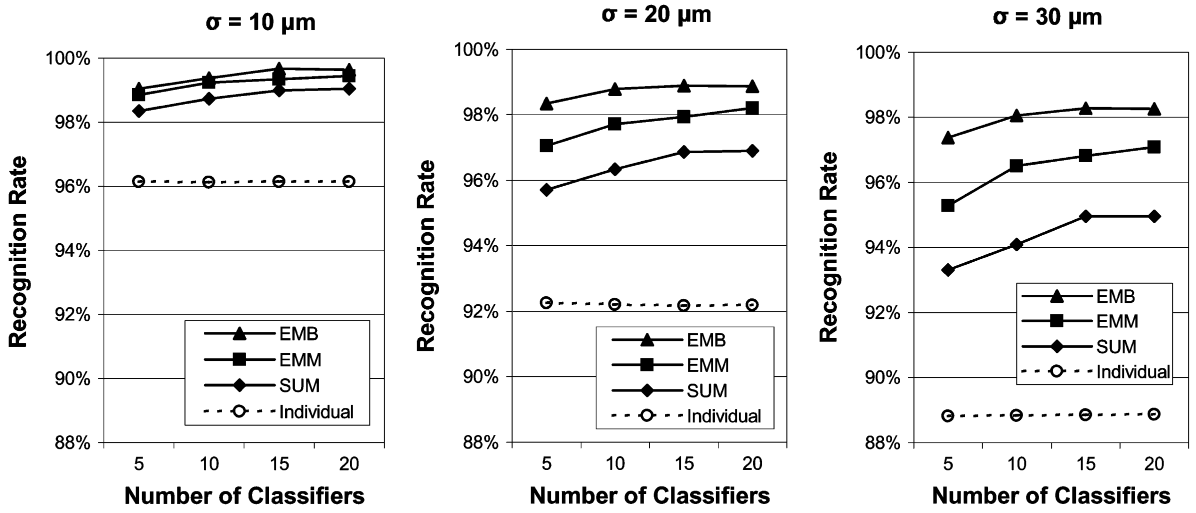


Fig. 3. Mean recognition rates of combined segmentation over 20 individuals versus number of random segmentations used. Results are shown for sum rule fusion (SUM), the binary-model EM algorithm (EMB), and the multiclass-model EM algorithm (EMM). Each method was applied to classifiers based on atlases after random deformations with magnitudes $\sigma = 10$ μm (left), $\sigma = 20$ μm (center), and $\sigma = 30$ μm (right). The dashed line in each graph shows the average individual recognition rates achieved by the respective set of classifiers. Modified from Rohlfing *et al.* [31].

20, and 30 μm), we selected the first $n = 3, 5,$ and 7 random segmentations, resulting in a total number of 9, 15, and 21 segmentations, respectively. The resulting recognition rates are shown in Fig. 4. Again, the recognition rates of all three combination methods increased as more segmentations were added. However, unlike in the presence of identical error levels, the multiclass EM algorithm outperformed the binary-model algorithm. Both EM methods again outperformed the sum rule combination method.

C. Validation Against a Manual Gold Standard

It is clear from the previous section that the properties of the individual segmentations have a substantial influence on the performance of the classifier combination methods. Most notably, the relative performance of the two EM methods with respect to each other varies, for example, depending on the independence of the individual segmentations. It is, therefore, clear that an evaluation with actual segmentations is needed to assess how the results of the numerical simulation translate to a real application scenario. This section describes and analyzes such an evaluation of the classifier combination techniques applied to actual atlas-based segmentations. Here, the ground truth for the segmentation quality assessment is provided by manual segmentations.

1) Evaluation Study Design: Using the aforementioned microscopy images from 20 bee brains, a leave-one-out study is performed. Each of the microscopy images is separately selected as the image to be segmented. The manual segmentation of this image serves as a ground truth for the automatic techniques. Using the manual segmentations of the remaining 19 images as atlases, 19 atlas-based segmentations are computed.

The registration transformation between image and atlas is determined using the nonrigid algorithm introduced by Rueckert *et al.* [29]. A parallel implementation of this technique [32] is applied in order to minimize computation times.

2) Parameter Estimation Accuracy: The sensitivity performance parameters estimated by the EM methods are plotted against the actual *a posteriori* recognition rates in Fig. 5 for both the binary [Fig. 5(a)] and the multiclass [Fig. 5(b)] performance models. Each plot shows the sensitivity parameters computed for all 22 classes (structures). In order to improve the visual presentation, only five segmented images were included in the plots. The five images were selected randomly and are identical for both plots. Thus there are 110 data points in each plot (one data point for each of 22 classes in each of five images). Both EM methods computed relatively accurate estimates of the true sensitivity parameters (linear regression, $R = 0.94$ and 0.87 for the binary and the multiclass models, respectively).

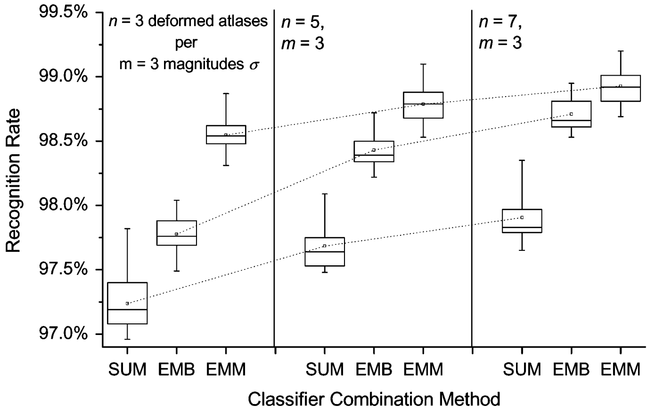
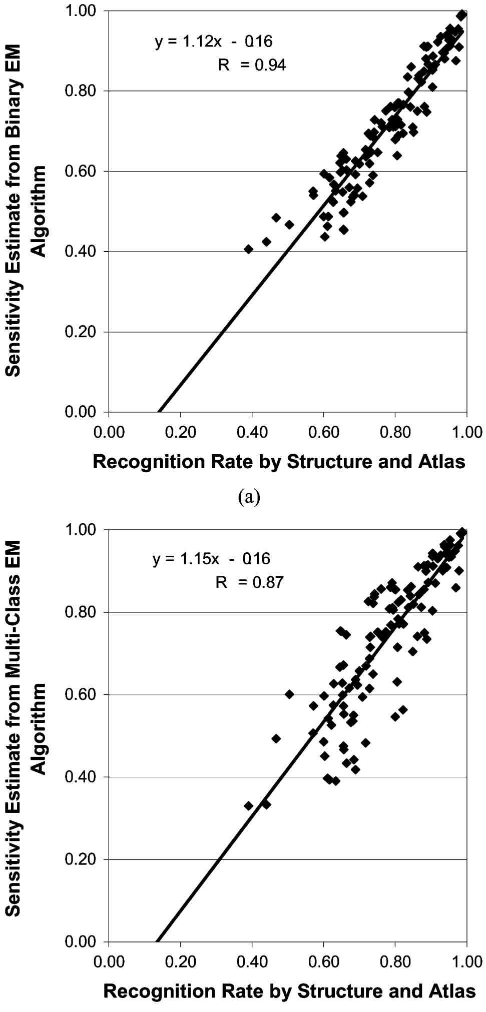


Fig. 4. Recognition rates of combined classifications based on random segmentations with different deformation magnitudes. The whiskers show the range of values. The boxes show the 25th and 75th percentiles. The horizontal bars are the median values, and the connected small squares are the mean values.

The difference in sensitivity parameter estimation accuracy may be due to the fact that the multiclass algorithm estimates a substantially larger number of parameters, $KL(L+1)$, than the binary algorithm, which estimates $2KL$ parameters. It is important to keep in mind that the sensitivity parameters represent only a fraction of the classifier parameterization, and in the case of the binary-model algorithm this parameterization is fundamentally incomplete.



(a)

(b)

Fig. 5. Estimated versus actual sensitivity performance parameters (a) Binary performance model, (b) Multiclass performance model.

The recognition rates achieved using the three decision fusion methods are compared to each other in Fig. 6. For reference, this figure also shows the recognition rates using a single individual atlas with no decision fusion. The single atlas was chosen based on the *a posteriori* recognition rates, that is, it was the one image out of the 20 subjects in our study, which, when used as the atlas, gave the best recognition rates for segmentations of the remaining 19 images [10].

3) Segmentation Accuracy: It is easy to see that each of the three decision fusion methods produced more accurate segmentations (i.e., higher recognition rates) than atlas-based segmentation using a single individual atlas. We used the best possible individual atlas, so no other subject, when chosen as the atlas, produced a higher recognition rate.

Between the decision fusion methods, the EM algorithm based on the binary performance model outperforms sum rule fusion. Both methods are outperformed by the EM algorithm based on the multiclass performance model. The mean recognition rates were: 95% for the multiclass performance model algorithm, 93% for the binary performance model algorithm, 91% for performance model-free sum rule fusion. The mean of the individual recognition rates of the classifiers was 90%.

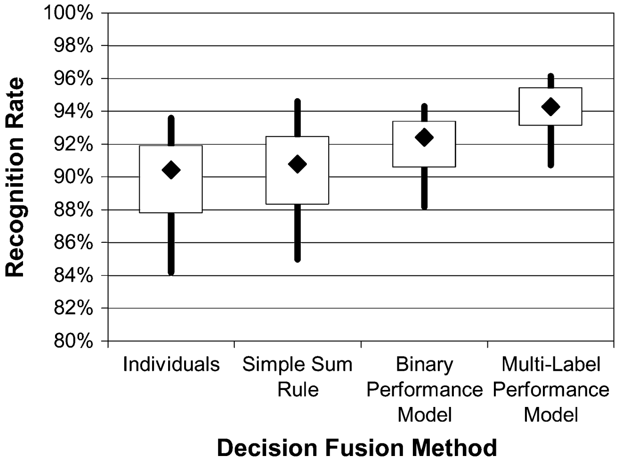


Fig. 6. Comparison of segmentation accuracy using a single atlas versus different decision fusion methods. The diamonds show the median recognition rate for each method over 20 segmented subjects. The ends of the solid lines represent the minimum and maximum recognition rates, while the lower and upper edges of the boxes represent the 25th and 75th percentiles, respectively.

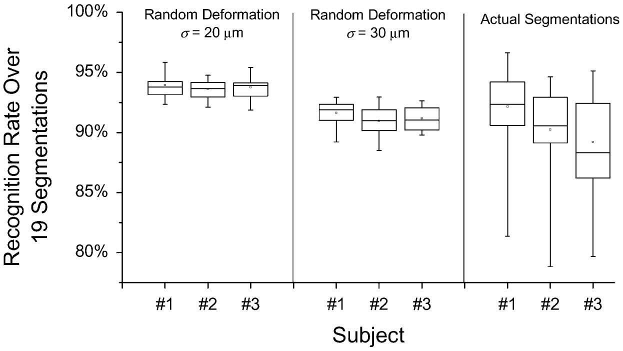


Fig. 7. Comparison of recognition rates between randomly generated and actual individual segmentations. For the same three subjects, this plot shows the distribution of recognition rates for one anatomical structure among 19 simulated segmentations with deformation magnitude $\sigma = 20$ μm (three left boxes), simulated segmentations with deformation magnitude $\sigma = 30$ μm (center boxes), and actual atlas-based segmentations (right boxes). The meanings of the box plot elements are the same as in Fig. 4.

Segmentations produced by the multiclass EM algorithm were significantly more accurate than those produced by the binary EM algorithm (two-sided paired t-test, $P < 10^{-7}$). Both EM algorithms outperformed simple sum fusion (binary, $P < 10^{-5}$; multiclass, $P < 10^{-7}$).

D. Comparison of Simulation and Actual Application

We note that the results of the first numerical simulation are somewhat different than those of the second simulation and those of the application to actual segmentations. To illustrate possible reasons for this discrepancy, Fig. 7 shows a comparison between the properties of the simulated segmentations and those of actual segmentations. It is easy to see that the individual actual segmentations cover a substantially larger range of accuracies than the simulated segmentations from one magnitude class alone. By combining multiple simulated segmentations from different classes in the second numerical simulation, this aspect of the actual segmentations was better simulated. Consequently, the relative performance of the binary and

TABLE I
MEMORY PORTIONS BY DATA TYPE REQUIRED TO COMBINE 5, 10, 15, AND 20 ATLAS-BASED CLASSIFIERS

Number of Classifiers	Classifier Decisions	Binary Parameters	Multi-Class Parameters
5	150 MB	880 bytes	9,680 bytes
10	300 MB	1,760 bytes	19,360 bytes
15	450 MB	2,640 bytes	29,040 bytes
20	600 MB	3,520 bytes	38,720 bytes

These values are based on an image of $748 \times 496 \times 84$ voxels, with 22 classes and one byte allocated per voxel in the segmentation.

the multiclass model algorithms are similar for the second numerical simulation and the actual application. This indicates that the binary model may be better able to take advantage of equal performances among the classifiers. It also illustrates, however, that this is not a realistic situation.

E. Computation Time and Memory Requirements

Computation time increases approximately linearly with the number of classifiers. Without split-sum computation by disputed and undisputed samples, the time per iteration is approximately 5–10 s per classifier (using a PC with a 3.0-GHz Intel Pentium 4 processor). When split-sum computation is used, time also increases with the magnitude of the deformations, as larger differences in the classifier decisions lead to a larger fraction of disputed voxels. For simulated segmentations, the relative speedup factor achieved by split-sum computation was between 10 (for $\sigma = 10$ μm) and 5 (for $\sigma = 30$ μm). For the actual segmentations, split-sum computation still reduces computation time by about half. Since one bad segmentation can substantially increase the number of disputed samples, this difference compared to the simulated case is not unexpected.

Between the two EM methods, the multiclass performance model requires slightly less computation time per iteration than the binary model. The binary model needs to update two parameter estimates (p and q) per sample, whereas the multiclass model requires only one matrix entry to be updated per sample. However, the convergence of the multiclass model is considerably slower than that of the binary model. For actual segmentations, the mean number of iterations was 17 for the binary model and 66 for the multiclass model. Overall, segmentation, including parameter estimation and combination but not including the nonrigid registrations, took approximately 0.3 h using the binary model and 0.9 h using the multiclass model. Nonrigid registration between an image and an atlas took about 1.5 h per image. Thus these EM algorithms are not the rate-limiting step in multiatlas segmentation. However, we note that we have a parallel implementation of our nonrigid registration algorithm [32], so the computation time required for the registration step can be substantially reduced using multiprocessor computing resources. In addition, as a result of the voxel-wise independence assumption that underlies both EM algorithms, we also have parallel implementations of these.

An overview of the working memory required to process and combine 5, 10, 15, and 20 individual segmentations is shown in Table I. Note that the memory allocation for both algorithms and all numbers of classifiers are dominated by the space required to store the original individual segmentations. If all voxel label probabilities were computed according to (15) or (22), additional memory would be required. With the image dimensions given in the above example, storing 22 real-valued probabilities per voxel would require approximately 2.6 GBytes of additional memory. Unless, for example, dependencies between neighboring voxels require them, the advantages of eliminating the explicit computation of all $W_i(x)$ by contracting the E- and M-steps into a single operation are clear.

V. DISCUSSION

We have described in this paper two methods for performance-based combination of multiple classifiers with self-supervised learning of the classifier parameters. These methods eliminate the need for a supervised training stage. This is particularly important in our application of atlas-based image segmentation, since the classifier performances depend on the image to be segmented. Training wouldneed to be repeated for every new image, which is not possible without generating the correct classifications by first segmenting the image.

Generation of multiple atlas-based classifiers requires repeated application of a (computationally expensive) nonrigid registration algorithm. One may, therefore, ask whether one could conceivably compile a single, potentially probabilistic, atlas to incorporate the variation among the different individual atlases that leads to independent classifiers. In previous work, we evaluated one possible such approach by segmenting images using an average shape atlas [33]. We found that sum fusion of multiple independent classifiers arising from multiple individual atlases significantly outperformed the average shape atlas [10]. In general, we believe that independent nonrigid registrations are vital for generating independent classifications. As the variation among the registrations to different individual atlases depends on the image to be segmented, it seems that this variation can not be achieved using a single atlas that is independent of the unsegmented image.

Better nonrigid transformations produce better atlas-based segmentations. Obviously if it is possible to generate a perfect transformation between an image and an atlas, then combining multiple perfect segmentations is not necessary. But nonrigid registration is a difficult problem, and we are not convinced that it will ever be solved. If a transformation model cannot perfectly describe all details of the true mapping between images from two subjects, which is likely the case for our B-spline model, it is possible that individual atlas-based segmentations and combined segmentations using any fusion strategy may not be able to adequately segment some target structures. But even a perfect transformation model that can describe normal and pathological variability in the application to which atlas-based segmentation may be applied will be a very high-dimensional mapping. Computing the correct transformation parameters that describe the particular transformation in each case will be quite difficult and errors in the parameters will lead to errors in segmentation. As long as there are errors in the segmentations, and as long as the errors in multiple segmentations are somewhat independent, there is a potential benefit from combining multiple segmentations.

Other methods have been proposed to learn classifier performance parameters and apply them for weighted classifier combination, see for example [18]–[20], and [34]. None of the published techniques as far as we are aware, however, works without a supervised training stage. They are, therefore, exclusively applied to classification problems where supervised training is feasible. We note that our methods, too, can be applied to such problems. An interesting question for future work in this field will, therefore, be how our algorithms compare with other performance models in more generic applications such as handwriting recognition.

Between the two methods covered in this paper, the algorithm based on the binary performance model outperformed the multiclass algorithm in one of the numerical simulations. It is, however, outperformed by the multiclass model in terms of the recognition rates achieved for simulated mixtures of classifiers with different performance levels and, more importantly, for actual atlas-based segmentations. Ultimately, it is of course the latter that we care about. While we note that the "gold standard" manual segmentation may be imperfect, thus, potentially reducing the validity of the accuracy evaluation, the additional results of the numerical simulations provide sufficient evidence that the effects observed when combining actual segmentations are real.

We conclude that the two methods described in this paper are compact and efficient algorithms for estimating classifier performance, and as a result, for improving the overall combined accuracy of a multiclassifier system. In particular, we demonstrated that these methods can improve the accuracy of atlas-based segmentation by combining multiple individual registration-based segmentations, which are weighted according to their EM-based performance estimate.

Appendix

We show in this appendix that the update rules of the multiclass parameter estimation, (22) for the E-step and (23) for the M-step, satisfy the conditions of an actual EM algorithm. We note that the expected log-likelihood function Q of a general EM algorithm with observed variables $\mathbf{x}$, hidden variables $\mathbf{y}$ and parameters $\mathbf{\Theta}$ is

$$\begin{aligned} Q\left(\mathbf{\Theta} \,|\, \mathbf{\Theta}^{(t-1)}\right) &= E_{\mathbf{y}}\left[\ln P(\mathbf{x}, \mathbf{y} \,|\, \mathbf{\Theta}) \,|\, \mathbf{x}, \mathbf{\Theta}^{(t-1)}\right] \\ &= \sum_{\mathbf{y}} P\left(\mathbf{y} \,|\, \mathbf{x}, \mathbf{\Theta}^{(t-1)}\right) \ln P(\mathbf{x}, \mathbf{y} \,|\, \mathbf{\Theta}). \end{aligned} \tag{31}$$

Here, $\mathbf{\Theta}^{(t-1)}$ is the current parameter estimate, either from the previous iteration of the algorithm or, in the first iteration, the initial guess.

In our multiclass algorithm, the parameters are the entries $n_{k,i,j}$ of the confusion matrix for each classifier, which is denoted by $\mathbf{N}_k$. For compactness of notation, we write $\mathbf{N}$ for the sequence of all K confusion matrices, i.e., $\mathbf{N} = (\mathbf{N}_1, \ldots, \mathbf{N}_K)$. The observed variables are the classifier decisions $e_1(x)$ through $e_K(x)$ for all voxels x, and like before we write $\mathbf{e}(x) = (e_1(x), \ldots, e_K(x))$. The hidden variables are the correct classifications $x \in C_i$ for all voxels. The conditional probabilities of the hidden data $\mathbf{y}$ are, therefore

$$P(\mathbf{y} \,|\, \mathbf{x}, \mathbf{\Theta}) \equiv P(x \in C_i \,|\, \mathbf{e}(x), \mathbf{N}). \tag{32}$$

For $\mathbf{\Theta} = \mathbf{\Theta}^{(t-1)}$ these are the coefficients of the Q function, which have to be computed in the E-step. This is precisely what (22) does using Bayes' rule.

Given $Q(\mathbf{\Theta} \,|\, \mathbf{\Theta}^{(t-1)})$ as a function of $\mathbf{\Theta}$ for fixed $\mathbf{\Theta}^{(t-1)}$, the M-step of the EM algorithm determines the new parameters $\mathbf{\Theta}^{(t)}$ so that Q is maximized

$$\mathbf{\Theta}^{(t)} = \arg\max_{\mathbf{\Theta}} Q\left(\mathbf{\Theta} \,|\, \mathbf{\Theta}^{(t-1)}\right). \tag{33}$$

With the joint distribution of hidden and observed data

$$P(\mathbf{x}, \mathbf{y} \,|\, \mathbf{\Theta}) \equiv P(\mathbf{e}(x), x \in C_i \,|\, \mathbf{N}) \tag{34}$$

the maximization problem in our algorithm, therefore, is

$$\begin{aligned} \mathbf{N}^{(t)} &= \arg\max_{\mathbf{N}} Q\left(\mathbf{N}, \mathbf{N}^{(t-1)}\right) \\ &= \arg\max_{\mathbf{N}} \sum_x \sum_i \Big[P\left(x \in C_i \,|\, \mathbf{e}(x), \mathbf{N}^{(t-1)}\right) \\ &\qquad\qquad \times \ln P(\mathbf{e}(x), x \in C_i \,|\, \mathbf{N})\Big] \end{aligned} \tag{35}$$

where x runs over all voxels and i runs over all classes. The class probabilities given the previous parameter estimates $\mathbf{N}^{(t-1)}$ are the weights W computed in the previous E-step, i.e.,

$$P\left(x \in C_i \,|\, \mathbf{e}(x), \mathbf{N}^{(t-1)}\right) \equiv W_i(x). \tag{36}$$

Assuming conditional independence of the individual classifiers, we can write the joint probability of their decisions and the ground truth as

$$P(\mathbf{e}(x), x \in C_i \,|\, \mathbf{N}) = \prod_k P(e_k(x), x \in C_i \,|\, \mathbf{N}). \tag{37}$$

Substituting the former two equations into (35) yields

$$\begin{aligned} \mathbf{N}^{(t)} &= \arg\max_{\mathbf{N}} \sum_x \sum_i \left[W_i(x) \ln \prod_k P(e_k(x), x \in C_i \,|\, \mathbf{N})\right] \\ &= \arg\max_{\mathbf{N}} \sum_x \sum_i \sum_k [W_i(x) \ln P(e_k(x), x \in C_i \,|\, \mathbf{N})]. \end{aligned} \tag{38}$$

The optimum can be determined by exploiting the necessary condition that all partial derivatives must vanish. We, therefore, set the partial derivatives of the target function with respect to each entry $n_{k,i,j}$ in $\mathbf{N}_k$ to zero and solve for $n_{k,i,j}$.

First, we observe that all joint probability terms for classifiers other than k are eliminated from the expression by derivation

$$\begin{aligned} &\frac{\partial}{\partial n_{k,i,j}} \sum_x \sum_{i'} \sum_{k'} [W_{i'}(x) \ln P(e_{k'}(x), x \in C_{i'} \,|\, \mathbf{N})] \\ &\quad = \frac{\partial}{\partial n_{k,i,j}} \sum_x \sum_{i'} [W_{i'}(x) \ln P(e_k(x), x \in C_{i'} \,|\, \mathbf{N})]. \end{aligned} \tag{39}$$

Next, we separate the sum over all classes i' into the index corresponding to the derivation variable plus the sum over all remaining classes

$$= \frac{\partial}{\partial n_{k,i,j}} \sum_x \Bigg[[W_i(x) \ln P(e_k(x), x \in C_i \,|\, \mathbf{N})] + \sum_{i' \neq i} [W_{i'}(x) \ln P(e_k(x), x \in C_{i'} \,|\, \mathbf{N})] \Bigg]$$

$$= \sum_x \Bigg[\Big[W_i(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_i \,|\, \mathbf{N}) \Big] + \sum_{i' \neq i} \Big[W_{i'}(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_{i'} \,|\, \mathbf{N}) \Big] \Bigg]. \quad (40)$$

Likewise, we separate all samples x into those for which classifier k outputs label j, and those for which it outputs any other label

$$= \sum_{x:e_k(x)=j} \Bigg[\Big[W_i(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_i \,|\, \mathbf{N}) \Big] + \sum_{i' \neq i} \Big[W_{i'}(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_{i'} \,|\, \mathbf{N}) \Big] \Bigg]$$
$$+ \sum_{x:e_k(x) nej} \Bigg[\Big[W_i(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_i \,|\, \mathbf{N}) \Big] + \sum_{i' \neq i} \Big[W_{i'}(x) \frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x), x \in C_{i'} \,|\, \mathbf{N}) \Big] \Bigg] \quad (41)$$

The actual partial derivatives of the conditional probabilities can be derived from the defining property of the entries in the confusion matrix

$$P(e_k(x) = j \wedge x \in C_i \,|\, \mathbf{N}) = \frac{n_{k,i,j}}{\sum_{i',j'} n_{k,i',j'}} = \frac{n_{k,i,j}}{N_k} \quad (42)$$

where $N_k = \sum_{i,j} n_{k,i,j}$ is the sum over all entries of the matrix $\mathbf{N}_k$. When computing the partial derivative

$$\frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x) = j', x \in C_{i'} \,|\, \mathbf{N}) \quad (43)$$

of any such conditional probability w.r.t. a given matrix entry, we need to consider the following two cases.

1) $i = i'$ and $j = j'$: By analogy to $(\partial/\partial x)\ln((x)/(a + x)) = (a)/(x(a + x))$ with $x \equiv n_{k,i,j}$ and $a + x \equiv \sum_{i'} \sum_{j'} n_{k,i',j'}$ one finds

$$\frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x) = j, x \in C_i \,|\, \mathbf{N}) = \frac{\partial}{\partial n_{k,i,j}} \ln \frac{n_{k,i,j}}{\sum_{i'} \sum_{j'} n_{k,i',j'}} = \frac{N_k - n_{k,i,j}}{n_{k,i,j} N_k} = \frac{1}{n_{k,i,j}} - \frac{1}{N_k}. \quad (44)$$

2) $i \neq i'$ or $j \neq j'$: By analogy to $(\partial/\partial x)\ln((b)/(a+x)) = -(1)/(a + x)$ one finds

$$\frac{\partial}{\partial n_{k,i,j}} \ln P(e_k(x) = j', x \in C_{i'} \,|\, \mathbf{N}) = \frac{\partial}{\partial n_{k,i,j}} \ln \frac{n_{k,i',j'}}{\sum_{i''} \sum_{j''} n_{k,i'',j''}} = -\frac{1}{N_k}. \quad (45)$$

Substituting (44) and (45) into (41) as appropriate yields

$$\sum_{x:e_k(x)=j} \left[W_i(x) \left(\frac{1}{n_{k,i,j}} - \frac{1}{N_k} \right) - \sum_{i' \neq i} \frac{W_{i'}(x)}{N_k} \right] - \sum_{x:e_k(x) \neq j} \left[\frac{W_i(x)}{N_k} - \sum_{i' \neq i} \frac{W_{i'}(x)}{N_k} \right]. \quad (46)$$

For any sample x, the weights $W_i(x)$ over all classes i sum to unity, so $\sum_{i' \neq i} W_{i'}(x) = 1 - W_i(x)$. Inserting this into the above, we find

$$\sum_{x:e_k(x)=j} \left[W_i(x) \left(\frac{1}{n_{k,i,j}} - \frac{1}{N_k} \right) - \frac{1 - W_i(x)}{N_k} \right] - \sum_{x:e_k(x) \neq j} \left[\frac{W_i(x)}{N_k} - \frac{1 - W_i(x)}{N_k} \right]$$
$$= \sum_{x:e_k(x)=j} \left[\frac{W_i(x)}{n_{k,i,j}} - \frac{W_i(x)}{N_k} - \frac{1 - W_i(x)}{N_k} \right] - \sum_{x:e_k(x) \neq j} \left[\frac{W_i(x)}{N_k} - \frac{1 - W_i(x)}{N_k} \right]$$
$$= \sum_{x:e_k(x)=j} \left[\frac{W_i(x)}{n_{k,i,j}} - \frac{1}{N_k} \right] - \sum_{x:e_k(x) \neq j} \frac{1}{N_k} \quad (47)$$

and by regrouping and combination of two partial sums over samples x

$$= \sum_{x:e_k(x)=j} \frac{W_i(x)}{n_{k,i,j}} - \sum_x \frac{1}{N_k}. \quad (48)$$

Recall that this is the final expression of the derivative of the Q function with respect to entry i, j in the confusion matrix of classifiers k, that is $n_{k,i,j}$. We can now set this expression to zero to satisfy the necessary criterion for a local maximum, i.e.,

$$\frac{\partial}{\partial n_{k,i,j}} Q = \sum_{x:e_k(x)=j} \frac{W_i(x)}{n_{k,i,j}} - \sum_x \frac{1}{N_k} \overset{!}{=} 0. \quad (49)$$

Solving this equation for $n_{k,i,j}$ finally leads to

$$n_{k,i,j} = \frac{N_k}{\sum_x 1} \sum_{x:e_k(x)=j} W_i(x). \quad (50)$$

Strictly, this is not a complete definition of $n_{k,i,j}$, since it is also an addend in the definition of N_k and, thus, appears on both sides of the equality. However, from this relation we can compute the conditional probabilities for the subsequent expectation step without ambiguity as

$$\lambda_{k,i,j}^{(t)} = \frac{n_{k,i,j}}{\sum_{j'} n_{k,i,j'}} = \frac{\frac{N_k}{\sum_x 1} \sum_{x:e_k(x)=j} W_i(x)}{\frac{N_k}{\sum_x 1} \sum_{j'} \sum_{x:e_k(x)=j'} W_i(x)} = \frac{\sum_{x:e_k(x)=j} W_i(x)}{\sum_{j'} \sum_{x:e_k(x)=j'} W_i(x)} = \frac{\sum_{x:e_k(x)=j} W_i(x)}{\sum_x W_i(x)}. \quad (51)$$

This is obviously identical to (23), which concludes the proof that our multiclass estimation method is in fact a true EM algorithm. ■

Acknowledgment

All computations were performed on an SGI Origin 3800 supercomputer in the Stanford University Bio-X Core Facility for Biomedical Computation. Bee brain microscopy images and atlases were provided by R. Brandt (Indeed—Visual Concepts GmbH, Berlin, Germany) and R. Menzel (Institut für Neurobiologie, Freie Universität Berlin, Berlin, Germany). Manual segmentations were performed by A. Steege and C. Kaps (Freie Universität Berlin, Berlin, Germany). T. Rohlfing would like to thank J. Denzler (Universität Passau, Passau, Germany) for lengthy discussions and valuable comments and suggestions. The authors would like to thank the anonymous reviewers for their constructive comments and valuable suggestions that have substantially improved this paper.

References

[1] 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. 11 944–11 948, 1993.

[2] J. C. Gee, M. Reivich, and R. Bajcsy, "Elastically deforming a three-dimensional atlas to match anatomical brain images," *J. Comput. Assist. Tomogr.*, vol. 17, no. 2, pp. 225–236, 1993.

[3] D. L. Collins, C. J. Holmes, T. M. Peters, and A. C. Evans, "Automatic 3-D model-based neuroanatomical segmentation," *Hum. Brain Mapp.*, vol. 3, no. 3, pp. 190–208, 1995.

[4] D. V. Iosifescu, M. E. Shenton, S. K. Warfield, R. Kikinis, J. Dengler, F. A. Jolesz, and R. W. McCarley, "An automated registration algorithm for measuring MRI subcortical brain structures," *NeuroImage*, vol. 6, no. 1, pp. 13–25, 1997.

[5] B. M. Dawant, S. L. Hartmann, J. P. Thirion, F. Maes, D. Vandermeulen, and P. Demaerel, "Automatic 3-D segmentation of internal structures of the head in MR images using a combination of similarity and free-form transformations: Part I, methodology and validation on normal subjects," *IEEE Trans. Med. Imag.*, vol. 18, pp. 909–916, Oct. 1999.

[6] S. L. Hartmann, M. H. Parks, P. R. Martin, and B. M. Dawant, "Automatic 3-D segmentation of internal structures of the head in MR images using a combination of similarity and free-form transformations: Part I, validation on severely atrophied brains," *IEEE Trans. Med. Imag.*, vol. 18, pp. 917–926, Oct. 1999.

[7] C. Baillard, P. Hellier, and C. Barillot, "Segmentation of brain 3D MR images using level sets and dense registration," *Med. Image Anal.*, vol. 5, no. 3, pp. 185–194, 2001.

[8] W. R. Crum, R. I. Scahill, and N. C. Fox, "Automated hippocampal segmentation by regional fluid registration of serial MRI: Validation and application in Alzheimer's disease," *NeuroImage*, vol. 13, no. 5, pp. 847–855, 2001.

[9] T. Rohlfing and C. R. Maurer Jr., "Multi-classifier framework for atlas-based image segmentation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Pt.I*, Washington, D.C., 2004, pp. 255–260.

[10] 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.

[11] ——, "Segmentation of three-dimensional images using nonrigid registration: Methods and validation with application to confocal microscopy images of bee brains," *Proc. SPIE (Med. Imag.: Image Processing)*, vol. 5032, pp. 363–374, Feb. 2003.

[12] 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, pp. 418–435, Mar. 1992.

[13] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 226–239, Mar. 1998.

[14] H. Altincay and M. Demirekler, "An information theoretic framework for weight estimation in the combination of probabilistic classifiers for speaker identification," *Speech Commun.*, vol. 30, no. 4, pp. 255–272, 2000.

[15] S. Hashem and B. Schmeiser, "Improving model accuracy using optimal linear combinations of trained neural networks," *IEEE Trans. Neural Networks*, vol. 6, pp. 792–794, Mar. 1995.

[16] S. N. Geok and H. Singh, "Democracy in pattern classifications: Combinations of votes from various pattern classifiers," *Artif. Intell. Eng.*, vol. 12, no. 3, pp. 189–204, 1998.

[17] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision combination in multiple classifier systems," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 66–75, Jan. 1994.

[18] L. Lam and C. Y. Suen, "Optimal combinations of pattern classifiers," *Pattern Recognit. Lett.*, vol. 16, no. 9, pp. 945–954, 1995.

[19] A. Baykut and A. Ercil, "Toward automated classifier combination for pattern recognition," in *Lecture Notes in Computer Science*, T. Windeatt and F. Roli, Eds. Berlin, Germany: Springer-Verlag, 2003, vol. 2709, Proc. Multiple Classifier Systems—4th Int. Workshop, MCS 2003, pp. 94–105.

[20] K. Woods, W. P. Kegelmeyer Jr., and K. Bowyer, "Combination of multiple classifiers using local accuracy estimates," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 405–410, Apr. 1997.

[21] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," *Neural Computation*, vol. 6, pp. 181–214, 1994.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc.*, ser. B, vol. 39, no. 1, pp. 1–38, 1977.

[23] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Mag.*, vol. 13, no. 6, pp. 47–60, 1996.

[24] S. K. Warfield, K. H. Zou, and W. M. Wells, "Validation of image segmentation and expert quality with an expectation-maximization algorithm," in *Lecture Notes in Computer Science*, T. Dohi and R. Kikinis, Eds. Berlin, Germany: Springer-Verlag, 2002, vol. 2488, Proc. 5th Int. Conf. Medical Image Computing and Computer-Assisted Intervention (MICCAI), Pt. I, pp. 298–306.

[25] ——, "Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation," *IEEE Trans. Med. Imag.*, vol. 23, pp. 903–921, July 2004.

[26] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE Trans. Med. Imag.*, vol. 16, pp. 187–198, Apr. 1997.

[27] J. Kittler and F. M. Alkoot, "Sum versus vote fusion in multiple classifier systems," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 110–115, Jan. 2003.

[28] P. M. Thompson and A. W. Toga, "Detection, visualization, and animation of abnormal anatomic structure with a deformable probabilistic brain atlas based on random vector field transformations," *Med. Image Anal.*, vol. 1, no. 4, pp. 271–294, 1997.

[29] 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, pp. 712–721, Aug. 1999.

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

[31] T. Rohlfing, D. B. Russakoff, and C. R. Maurer Jr., "Expectation maximization strategies for multi-atlas multi-label segmentation," in *Lecture Notes in Computer Science*, C. Taylor and J. A. Noble, Eds., Berlin, Heidelberg, 2003, 18th Int. Conf., Information Processing in Medical Imaging (IPMI 2003), pp. 210–221.

[32] T. Rohlfing and C. R. Maurer Jr., "Nonrigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees," *IEEE Trans. Inform. Technol. Biomed.*, vol. 7, pp. 16–25, Mar. 2003.

[33] T. Rohlfing, R. Brandt, C. R. Maurer Jr., and R. Menzel, "Bee brains, B-splines, and computational democracy: Generating an average shape atlas," in *Proc. IEEE Workshop Mathematical Methods in Biomedical Image Analysis*, L. Staib, Ed., Kauai, HI, 2001, pp. 187–194.

[34] Y. S. Huang and C. Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 90–94, Jan 1995.