

An Adaptive Spatial Fuzzy Clustering Algorithm for 3-D MR Image Segmentation

Author

Liew, AWC, Yan, H

Published

2003

Journal Title

IEEE Transactions on Medical Imaging

DOI

<https://doi.org/10.1109/TMI.2003.816956>

Copyright Statement

© 2003 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Downloaded from

<http://hdl.handle.net/10072/15104>

Link to published version

<http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=42>

Griffith Research Online

<https://research-repository.griffith.edu.au>

An Adaptive Spatial Fuzzy Clustering Algorithm for 3-D MR Image Segmentation

Alan Wee-Chung Liew*, *Member, IEEE*, and Hong Yan, *Senior Member, IEEE*

Abstract—An adaptive spatial fuzzy c-means clustering algorithm is presented in this paper for the segmentation of three-dimensional (3-D) magnetic resonance (MR) images. The input images may be corrupted by noise and intensity nonuniformity (INU) artifact. The proposed algorithm takes into account the spatial continuity constraints by using a dissimilarity index that allows spatial interactions between image voxels. The local spatial continuity constraint reduces the noise effect and the classification ambiguity. The INU artifact is formulated as a multiplicative bias field affecting the true MR imaging signal. By modeling the log bias field as a stack of smoothing B -spline surfaces, with continuity enforced across slices, the computation of the 3-D bias field reduces to that of finding the B -spline coefficients, which can be obtained using a computationally efficient two-stage algorithm. The efficacy of the proposed algorithm is demonstrated by extensive segmentation experiments using both simulated and real MR images and by comparison with other published algorithms.

Index Terms—Adaptive spatial fuzzy clustering, intensity nonuniformity correction, MR image segmentation, spatial continuity constraint, spline approximation.

I. INTRODUCTION

A N IMPORTANT first step in image analysis is image segmentation, or separation of the input image into meaningful regions. In medical imaging, this could involve organ detection or tissue characterization. A commonly used image segmentation method is the fuzzy c-means (FCM) clustering algorithm [1]–[4], which assigns pixels in the image into different classes according to their features.

In image segmentation, we expect the pixels in the same class to have similar pixel values independent of their locations. However, in magnetic resonance imaging (MRI), inhomogeneity in the magnetic field usually give rises to the so-called intensity nonuniformity (INU) artifact [5]. This common artifact exhibits itself as a smooth, slowly varying change in image pixel values and could have adverse effect on the performance of intensity-based automatic segmentation methods [37], [38]. In addition to intensity variation due to field inhomogeneity, there may be

a lack of tissue specific meanings for MRI intensities within scans, even for the same patient obtained on the same scanner using the same protocol. Data preprocessing using a tissue intensity calibration procedure as proposed in [6] prior to the segmentation process would ensure a more accurate segmentation.

Besides INU artifact, issues such as poor contrast and imaging noise also make accurate segmentation difficult. In addition, many pixels in a real image are ambiguous and cannot be classified consistently based on feature attributes alone. In an image, pixels of the same object usually form coherent patches. Thus, the incorporation of local spatial information in the clustering process could filter out noise and artifacts and reduce classification ambiguities.

Several methods have been proposed to correct the INU artifact [7]–[15], [30]–[33]. In [7], [8], and [11], a polynomial surface and a thin-plate spline surface are used to approximate the bias field associated with the INU, respectively. In [9], the bias field for the INU is estimated by sharpening the image histogram in an iterative process. In these methods, after correcting the INU artifact, the MR image is then segmented using an algorithm that assumes that the inhomogeneity is not present. In [12], [13], and [33], the bias field is estimated by homomorphic filtering. In [10], the problem of estimating the bias field is cast in a Bayesian framework and the EM algorithm is used to estimate the inhomogeneity and the tissue classes. In [39] and [40], the MRI brain tissue classes are modeled as finite Gaussian mixtures with Markov random field regularization for contextual information and *a priori* digital brain atlas initialization, whereas the bias field is modeled as a fourth order least square polynomial fit. A method of estimating the INU based on fuzzy clustering has been reported in [15], where intermediate segmentation results are utilized for the INU estimation. The method uses a modified FCM cost functional to model the variation in intensity values via a multiplicative bias field applied to the cluster centroids. The computation of the bias field is formulated as a variational problem and the bias field is estimated at every voxel using a multigrid algorithm.

Many attempts have also been made to introduce spatial context into the classification and segmentation procedures. One popular approach is the relaxation labeling method [16]–[18]. However, relaxation labeling requires the initial labeling probabilities of each pixel to be available. In [19], a spatial continuity constraint is incorporated into the fuzzy clustering algorithm by either the addition of a small positive constant to, or subtraction of a small positive constant from, the membership value of the center pixel in a 3×3 window. The decision depends on whether the optimal cluster assignment for the pixel in the 8-neighborhood is the same as, or different from, that

Manuscript received June 29, 2001; revised March 18, 2003. This work was supported by the Hong Kong Research Grant Council under Project CityU 1088/00E. The Associate Editor responsible for coordinating the review of this paper and recommending its publication was C. Meyer. *Asterisk indicates corresponding author.*

*A. W. C. Liew is with the Department of Computer Engineering and Information Technology, City University of Hong Kong, Hong Kong (e-mail: itwcliew@cityu.edu.hk).

H. Yan is with the Department of Computer Engineering and Information Technology, City University of Hong Kong, Hong Kong, and School of Electrical and Information Engineering, University of Sydney, NSW 2006, Australia (e-mail: ityan@cityu.edu.hk).

Digital Object Identifier 10.1109/TMI.2003.816956

of the center pixel. Recently, a supervised segmentation technique using the idea of fuzzy-connectedness has been proposed in [21] and [22], which take into account pixel affinities and spatial continuities using the notion of fuzzy “hanging togetherness.” Given some prelabeled pixels from each of the object classes, the fuzzy-connectedness map of pairs of pixels is computed. The segmentation can then be obtained by appropriate thresholding of the fuzzy-connectedness map.

In this paper, we proposed an FCM-based algorithm that addresses both the INU artifact and the local spatial continuity. Our method incorporates the local spatial continuity into the clustering algorithm using a dissimilarity index [20], in place of the usual distance metric. The log of the three-dimensional (3-D) multiplicative bias field is modeled by a stack of smoothing B -spline surfaces and estimated by an efficient two-stage algorithm. Although the method in [15] is also FCM-based, our approach is different in several aspects. First, the bias field is derived according to the commonly used multiplicative model [7]–[11], [14]. Second, the 3-D log bias field is modeled as a stack of smoothing B -spline surfaces, instead of the solution to a variational problem. Finally, the spatial continuity constraint is taken into account in our fuzzy objective function. Extensive experiments using both simulated and real MR brain image data show that the proposed algorithm can suppress INU and noise to produce good segmentation results.

II. PROBLEM FORMULATION

The task of 3-D MR image segmentation involves the separation of image voxels into regions comprising different tissue types. Let $\underline{x} = (x, y, z)$ be the 3-D image coordinate of a voxel. We assume that each tissue class k has a specific value v_k , that is, a quantity being measured. Then, the ideal signal $o(\underline{x})$ would consist of piecewise constant regions,¹ each having one of the v_k values. However, imperfection in the magnetic field often introduces an unwanted low frequency bias term into the signal, which gives rise to the INU artifact. The bias field that gives rise to the INU artifact in an MR image is usually modeled as a smooth multiplicative field. This model is widely used [7]–[11], [14] and is consistent with the spatial inhomogeneity arising from the variation in the sensitivity of the RF coils and the nonuniform excitations. The image formation process in MRI can be modeled as

$$s(\underline{x}) = o(\underline{x})b(\underline{x}) + n(\underline{x}) \quad (1)$$

where $s(\underline{x})$ is the measured MR signal, $o(\underline{x})$ is the true signal emitted by the tissue, $b(\underline{x})$ is the unknown smoothly varying bias field, and $n(\underline{x})$ is an additive noise assumed to be independent of $b(\underline{x})$. Accurate segmentation of an MR image thus involves an accurate estimation of the unknown bias field $b(\underline{x})$ and removing this bias field from the measured MR signal. Using the

estimated $\tilde{b}(\underline{x})$, the log-transformed true signal can be recovered as

$$\log \tilde{o}(\underline{x}) = \log s(\underline{x}) - \log \tilde{b}(\underline{x}) \approx \log \left(o(\underline{x}) + \frac{n(\underline{x})}{\tilde{b}(\underline{x})} \right). \quad (2)$$

III. METHODS

A. Conventional FCM Segmentation

The FCM clustering algorithm assigns a fuzzy membership value to each data point based on its proximity to the cluster centroids in the feature space [1]. Let $S = \{s(\underline{x})\}$ be the set of feature vectors associated with a 3-D image defined in the domain I . The conventional FCM algorithm is formulated as the minimization of the objective functional J_{FCM} with respect to the membership values U and cluster centroids v

$$J_{\text{FCM}}(U, v) = \sum_{\underline{x} \in I} \sum_{k=1}^c u_{k,\underline{x}}^m d_{k,\underline{x}}^2$$

subject to $\sum_{k=1}^c u_{k,\underline{x}} = 1 \quad \forall \underline{x} \in I \quad (3)$

where the matrix $U = \{u_{k,\underline{x}}\}$ is a fuzzy c -partition of S , $v = \{v_1, v_2, \dots, v_c\}$ is the set of fuzzy cluster centroids, $m \in (1, \infty)$ is the fuzzy index, c is the total number of clusters, and $u_{k,\underline{x}}$ gives the membership of pixel $s(\underline{x})$ in the k th cluster c_k . Using the Euclidean norm, the distance metric d measures the vector distance of a feature vector from a cluster centroid in the feature space, i.e.,

$$d_{k,\underline{x}}^2 = \|s(\underline{x}) - v_k\|^2. \quad (4)$$

The FCM objective function is minimized when high membership values are assigned to $s(\underline{x})$ that are close to the centroid for their particular class, and low membership values are assigned when they are far from the centroid. Letting the first derivatives of J_{FCM} with respect to u and v equal to zero yields the two necessary conditions for minimizing J_{FCM} . The FCM algorithm proceeds by iterating the two necessary conditions until a solution is reached. After FCM clustering, each data sample will be associated with a membership value for each class. By assigning the data sample to the class with the highest membership value, a segmentation of the data can be obtained.

In the conventional FCM formulation, each class is assumed to have a uniform value as given by its centroid. Each data point is also assumed to be independent of every other data point and spatial interaction between data points is not considered. However, for image data, there is strong correlation between neighboring pixels. In addition, due to the INU artifact, the data in a class no longer have a uniform value. Therefore, to produce meaningful segmentation, the conventional FCM algorithm has to be modified to take into account both local spatial continuity between neighboring data and INU artifact compensation.

B. Incorporation of Local Spatial Continuity

The incorporation of local spatial continuity considers the influence of neighboring voxels on the center voxel of interest during classification [20]. Let \mathfrak{N}_ρ denote a chosen 3-D local

¹In practice, the limited resolution of the imaging device leads to blurring along border regions between tissue classes, i.e., the partial volume effect. However, this effect is confined to the border regions, in contrast to the more global INU artifact.

neighborhood configuration with respect to a center voxel ρ . If the voxels in \aleph_ρ and the center voxel ρ belong to the same class, then ρ should be smoothed by the clustering results of its neighboring voxels so that they all eventually have high and similar membership values in one of the clusters. This is done as follows. Let $dist(a, b) = \sqrt{\|a - b\|^2}$ denote the L_2 distance between vectors a and b . For every voxel $s(\underline{x})$ in the 3-D MR image, we compute the following L_2 distances

$$\partial_{\underline{x}, \underline{y}} = dist(s(\underline{x}), s(\underline{y})), \quad s(\underline{y}) \in \aleph_{\underline{x}} \quad (5)$$

$$d_{k, \underline{x}} = dist(s(\underline{x}), v_k) \quad (6)$$

where $\aleph_{\underline{x}}$ is the neighborhood of $s(\underline{x})$ and v_k is the centroid of the k th cluster. Now, if the distance $\partial_{\underline{x}, \underline{y}}$ is small, we would like $d_{k, \underline{x}}$ to be greatly influenced by $d_{k, \underline{y}}$. Otherwise, $d_{k, \underline{x}}$ should be largely independent of $d_{k, \underline{y}}$. Taking all voxels in $\aleph_{\underline{x}}$ into account, we define a dissimilarity index $D_{k, \underline{x}}$ which measures the dissimilarity between $s(\underline{x})$ and the k th cluster centroid v_k , as

$$D_{k, \underline{x}} = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y} \in \aleph_{\underline{x}}} \left[d_{k, \underline{x}}^2 \lambda_{\underline{x}, \underline{y}} + d_{k, \underline{y}}^2 (1 - \lambda_{\underline{x}, \underline{y}}) \right] \quad (7)$$

where $|\aleph_{\underline{x}}|$ is the cardinality of the neighborhood configuration, and $\lambda(\partial_{\underline{x}, \underline{y}}) = \lambda_{\underline{x}, \underline{y}}$, with ranges between zero and one, is the weighting factor controlling the degree of influence of the neighboring voxels $s(\underline{y}) \in \aleph_{\underline{x}}$ on the center voxel, defined by

$$\lambda(\partial) = \frac{1}{1 + e^{-(\partial - \mu)/\sigma}}. \quad (8)$$

The parameters μ and σ specify the displacement of λ from zero, and the steepness of the sigmoid curve, respectively.

The parameter μ can be viewed as measuring the average ‘‘randomness’’ of the homogeneous region with respect to the chosen neighborhood $\aleph_{\underline{x}}$. Assuming that the majority of $\aleph_{\underline{x}}$ fall on homogeneous regions, the parameter μ can be computed by

$$\mu = \frac{1}{|I|} \sum_{\underline{x} \in I} \partial_{av}(\underline{x}) \quad (9)$$

$$\text{where } \partial_{av}(\underline{x}) = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y} \in \aleph_{\underline{x}}} \partial_{\underline{x}, \underline{y}}. \quad (10)$$

Thus, when the difference between $s(\underline{x})$ and its neighbor $s(\underline{y})$ is much larger than the average ‘‘randomness’’, i.e., $\partial \gg \mu$, $s(\underline{x})$ and $s(\underline{y})$ are less likely to belong to the same class, and the influence of $s(\underline{y})$ on the center voxel $s(\underline{x})$ is suppressed in $D_{k, \underline{x}}$.

The steepness parameter σ controls the degree of influence of the neighboring voxels on the center voxel. Clearly, σ should be chosen such that the clustering results of important image structures are not smoothed out, i.e., $\lambda(\partial) \approx 1$ when ∂ is due to genuine structures, such as region borders or edges, in the image. We determine σ as follows. From the $\partial_{av}(\underline{x})$ computed over the image data, we take ∂_t equal to the 95 percentile of $\partial_{av}(\underline{x})$. Then we let $\lambda(\partial_t) = 0.8$ and solve for σ using (8).

The dissimilarity index $D_{k, \underline{x}}$ effectively smoothes the cluster assignment of $s(\underline{x})$ by the cluster assignment of its neighboring voxels. When $s(\underline{x})$ is along an edge, its value will be very different from that of its neighbors, reflecting that they are unlikely

to belong to the same class. Hence, ∂ will be large and $\lambda \rightarrow 1$ for all its neighbors. In this case, $D_{k, \underline{x}} \approx d_{k, \underline{x}}^2$ i.e., neighboring influence is turned off. When $\aleph_{\underline{x}}$ falls on a step boundary, $s(\underline{x})$ is only affected by those neighboring voxels in the same class (i.e., neighboring voxels on the same step level as $s(\underline{x})$). When $s(\underline{x})$ is on a smooth region and is affected by all its neighbors, the influence of each neighbor $s(\underline{y})$ on $s(\underline{x})$ is affected by the distance between them, i.e., the distance $\partial_{\underline{x}, \underline{y}}$, through the weighting λ . Hence, $D_{k, \underline{x}}$ enables spatial interaction between neighboring voxels and is adaptive to image content.

The spatial continuity constraint also has a noise suppression capability due to the adaptive smoothing operation. Random noise would either increase or decrease the distance of the center voxel and the distances of its neighbors to the cluster centroids randomly. When the weighted average of these distances, i.e., (7), is taken, the effect of random noise is smoothed out.

Finally, we would like to point out that the incorporation of local spatial continuity actually takes into account explicitly the spatial dimensionality of the data. Without the local spatial continuity constraint, the FCM clustering algorithm is oblivious to the spatial arrangement of the data, i.e., the FCM algorithm just treats each data point as an independent instance, regardless of whether the data are from two-dimensional (2-D), 3-D, or from N -dimensional space. Therefore, the incorporation of local spatial continuity into the FCM algorithm is well justified for data with a dimensionality interpretation such as 3-D MRI data.

C. INU Bias Field Compensation

When a bias field is present, the piecewise constant signal assumption of the MRI data is no longer valid. In view of the MR image formation model of (1), the data should be compensated for the bias field when computing the L_2 distance between the data and the cluster centroids, i.e., $d_{k, \underline{x}}^2$ should be given by

$$d_{k, \underline{x}}^2 = \left\| \frac{s(\underline{x})}{\tilde{b}(\underline{x})} - v_k \right\|^2 \quad (11)$$

where $\tilde{b}(\underline{x})$ is the estimate for the unknown bias field $b(\underline{x})$. Substituting (11) into (3) (or into (7) if spatial continuity is imposed) and by incorporating a regularizing term, we can formulate the computation of the bias field as a variational problem.

However, the variational formulation has several disadvantages. First, the number of unknowns is equal to the image dimension, so the problem becomes computationally very expensive. Moreover, the resulting system of equations is not spatial invariant in $\tilde{b}(\underline{x})$, and expensive iterative numerical procedures such as a multigrid method or a gradient descent method is needed to reach a solution. Finally, the system of equations is ill-conditioned and error prone, making convergence difficult.

Instead of estimating the bias field $b(\underline{x})$ in (11) directly, we estimate its log-transformation. This results in a simpler expression and implementation. Let $w(\underline{x}) = \log \tilde{b}(\underline{x})$. We model the 3-D log bias field $w(\underline{x})$ as a stack of 2-D spline surfaces $\{w_z(x, y)\}$, where each of the spline surfaces $w_z(x, y)$ is computed over the 2-D x - y plane at the particular z index. Then, we employ a novel technique that couples the 2-D surfaces together, such that they form a smooth 3-D field. This approach reduces

computation time significantly, and at the same time produces a good estimate of the actual 3-D field, as will be shown later.

Specifically, we consider the cubic B -spline [23], which has a continuous derivative up to the second order. The normalized cubic B -spline basis $N_{i,4}$ with knots $\lambda_i, \dots, \lambda_{i+4}$ is given by

$$N_{i,4}(x) = (\lambda_{i+4} - \lambda_i) \sum_{j=0}^4 \frac{(\lambda_{i+j} - x)_+^3}{\prod_{\substack{l=0 \\ l \neq j}}^4 (\lambda_{i+j} - \lambda_{i+l})} \quad (12)$$

where

$$(\lambda_{i+j} - x)_+^3 = \begin{cases} (\lambda_{i+j} - x)^3, & \text{if } \lambda_{i+j} \geq x \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

The 2-D log bias field $w_z(x, y)$ at index z is formed by the tensor products of cubic B -spline bases, i.e.,

$$w_z(x, y) = \sum_{i=-3}^g \sum_{j=-3}^h \alpha_{ij}^z N_{i,4}(x) M_{j,4}(y) \quad (14)$$

with the knot sequences $\{\lambda_{-3}, \lambda_{-2}, \dots, \lambda_g\}$ and $\{\mu_{-3}, \mu_{-2}, \dots, \mu_h\}$. The superscript z on the spline coefficients $\{\alpha_{ij}^z\}$ denotes that they are for the spline surface at index z . The spline surface $w_z(x, y)$ is assumed to have coincident boundary knots, i.e., for x dimension spanning $[a, b]$ and for y dimension spanning $[c, d]$

$$\begin{aligned} \lambda_{-3} &= \lambda_{-2} = \lambda_{-1} = \lambda_0 = a \\ b &= \lambda_{g+1} = \lambda_{g+2} = \lambda_{g+3} = \lambda_{g+4} \\ \mu_{-3} &= \mu_{-2} = \mu_{-1} = \mu_0 = c \\ d &= \mu_{h+1} = \mu_{h+2} = \mu_{h+3} = \mu_{h+4}. \end{aligned} \quad (15)$$

With this choice, all B -spline bases vanish outside the region $[a, b] \times [c, d]$. Using the local support property $N_{i,4}(x) = 0$ if $x \notin [\lambda_i, \lambda_{i+4}]$, the tensor product B -splines can be shown to be

$$\begin{aligned} N_{i,4}(x) M_{j,4}(y) &\geq 0 \text{ for all } x, y \in R \\ &= 0 \text{ for all } x, y \notin [\lambda_i, \lambda_{i+4}] \times [\mu_j, \mu_{j+4}]. \end{aligned} \quad (16)$$

Using the tensor product spline representation of $w_z(x, y)$, the computation of the log bias field becomes that of finding the set of B -spline coefficients $\{\alpha_{ij}^z\}$. Let the x and y dimensions be divided into t_x and t_y intervals, respectively. Then the number of B -spline coefficients to be computed is $(t_x + 3) \times (t_y + 3)$. Since the log bias field to be estimated is smooth and slowly varying, the number of intervals needed is small. Thus, the number of unknown B -spline coefficients is much less than the number of unknowns in the variational formulation.

D. The Proposed Adaptive Spatial FCM

The objective function for the proposed adaptive spatial FCM (ASFCM) clustering algorithm is given by

$$J_{\text{ASFCM}} = \sum_{\underline{x} \in I} \sum_{k=1}^c u_{k,\underline{x}}^m D_{k,\underline{x}} + \beta \eta(w_z(x, y)) + \gamma \varphi(w_z(x, y)) \quad (17)$$

with $D_{k,\underline{x}}$ defined by (7) and (11), subject to

$$\sum_{k=1}^c u_{k,\underline{x}} = 1 \quad \forall \underline{x} \in I \quad (18)$$

where the first regularizing term is given by

$$\begin{aligned} \eta(w_z(x, y)) &= \\ &\iint \left\{ \left[\frac{\partial^2 w_z(x, y)}{\partial x^2} \right]^2 + 2 \left[\frac{\partial^2 w_z(x, y)}{\partial x \partial y} \right]^2 + \left[\frac{\partial^2 w_z(x, y)}{\partial y^2} \right]^2 \right\} dx dy \end{aligned} \quad (19)$$

and the second regularizing term is given by

$$\varphi(w_z(x, y)) = \iint \left[\frac{\partial^2 w_z(x, y)}{\partial z^2} \right]^2 dx dy. \quad (20)$$

The first regularizing term of (19) minimizes the thin plate energy of each of the spline surfaces $w_z(x, y)$. Although the smoothness of the spline surface $w_z(x, y)$ can be ensured to some degree by using fewer knots in the x - y plane, the incorporation of (19) further minimizes the variation of the spline surface. This is important since we are seeking a smoothing spline surface fitting instead of an interpolating spline surface fitting.²

The second regularizing term of (20) forces smoothness between slices of spline surfaces. It couples the slices together to form a smooth 3-D field. The parameters β and γ control the fidelity of the fit to the data and the smoothness of the field. Note that due to the functional representation of the surface using B -splines, the smooth functional η can be evaluated analytically during implementation.

An important observation about the objective function (17) is that the two regularizing terms only involve double integration over the x - y plan, instead of the usual 3-D triple integration $\iiint f(\cdot) dx dy dz$. This formulation allows the spline surfaces to be estimated slice by slice, resulting in great computational saving without compromising the accuracy of the estimated field, as we will show later. It also ensures that smoothness is forced onto each individual slice, as well as globally over the entire 3-D domain.

The necessary conditions for the minimization of J_{ASFCM} over the memberships $u_{k,\underline{x}}$, cluster centroids v_k , and B -spline coefficients α_{ij}^z are obtained by setting the respective first partial derivatives of J_{ASFCM} to zero while keeping the other variables constant.

Consider the following Lagrange functional

$$L(u) = J_{\text{ASFCM}}(u) + \sum_{\underline{x} \in I} \delta_{\underline{x}} \left(1 - \sum_{k=1}^c u_{k,\underline{x}} \right) \quad (21)$$

where δ is the Lagrange multiplier. Differentiating L with respect to $u_{k,\underline{x}}$, setting the result to zero and using (18) yields

$$u_{k,\underline{x}} = \left[\sum_{l=1}^c \left(\frac{D_{k,\underline{x}}}{D_{l,\underline{x}}} \right)^{1/m-1} \right]^{-1}. \quad (22)$$

²See [23] for a detail exposition of the two classes of surface fitting problem.

Differentiating J_{ASFCM} with respect to v_k and setting the result to zero yields,

$$v_k = \frac{\sum_{\underline{x} \in I} u_{k,\underline{x}}^m g_{\underline{x}}}{\sum_{\underline{x} \in I} u_{k,\underline{x}}^m} \quad (23)$$

$$\text{where } g_{\underline{x}} = \frac{1}{|\mathbb{N}_{\underline{x}}|} \sum_{\underline{y} \in \mathbb{N}_{\underline{x}}} \left[\lambda_{\underline{x},\underline{y}} \left(\frac{s(\underline{x})}{\hat{b}(\underline{x})} \right) + (1 - \lambda_{\underline{x},\underline{y}}) \left(\frac{s(\underline{y})}{\hat{b}(\underline{y})} \right) \right]. \quad (24)$$

In deriving the first derivative of J_{ASFCM} with respect to the B -spline coefficients $\{\alpha_{ij}^z\}$, we make two modifications to the objective function of (17). The first modification involves ignoring the spatial interactions between neighboring voxels. Although it is straightforward to include the spatial influence into the derivation, doing so would increase the computation cost while having negligible effect on the estimation of the bias field. This is because the spline surface we are trying to estimate is already very smooth, so that the additional noise smoothing effect offered by the spatial interactions between neighboring voxels is insignificant by comparison.

The second modification is to replace the original $d_{k,\underline{x}}^2$ of (11) by the following expression:

$$d_{k,\underline{x}}^2 = \|\hat{s}(\underline{x}) - w(\underline{x}) - \hat{v}_k\|^2 \quad (25)$$

where $w(\underline{x})$ is the log bias field, $\hat{s}(\underline{x}) = \log s(\underline{x})$ and $\hat{v}_k = \log v_k$. This is equivalent to estimating the bias field in the log domain. The replacement of $d_{k,\underline{x}}^2$ is valid mathematically since: 1) the variables $u_{k,\underline{x}}$ and v_k are held fixed when computing the B -spline coefficients $\{\alpha_{ij}^z\}$; 2) the solutions of $\{\alpha_{ij}^z\}$ are obtained by a direct least square method instead of an iterative gradient descent minimization method, where the latter is sensitive to the expression and initialization. This modification results in a much simpler expression for the data term as shown in (27)–(29), whereas the original $d_{k,\underline{x}}^2$ of (11) results in the complicated data term involving inverse powers, as shown in (26) at the bottom of the page, where (26) is over one slice of the B -spline surface.

To find an expression for $\{\alpha_{ij}^z\}$, we fix $u_{k,\underline{x}}$ and v_k , and discretize the second regularizing term using finite difference, i.e.,

$$\iint \left[\frac{\partial^2 w_z}{\partial z^2} \right]^2 dx dy = \iint [w_{z-2} - 4w_{z-1} + 6w_z - 4w_{z+1} + w_{z+2}]^2 dx dy. \quad (27)$$

Then, differentiating the modified J_{ASFCM} with respect to α_{pq}^z for a particular $w_z(x, y)$, and setting the result to zero, yields the set of linear equations, shown in (28) at the bottom of the page, for all $p = -3, \dots, g$ and $q = -3, \dots, h$, where

$$L_z(p, q, i, j) = \sum_{(x,y) \in I} \sum_{k=1}^c u_{k,\underline{x}}^m N_{p,4}(x) M_{q,4}(y) N_{i,4}(x) M_{j,4}(y) \quad (29)$$

$$b^z(p, q) = \sum_{(x,y) \in I} \sum_{k=1}^c u_{k,\underline{x}}^m (\hat{s}(\underline{x}) - \hat{v}_k) N_{p,4}(x) M_{q,4}(y) \quad (30)$$

$$D_z(\alpha_{i,j}) = \alpha_{ij}^{z-2} - 4\alpha_{ij}^{z-1} + 6\alpha_{ij}^z - 4\alpha_{ij}^{z+1} + \alpha_{ij}^{z+2} \quad (31)$$

$$\omega_0(p, q, i, j) = \int N_{i,4}(x) N_{p,4}(x) dx \int M_{j,4}(y) M_{q,4}(y) dy \quad (32)$$

$$\omega_1(p, q, i, j) = \int N_{i,4}''(x) N_{p,4}''(x) dx \int M_{j,4}(y) M_{q,4}(y) dy \quad (33)$$

$$\omega_2(p, q, i, j) = \int N_{i,4}'(x) N_{p,4}'(x) dx \int M_{j,4}'(y) M_{q,4}'(y) dy \quad (34)$$

$$\omega_3(p, q, i, j) = \int N_{i,4}(x) N_{p,4}(x) dx \int M_{j,4}''(y) M_{q,4}''(y) dy \quad (35)$$

where the single and double prime in N and M denote first and second derivatives, respectively. The first curly bracket in (28) corresponds to the data term, whereas the second curly bracket corresponds to the regularizing terms. Unlike the complicated form of (26), the data term in (28) represents a weighted least square smoothing B -spline surface fitting to the residual signal $res(\underline{x}) = \sum_{k=1}^c u_{k,\underline{x}}^m (\hat{s}(\underline{x}) - \hat{v}_k)$. Note also that the finite difference of the spline surfaces in (27) becomes the finite difference between the spline coefficients in (31).

Equation (28) indicates that we are trying to fit a smoothing spline surface $w_z(x, y)$ to the 3-D residual signal $res_z(\underline{x})$ be-

$$\sum_{(x,y) \in I} \sum_{k=1}^c u_{k,\underline{x}}^m \left[s(\underline{x}) \left(\sum_{i=-3}^g \sum_{j=-3}^h \alpha_{ij}^z N_{i,4}(x) M_{j,4}(y) \right)^{-3} - v_k s(\underline{x}) \left(\sum_{i=-3}^g \sum_{j=-3}^h \alpha_{ij}^z N_{i,4}(x) M_{j,4}(y) \right)^{-2} \right] N_{p,4}(x) M_{q,4}(y) \quad (26)$$

$$\left\{ \sum_{i=-3}^g \sum_{j=-3}^h \alpha_{ij}^z L_z(p, q, i, j) - b^z(p, q) \right\} + \left\{ \sum_{i=-3}^g \sum_{j=-3}^h [\beta \alpha_{ij}^z (\omega_1(p, q, i, j) + 2\omega_2(p, q, i, j) + \omega_3(p, q, i, j)) + \gamma D_z(\alpha_{i,j}) \omega_0(p, q, i, j)] \right\} = 0 \quad (28)$$

tween the actual data and a piecewise constant FCM solution at a particular z index, i.e., from (28)

$$\sum_{(x,y) \in I} [Err_z(x,y)] N_{p,4}(x) M_{q,4}(y) + \text{smooth terms} = 0 \quad (36)$$

where the fitting error is given by $Err_z(x,y)$

$$Err_z(x,y) = \sum_{k=1}^c u_{k,\underline{x}}^m \left(\sum_{i=-3}^g \sum_{j=-3}^h \alpha_{i,j} N_{i,4}(x) M_{j,4}(y) \right) res_z(\underline{x}). \quad (37)$$

Since $res_z(\underline{x})$ is obtained by a 3-D FCM-based algorithm, it would inherit the within-slice and between-slice continuity from the 3-D data. Fitting slice-wise smoothing 2-D splines over the residual would therefore not incur significant discontinuity between slices, even without the second regularizing term of (20), as we have observed experimentally. Nevertheless, (20) explicitly forces the stack of spline surfaces to be smooth over the z direction. Another computational advantage of being able to identify the residual signal $res_z(\underline{x})$ in the formulation is that local smoothing in the z direction can be applied to $res_z(\underline{x})$, such that the iterative procedure we used to enforce (20) can converge faster. We note that (26) does not allow such simple interpretation and manipulation, since no such residual signal can be easily identified in (26). The “residual signal fitting by smoothing splines” interpretation of our INU correction method also allows the procedure to be used as an efficient bias field estimation technique in data preprocessing [7]–[9], [11]–[14] independent of the FCM discussed here, prior to applying any of the existing segmentation methods.

IV. IMPLEMENTATION

To compute the spline coefficients $\{\alpha_{ij}^z\}$, we proposed a novel two-stage algorithm. In the first stage, the second regularizing term of (20) is ignored. The bias field is estimated slice by slice, with no explicit coupling between adjacent slices of spline surface. In the second stage, an iterative procedure is used, whereby the previously computed $\{\alpha_{ij}^z\}$ is updated iteratively, taking into account the explicit coupling between adjacent spline surfaces resulting from the second regularizing term of (20).

A. Slice-Wise Spline Surface Computation

As the resultant system of equations is linear, the B -spline coefficients $\{\alpha_{ij}^z\}$ can be solved efficiently using the direct least squares approach. By using the ordering $t = (p+3) \times (h+4) + (q+4)$ and $u = (i+3) \times (h+4) + (j+4)$, we rearrange $L_z(p,q,i,j)$, $\omega_k(p,q,i,j)$, $k = 1, 2, 3$, into square matrices L_z and $\{\Omega_k\}$ with indices t and u . When $g = h$, i.e., the number of knots in the x and y dimensions are equal, L_z is a symmetric matrix with $L_z = L_z^T$. Hence, not all elements in L_z need to be computed. In addition, using the local support property of (16), the summations in (29) can be restricted to be over a small region in the image domain, instead of over the entire image dimension. The matrices $\{\Omega_k\}$ need to be calculated only once, since they do not change during the FCM iteration. When the x and y dimensions of the image are equal, symmetry can also

be observed in $\{\Omega_k\}$. Note that due to the use of tensor splines, the double integral in (19) and (20) is separable, resulting in the separable integrals of (32)–(35), which can be computed efficiently. The B -spline values $N_{i,4}(x)$ can be evaluated in a numerically stable way using the recurrence relation

$$N_{i,l+1}(x) = \frac{x - \lambda_i}{\lambda_{i+l} - \lambda_i} N_{i,l}(x) + \frac{\lambda_{i+l+1} - x}{\lambda_{i+l+1} - \lambda_{i+1}} N_{i+1,l}(x),$$

$$N_{i,1}(x) = \begin{cases} 1, & \text{if } x \in [\lambda_i, \lambda_{i+1}) \\ 0, & \text{if } x \notin [\lambda_i, \lambda_{i+1}) \end{cases}. \quad (38)$$

while the first and second derivatives of $N_{i,4}(x)$ and $M_{j,4}(y)$ can be computed using the following recursion:

$$N'_{i,k+1}(x) = k \left\{ \frac{N_{i,k}(x)}{\lambda_{i+k} - \lambda_i} - \frac{N_{i+1,k}(x)}{\lambda_{i+k+1} - \lambda_{i+1}} \right\}. \quad (39)$$

With L_z and $\{\Omega_k\}$, the set of $(g+4) \times (h+4)$ equations from (28), ignoring the second regularizing term, can be expressed in matrix notation

$$A_z \theta_z = b_z \quad (40)$$

where A_z is the $(g+4) \times (h+4)$ by $(g+4) \times (h+4)$ sparse matrix given by $L_z + \beta(\Omega_1 + 2\Omega_2 + \Omega_3)$, θ_z is the vector of B -spline coefficients $\{\alpha_{ij}^z\}$ arranged using the ordering u , and b_z is the vector obtained from (30) using the ordering t .

The solution to (40) can be obtained by singular value decomposition (SVD) [24]. Although A_z is usually of full rank, the SVD will provide a minimum norm solution when A_z is close to singular. The ability to obtain a minimum norm solution is very useful in near singular situation, whereby small perturbations due to noise or rounding error could be greatly amplified and rendered the solution useless. The B -spline coefficients $\{\alpha_{ij}^z\}$ are obtained as

$$\theta_z = V S^+ U^T b_z \quad (41)$$

where S^+ is a diagonal matrix with diagonal elements given by the reciprocal of that in S , if it is greater than a small tolerance, or zero otherwise, and U and V are column orthogonal and square orthogonal matrices, respectively.

B. Iterative Update of Spline Coefficients

Let us define the quantity $c_{p,q}^z$ as

$$c_{p,q}^z = \gamma \sum_{i=-3}^g \sum_{j=-3}^h (4\alpha_{ij}^{z-1} + 4\alpha_{ij}^{z+1} - \alpha_{ij}^{z-2} - \alpha_{ij}^{z+2}) \omega_0(p,q,i,j). \quad (42)$$

Then, (28) can be written in matrix form as

$$[A_z + 6\gamma\Omega_0] \theta_z = b_z + c_z \quad (43)$$

where Ω_0 and c_z are obtained in a similar way as Ω_k and b_z . Equation (43) forms the basis of our iterative procedure for updating the spline coefficients $\{\alpha_{ij}^z\}$, where $c_{p,q}^z$ is computed using the spline coefficients found during the previous iteration. Unlike (40), (43) explicitly constrains the stack of spline surfaces to be continuous over the z direction. We have chosen

$\gamma = 2 \times 10^4$ in (43) in our experiments and found it to work well.

Note that the bracket term and b_z in (43) do not change in the iterations. Also, Ω_0 can be precomputed, and there is no need to explicitly evaluate the spline surfaces during iterations since only the spline coefficients are involved in the update, but not the actual spline surfaces. In addition, the small size of the matrix in (43), due to the use of a small number of spline intervals, makes it very fast to compute.

The two-stage algorithm allows fast computation of a smooth 3-D field. If we were to compute a true 3-D tensor spline field, the computation involved in evaluating (29) will be prohibitive, since it cannot be evaluated in a separable form in each of the x , y and z dimensions due to the weight term $\sum_{k=1}^c u_{k,x}^m$ [23]. To illustrate, we let the number of spline intervals be k and the image dimension be m in all three dimensions. A rough calculation indicates that each dimension would require $op = (4 \text{ km})/3$ unit of computer operations.³ For 2-D, the number of operations would be $(op)^2$, whereas for 3-D, it would be $(op)^3$. If we let $k = 3$ and $m = 180$, then $op = 720$, and the number of operations in 3-D is therefore 720 times that of 2-D. For a stack of 2-D spline surfaces of size $217 \times 181 \times 181$, typical computer time for evaluating (29) on a Pentium-4 2 GHz PC is around 10 s. This would increase to > 2 h for the true 3-D spline case, which would make the algorithm impractical for 3-D applications. In contrast, the novel two-stage algorithm takes an overall time of around 15 s for computing the spline coefficients, with the number of iterations on (43) set to 500.

With the B -spline coefficients obtained from (43), the B -spline surface $w_z(x, y)$ can be obtained by evaluating (14) at every location (x, y) . The INU compensated MR image can be obtained by dividing $s(\underline{x})$ by the exponential of the log bias field $w_z(x, y)$.

The 3-D local spatial neighborhood that we used in this work is a 3-D six-point neighborhood and is given by the N_4 neighborhood on the plane, i.e., north, east, south, west of the center voxel, plus the voxels immediately before and after the center voxel.

During updating of the bias field, the membership values and cluster centroids of the data are fixed. When the membership values and cluster centroids are still changing rapidly between FCM iterations, the bias field cannot be updated in a stable manner. Therefore, we update the bias field when the L_∞ change in membership value between two iterations is less than $1/(2c)$, where c is the number of clusters in the data. The bias field is also held fixed at least once between two successive iterations to allow the updated results to propagate sufficiently to the membership update and centroid update steps.

We initialize the ASFCM by specifying the initial locations of the cluster centroids. Like the conventional FCM, the ASFCM iterates to the final solution by a local optimization of the objective function (17). Hence, proper selection of the initial cluster centroids will generally improve accuracy and convergence. Whenever the values of the true tissue class intensities are approximately known, they can be used as initial

cluster centroids. When such knowledge is not available, the initial centroids can be estimated as follows. For scalar data, we compute a smooth histogram of the data and use the c most prominent peaks as the c cluster centroids. For vector-valued data, we compute the smooth histogram for each dimension, and record the location of the prominent peaks that are above a certain threshold. The intersections of these locations specify the possible concentration of data in the multi-dimensional space. The data density within a local region around these intersections is computed, and the c intersections with the highest data density are chosen as the c initial cluster centroids.

The procedure for carrying out adaptive spatial FCM segmentation of MR images can now be stated as follows.

Adaptive Spatial FCM Segmentation

- 1) Set the number of clusters c . Set $m = 2$. Choose a value for the spline smoothness weighting coefficient β . Set the number of splines knots in the x and y dimensions. Set the maximum number of iterations $ITMAX$. Initialize the log bias field $\{w_z(x, y)\}$ to zero.
- 2) Obtain initial estimates of the cluster centroids v_k as outlined above.
- 3) Compute $\lambda_{\underline{x}, y}$ using (8)–(10).
- 4) Compute the initial membership for every voxel using (22).
- 5) Compute the regularizing matrices $\Omega = \Omega_1 + 2\Omega_2 + \Omega_3$ and Ω_0 using (32)–(35).
- 6) Repeat for $t = 1$ to $ITMAX$ or until maximum change in membership value is less than a small threshold of 0.005:
 - i) When $t > 1$ AND the maximum change in membership value is less than $1/(2c)$ AND $\{w_z(x, y)\}$ is not updated during the last iteration, updates the $\{w_z(x, y)\}$ by solving, for every slice of B -spline surface in the z direction, the B -spline coefficients $\{\alpha_{ij}^z\}$ using the two-stage algorithm.
 - ii) Update the fuzzy cluster centroids using (23).
 - iii) Update the membership values using (22).
- 7) Perform a final hard classification by assigning the data to the cluster with the highest membership value.

V. MR BRAIN IMAGE SEGMENTATION

The algorithm is implemented in C and tested on both simulated 3-D MR images obtained from the BrainWeb Simulated Brain Database at the McConnell Brain Imaging Centre of the Montreal Neurological Institute (MNI), McGill University [25]–[28], and on real MRI data. Simulated brain data of varying noise and INU levels are used to perform quantitative assessment of the proposed algorithm since ground truths are known for these data. In these data sets, the INU artifact was

³We do not differentiate the type of computer operation here. One operation could involve several logical and arithmetic operations. Our purpose here is to illustrate the additional computer time in going from 2-D to 3-D.

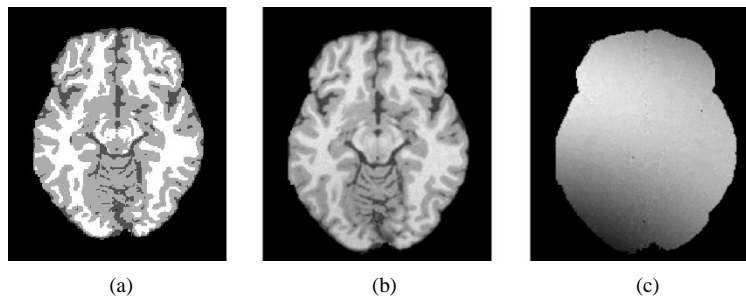


Fig. 1. A slice of the simulated 3-D brain image from MNI ($z = 60$). (a) True model. (b) Image corrupted with noise and INU artifact. (c) The corresponding bias field.

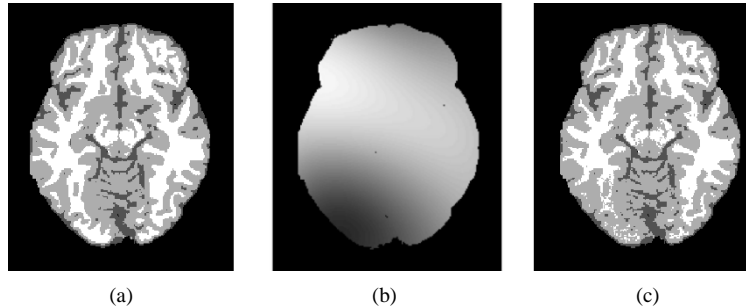


Fig. 2. Segmentation result for the MRI image of Fig. 1(b). (a) The segmentation using the proposed algorithm. (b) The recovered bias field. (c) The segmentation using the conventional FCM algorithm.

produced by multiplying the simulated MR image by a bias field recovered from an actual MR scan according to the image formation model of (1).⁴ Extra-cranial tissues were removed from all images prior to segmentation. For real data, this can be done using any of the techniques reported in the literature [32]–[36]. Fig. 1 shows a slice of the simulated MRI brain data, taken at $z = 60$. The brain image of Fig. 1(a) was generated based on a discrete anatomical normal brain model, and serves as the true model. The image of Fig. 1(b) was simulated from the true model with the following settings: T_1 modality, ICBM protocol [29], slice thickness of 1 mm (1 mm^3 voxels), 3% noise level and 40% INU. Fig. 1(c) shows the actual bias field that produces the INU artifact. It was obtained by solving for (1), using the noise-free, INU artifact-free data and the noise-free, INU affected data, obtained also from MNI.

The number of tissue classes in the segmentation was set to three, which corresponds to gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF). Background pixels are ignored in the computation. For all the segmentation experiments, the default parameter values used are: $m = 2$, $\beta = 5000$, number of spline intervals in the x and y dimensions, $nx = ny = 2$, maximum number of iterations $ITMAX = 50$. The algorithm usually converges in around 6–7 iterations. For the simulated 3-D MRI brain image of dimension $217 \times 181 \times 181$ [row (y) \times column (x) \times depth (z)], the total computation time is around 1.5–2 min on a Pentium 4 2-GHz PC.

A. Visual Evaluation

Fig. 2(a) shows the segmented image. The segmentation can be observed to correspond well to the true model in Fig. 1(a).

⁴This was done by MNI. However, we have verified experimentally, through the computation of the actual bias field shown Fig. 1(c), that they actually used the multiplicative model of (1) in simulating the INU artifact.

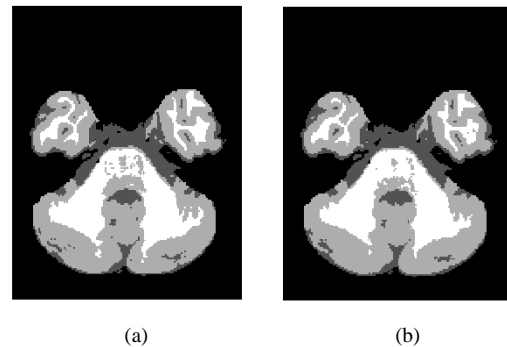


Fig. 3. Segmentation result for the slice near the base of the brain ($z = 35$). (a) True model. (b) Segmented result.

Fig. 2(b) shows the recovered bias field, which resembles very closely the actual bias field of Fig. 1(c). In comparison, we also show in Fig. 2(c) the segmentation by the conventional FCM algorithm, whose accuracy is severely affected by noise and INU. The results clearly indicate that the proposed algorithm is able to compensate for noise and INU artifact in the input image. Figs. 3 and 4 show the ground truth and the segmentation result for slices taken at $z = 35$ (near the base of the brain) and $z = 140$ (near the top of the skull), respectively. Although they are more difficult to segment than slices from the center of the brain, the results show that accurate segmentation can still be achieved.

Fig. 5(a) shows an across-slice view of the actual bias field, taken at $y = 110$, for the same data set. Fig. 5(b) shows the estimated bias field taken at the same location. As can be seen, the estimated bias field has captured accurately the intensity inhomogeneity across slices without exhibiting between-slice discontinuity in spite of the modeling of the 3-D bias field by a stack of 2-D spline surfaces. The second regularizing term of

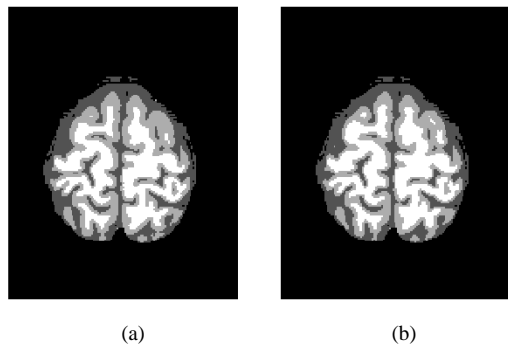


Fig. 4. Segmentation result for the slice near the top of the skull ($z = 140$). (a) True model. (b) Segmented result.

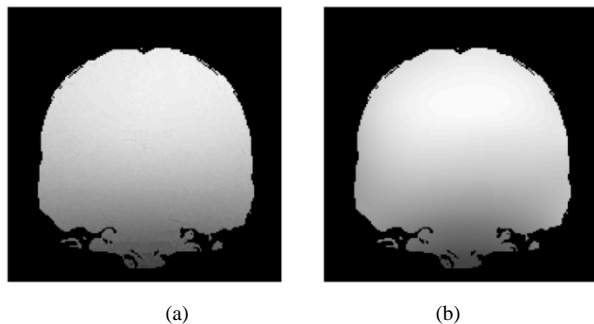


Fig. 5. (a) Actual, and (b) computed bias field for $y = 110$. The x coordinate increases from left to right and the z coordinate increases from bottom to top.

(20) has successfully constrained the estimated 3-D field to be smooth in the z direction. To show the effectiveness of the bias correction method, Fig. 6 shows the mean intensity value of the WM from the base of the brain to the top of the brain. The uncorrected data with 40% INU (dashed line) shows significant variation in intensity value whereas the INU corrected data (solid line) has a more uniform WM intensity value. For comparison, the WM mean intensity value for the data with no INU (dotted line) is also shown. Besides a constant offset, the remarkable similarity in the shape of the two curves indicates that INU has been correctly compensated for. Note that the mean intensity value for the data with no INU is not a constant straight line. This is due largely to partial volume effect, where a voxel is partially shared by two or more tissue types. This phenomenon is particularly noticeable at the two extremes of the brain, where boundaries between tissue types become less defined. The proposed algorithm is also able to correctly take that into account as reflected in the closely matched shape around the two ends of the curve.

B. Quantitative Evaluation

For a quantitative evaluation of the performance of the algorithm, we compute the misclassification rate (MCR) for the segmentation of the simulated MRI data (T_1 weighted, 1 mm^3 voxels, 3% noise) with varying level of INU inhomogeneity (i.e., 0% INU, 20% INU, 40% INU). The MCR is defined as the number of pixels misclassified by the algorithm divided by the total number of pixels in the image. For comparison, we quote the MCR of the different INU estimation algorithms as reported in [15]. The results are presented in Table I. FCM denotes the

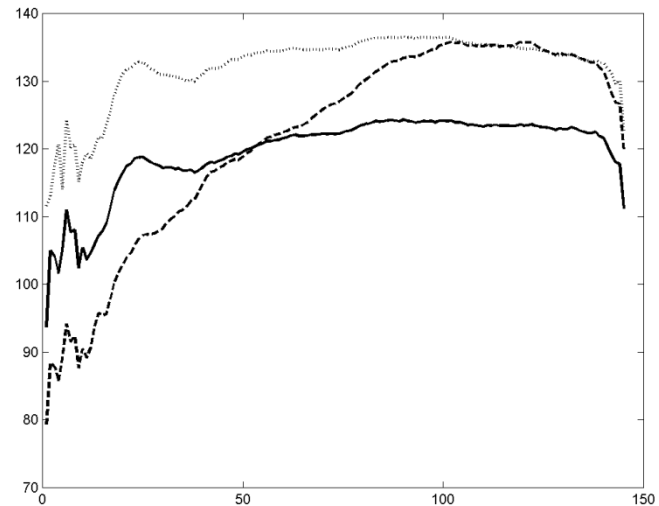


Fig. 6. Mean intensity value of WM as a function of z -coordinate, from the base of the brain to the top of the brain. The dashed line is for the uncorrected data (INU = 40%); the solid line is for the INU corrected data. The dotted line is for the data with no INU.

TABLE I
MISCLASSIFICATION RATE FOR DIFFERENT INU CORRECTION AND SEGMENTATION METHODS AT DIFFERENT INU LEVEL

Method	0% INU	20% INU	40% INU
FCM	3.988 %	5.450 %	9.016 %
FM-AFCM	4.171 %	4.322 %	5.065 %
TM-AFCM	4.168 %	4.322 %	4.938 %
EM1	6.344 %	7.591 %	13.768 %
EM2	4.242 %	5.638 %	9.604 %
AMRF	3.876 %	4.795 %	6.874 %
MNI-FCM	4.979 %	4.970 %	5.625 %
Proposed	3.717 %	3.676 %	3.832 %

conventional FCM algorithm. FM -AFCM and TM-AFCM denote the full multigrid adaptive FCM algorithm and the truncated multigrid adaptive FCM algorithm, respectively [15]. EM1 and EM2 denote the unsupervised EM algorithm for finite Gaussian mixture models, where EM1 refers to the standard model and EM2 refers to the model where variances and mixture coefficients of the Gaussian components are assumed equal [30]. AMRF denotes the adaptive Markov random field algorithm [31], [32]. MNI-FCM denotes the method where the $N3$ inhomogeneity correction technique [9] from MNI is applied first, followed by FCM segmentation.

From Table I, we see that the MCR generally increases with an increase in the INU level. Also, our method has significantly better performance than other methods and is more robust to increased inhomogeneity. For the 40% INU level, our method shows an improvement of 58% over the FCM method, 24% over the FM -AFCM method, 22% over the TM-AFCM method, 72% over the EM1 method, 60% over the EM2 method, 44% over the AMRF method and 32% over the MNI-FCM method.

We also compare our results with the results obtained by the EM-MRF algorithm of Leemput *et al.* [39], [40].⁵ In the EM-MRF algorithm, the MRI brain tissue classes are modeled

⁵Downloaded from <http://bilbo.esat.kuleuven.ac.be/web-pages/downloads/ems/ems.html>

TABLE II
COMPARISON BETWEEN OUR ALGORITHM AND THE EM-MRF ALGORITHM: IN TERMS OF OVERLAP METRIC (IN %) FOR DIFFERENT TISSUE CLASS, AND THE MCR (IN %), FOR DIFFERENT INU LEVEL. "NO MRF" DENOTES THE EM-MRF ALGORITHM WITHOUT MRF REGULARIZATION, "WITH MRF" DENOTES THE EM-MRF ALGORITHM WITH MRF REGULARIZATION

	0% INU			20 % INU			40% INU		
	our algorithm	no MRF	with MRF	our algorithm	no MRF	with MRF	our algorithm	no MRF	with MRF
WM	96.44	93.70	93.31	96.62	93.52	93.53	96.59	93.49	93.13
GM	95.87	94.03	92.12	96.02	93.95	91.81	95.97	93.89	91.24
CSF	96.68	86.67	85.03	96.67	87.17	85.37	96.61	86.66	85.33
GM+WM	99.22	98.98	97.97	99.22	99.01	97.73	99.20	98.98	97.52
MCR (%)	3.717	9.294	11.810	3.676	9.224	12.002	3.832	9.459	12.557

as finite Gaussian mixtures with Markov random field regularization and digital brain atlas initialization, and the bias field is modeled as a fourth order least square polynomial fit. For the EM-MRF method, we calculated both the MCR and the overlap metric [41]. The overlap metric is define as $2V_{ae}^k / (V_a^k + V_e^k)$, where V_{ae}^k denotes the number of voxels assigned to class k by both the ground truth and the algorithm, V_a^k and V_e^k denote the number of voxels assigned to class k by the algorithm and the ground truth, respectively, for the three tissue classes. The overlap metric attains the value of one if both segmentations are in full agreement and zero if there is no overlap at all. We run both our algorithm and the EM-MRF algorithm on the three simulated MRI datasets. The comparison results are tabulated in Table II. For the EM-MRF algorithm, we used two different settings: one with MRF regularization and the other one without MRF regularization. The results show that our algorithm has performed well over all three tissue types compared to the EM-MRF algorithm, even though the EM-MRF algorithm uses a prior classification derived from a digital brain atlas that contains spatially varying prior probability maps for the location of CSF, GM, and WM.

It is also remarkable that the performance of our method, even at 40% INU, is still better than the FCM method at 0% INU. Our method has a slightly inferior performance at 0% INU than at 20% INU. This is due to the additional degree of freedom associated with the bias field. This effect is also observed in the MNI-FCM method. One can easily reduce the error by increasing β if the amount of inhomogeneity is known to be low. It is interesting to note that at 0% INU, our method, with the additional degree of freedom, still performs better than the FCM method. This is due to the spatial continuity constraint in our algorithm, which smoothes out noise and reduces classification ambiguity. To see the advantage of incorporating the spatial continuity constraint, we set the bias field in our algorithm to zero and carried out the segmentation on 0% INU data with just the spatial continuity constraint. The MCR for this case is 3.474%, an improvement of over 13% over the FCM method. If we just have the bias field correction but not the spatial continuity constraint, we get an MCR of 4.003% at 0% INU, which is inferior to the FCM method, as expected. The difference, however, is small, indicating that our method performs well on images of varying inhomogeneity, without the need for tedious adjustment of the regularizing parameters. The advantage of having a noise smoothing property is also observed in the AMRF method at 0% INU, which has a slightly better performance than the FCM method.

TABLE III
MISCLASSIFICATION RATE FOR THE SAME DATASET REFORMATTED AT DIFFERENT SLICE PLANES

	0% INU	20% INU	40% INU
Transverse slices	3.717 %	3.676 %	3.832 %
Coronal slices	3.825 %	3.823 %	4.065 %
Sagittal slices	3.815 %	3.802 %	3.879 %

TABLE IV
MISCLASSIFICATION RATE AS A FUNCTION OF REGULARIZATION PARAMETER β AND THE NUMBER OF SPLINE INTERVALS nx WITH $\gamma = 2 \times 10^4$

$\beta \backslash nx$	1	2	3	4	5
3×10^3	3.779 %	3.834 %	3.862 %	3.945 %	3.917 %
4×10^3	3.780 %	3.834 %	3.941 %	3.928 %	3.923 %
5×10^3	3.780 %	3.832 %	3.852 %	3.916 %	3.941 %
6×10^3	3.778 %	3.837 %	3.847 %	3.909 %	3.942 %
7×10^3	3.777 %	3.844 %	3.839 %	3.903 %	3.879 %

TABLE V
MISCLASSIFICATION RATE AS A FUNCTION OF REGULARIZATION PARAMETERS β AND γ WITH $nx = 2$

$\beta \backslash \gamma$	1×10^4	1.5×10^4	2×10^4	2.5×10^4	3×10^4
3×10^3	3.835 %	3.834 %	3.834 %	3.834 %	3.835 %
4×10^3	3.835 %	3.834 %	3.834 %	3.834 %	3.834 %
5×10^3	3.833 %	3.833 %	3.832 %	3.832 %	3.833 %
6×10^3	3.837 %	3.837 %	3.837 %	3.837 %	3.837 %
7×10^3	3.845 %	3.844 %	3.844 %	3.845 %	3.845 %

As a mean of quantitatively validating the two-stage approach for 3-D bias field estimation for slices taken at different planes, we computed the MCR for the same dataset (which is originally in transverse slices) reformatted as coronal slices and sagittal slices. The results are presented in Table III. The results show that the two-stage approach for estimating the 3-D bias field is insensitive to the slice plane. We remark that if the bias field estimation is required to be totally invariant to the slice plane, one possible approach is to compute a bias field for each slice plane, and then average the three bias fields to get the slice-plane invariant bias field in each iteration.

C. Variation of Algorithmic Parameters

Table IV shows the MCR after applying the proposed ASFCM algorithm to the simulated data set (1 mm^3 voxels, 3% noise, and 40% INU) when varying the regularizing

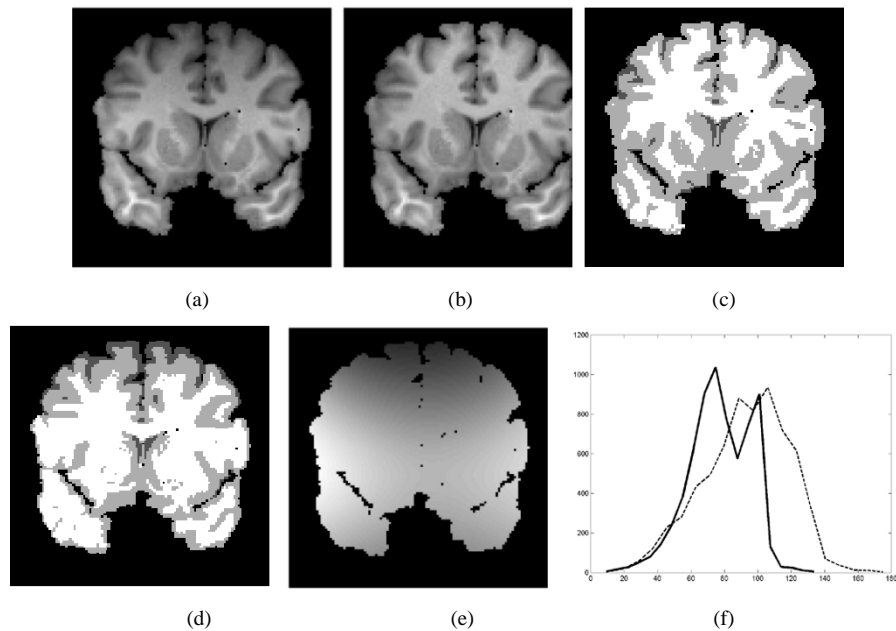


Fig. 7. Segmentation of real MRI image. (a) Original image. (b) INU corrected image. (c) Proposed ASFCM segmentation. (d) FCM segmentation. (e) Estimated bias field. (f) Intensity histogram before (dashed line) and after (solid line) INU correction.

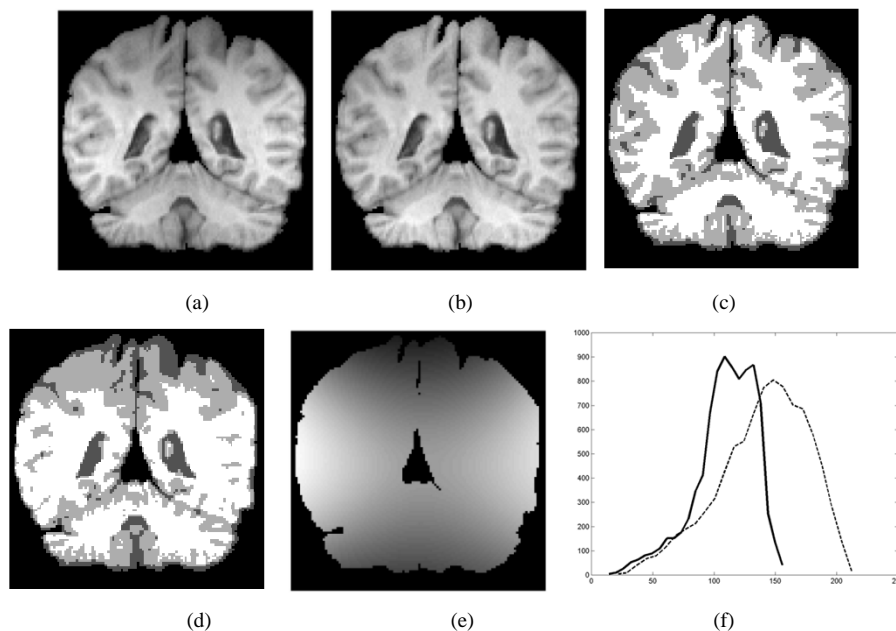


Fig. 8. Segmentation of another real MRI image. (a) Original image. (b) INU corrected image. (c) Proposed ASFCM segmentation. (d) FCM segmentation. (e) Estimated bias field. (f) Intensity histogram before (dashed line) and after (solid line) INU correction.

parameter β and the number of spline intervals, while setting $\gamma = 2 \times 10^4$. As can be seen, the algorithm is insensitive to both parameters. The bias field can adequately be modeled by B -spline surfaces with very few intervals since the bias field is known *a priori* to be smooth. We observe that fewer spline intervals generally give better results. This can be explained by the fewer degree-of-freedom, which implicitly imposes a strong constraint on the smoothness of the spline surface. Nevertheless, if the bias field were known to have large spatial variation, more spline intervals would be needed to ensure an adequate fit. Even so, spline intervals = 2 ~ 5 would suffice in most cases. Table V shows the MCR when varying β and γ , while setting the number of spline intervals to two. Again,

the proposed algorithm is insensitive to the parameter settings. Finally, we observe that the different parameter combinations in Tables IV and V all have an MCR of less than the seven other methods we compare in Table I (see last column in Table I), and even the FCM at 0% INU, indicating the robustness and efficacy of our algorithm.

D. Performance on Actual MRI Data

Figs. 7 and 8 show one slice of the segmentation results for two real T_1 -weighted MR images using the proposed algorithm. Figs. 7(a) and 8(a) are the original images. Figs. 7(b) and 8(b) are the INU corrected images. The INU artifact has largely been suppressed and one can see a fairly uniform image compared

to the original image. Figs. 7(c) and 8(c) show the segmentations using our method. Figs. 7(d) and 8(d) show the segmentations using the FCM algorithm. Visual inspection shows that our method produces better segmentations than the FCM algorithm. In particular, the GM and WM in the top region of both images can be better separated by our method. Figs. 7(e) and 8(e) show the estimated bias fields. The bias field has successfully captured the different shading in the original images. Figs. 7(f) and 8(f) show the intensity histogram of the slices before and after INU correction. One can see that there is less spread in the histogram after INU correction. Also, the two prominent peaks, which correspond to the GM and WM are sharper and better resolved in the INU corrected images.

VI. CONCLUSION

We have presented an adaptive spatial FCM segmentation algorithm that takes into account a local spatial continuity constraint, as well as the suppression of the INU artifact in 3-D MR images. The algorithm employs a novel dissimilarity index that considers the local influence of neighboring pixels in an adaptive manner. If the neighborhood window is in a nonhomogeneous region, the influence of the neighboring voxels on the center voxel is suppressed; otherwise, the center voxel is smoothed by its neighboring voxels during membership and cluster centroid computation. To suppress the INU artifact, a multiplicative MR image formation model is used. The estimation of the 3-D multiplicative bias field is formulated as the estimation of a stack of 2-D smoothing spline surfaces, with continuity enforced across slices, minimizing the 3-D residual signal between the actual data and a piecewise constant FCM solution. The B -spline coefficients of the spline surfaces are obtained by a computationally efficient two-stage algorithm. Extensive experiment results and comparative studies with several existing methods on the segmentation of both simulated and real MR brain images illustrate the effectiveness and robustness of our algorithm.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers whose comments greatly improved the paper.

REFERENCES

- [1] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.
- [2] J. C. Bezdek, L. O. Hall, and L. P. Clark, "Review of MR image segmentation techniques using pattern recognition," *Med. Phys.*, vol. 20, pp. 1033–1048, 1993.
- [3] M. Singh, P. Patel, D. Khosla, and T. Kim, "Segmentation of functional MRI by K-means clustering," *IEEE Trans. Nucl. Sci.*, vol. 43, pp. 2030–2036, June 1996.
- [4] L. O. Hall, A. M. Bensaid, L. P. Clarke, R. P. Velthuizen, M. S. Silbiger, and J. C. Bezdek, "A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain," *IEEE Trans. Neural Networks*, vol. 3, pp. 672–682, Sept. 1992.
- [5] A. Simmons, P. S. Tofts, G. J. Barker, and S. R. Arridge, "Sources of intensity nonuniformity in spin echo images at 1.5 T," *Magn. Reson. Med.*, vol. 32, pp. 121–128, 1994.
- [6] L. G. Nyul, J. K. Udupa, and X. Zhang, "New variants of a method of MRI scale standardization," *IEEE Trans. Med. Imag.*, vol. 19, pp. 143–150, Feb 2000.
- [7] C. R. Meyer, P. H. Bland, and J. Pipe, "Retrospective correction of intensity inhomogeneities in MRI," *IEEE Trans. Med. Imag.*, vol. 12, pp. 36–41, Mar. 1995.
- [8] B. M. Dawant, A. P. Zijdenbos, and R. A. Margolin, "Correction of intensity variations in MR images for computer-aided tissue classification," *IEEE Trans. Med. Imag.*, vol. 12, pp. 770–781, Dec 1993.
- [9] J. G. Sled, A. P. Zijdenbos, and A. C. Evans, "A nonparametric method for automatic correction of intensity nonuniformity in MRI data," *IEEE Trans. Med. Imag.*, vol. 17, pp. 87–97, Feb 1998.
- [10] W. M. Wells III, W. E. L. Grimson, R. Kikinis, and F. A. Jolesz, "Adaptive segmentation of MRI data," *IEEE Trans. Med. Imag.*, vol. 15, pp. 429–442, Aug 1996.
- [11] M. Tincher, C. R. Meyer, R. Gupta, and D. M. Williams, "Polynomial modeling and reduction of RF body coil spatial inhomogeneity in MRI," *IEEE Trans. Med. Imag.*, vol. 12, pp. 361–365, June 1993.
- [12] B. H. Brinkmann, A. Manduca, and R. A. Robb, "Optimized homomorphic unsharp masking for MR grayscale inhomogeneity correction," *IEEE Trans. Med. Imag.*, vol. 17, pp. 161–171, Apr. 1998.
- [13] B. Johnston, M. S. Aitkins, B. Mackiewicz, and M. Anderson, "Segmentation of multiple sclerosis lesions in intensity corrected multispectral MRI," *IEEE Trans. Med. Imag.*, vol. 15, pp. 154–169, Apr. 1996.
- [14] M. Styner, C. Brechbuhler, G. Szekely, and G. Gerig, "Parametric estimate of intensity inhomogeneities applied to MRI," *IEEE Trans. Med. Imag.*, vol. 9, pp. 153–165, Mar. 2000.
- [15] D. L. Pham and J. L. Prince, "Adaptive fuzzy segmentation of magnetic resonance images," *IEEE Trans. Med. Imag.*, vol. 18, pp. 737–751, Sept. 1999.
- [16] J. O. Eklundh, H. Yamamoto, and A. Rosenfeld, "A relaxation method for multi-spectral pixel classification," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, pp. 72–75, 1980.
- [17] J. Y. Hsiao and A. Sawchuk, "Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-11, pp. 1279–1292, 1989.
- [18] —, "Unsupervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," *Comput. Vis., Graph., Image Processing*, vol. 48, pp. 1–21, 1989.
- [19] Y. A. Tolia and S. M. Panas, "Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 359–369, May 1998.
- [20] A. W. C. Liew, S. H. Leung, and W. H. Lau, "Fuzzy image clustering incorporating spatial continuity," *Inst. Elect. Eng. Proc.-Vision, Image and Signal Processing*, vol. 147, pp. 185–192, Apr. 2000.
- [21] J. K. Udupa and S. Samarasekera, "Fuzzy connectedness and object definition: Theory, algorithms and applications in image segmentation," *Graph. Models Image Process.*, vol. 58, pp. 246–261, 1996.
- [22] J. K. Udupa, L. Wei, S. Samarasekera, Y. Miki, M. A. van Buchem, and R. I. Grossman, "Multiple sclerosis lesion quantification using fuzzy connectedness principles," *IEEE Trans. Med. Imag.*, vol. 16, pp. 598–609, Oct. 1997.
- [23] P. Dierckx, *Curve and Surface Fitting with Splines*. Oxford, U.K.: Oxford Univ., 1993, Monographs on Numerical Analysis.
- [24] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992.
- [25] McGill Univ., Quebec, ON, Canada.. [Online]. Available: <http://www.bic.mni.mcgill.ca/brainweb>
- [26] C. A. Cocosco, V. Kollokian, R. K. S. Kwan, and A. C. Evans, "BrainWeb: Online interface to a 3D MRI simulated brain database," *NeuroImage*, pt. 2/4, S425, vol. 5, no. 4, 1997.
- [27] D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes, and A. C. Evans, "Design and construction of a realistic digital brain phantom," *IEEE Trans. Med. Imag.*, vol. 17, pp. 463–468, June 1998.
- [28] R. K. S. Kwan, A. C. Evans, and G. B. Pike, "An extensible MRI simulator for post-processing evaluation," in *Lecture Notes in Computer Science*. Berlin, Germany: Springer-Verlag, 1996, vol. 1131, Proceedings Visualization in Biomedical Computing (VBC'96), pp. 135–140.
- [29] Univ. California, Int. Consortium Brain Mapping (ICBM), Lab. Neuro Imag., Quebec, ON, Canada., Los Angeles, CA. [Online]. Available: <http://www.loni.ucla.edu/ICBM/>
- [30] Z. Liang, R. J. Jaszczak, and R. E. Coleman, "Parameter estimation of finite mixtures using the EM algorithm and information criteria with application to medical image processing," *IEEE Trans. Nucl. Sci.*, vol. 39, pp. 1126–1133, Aug. 1992.

- [31] M. X. H. Yan and J. S. Karp, "An adaptive Bayesian approach to three-dimensional MR brain segmentation," in *Proc. Information Processing and Medical Imaging (IPMI'95)*, 1995, pp. 201–213.
- [32] A. F. Goldszal, C. Davatzikos, D. L. Pham, M. X. H. Yan, R. N. Bryan, and S. M. Resnick, "An image processing system for qualitative and quantitative volumetric analysis of brain images," *J. Comput. Assist. Tomogr.*, vol. 22, pp. 827–837, 1998.
- [33] B. Mackiewicz, "Intracranial boundary detection and radio frequency correction in magnetic resonance images," Master's, Simon Fraser Univ., Comput. Sci. Dept., Burnaby, BC, Canada, 1995.
- [34] S. Sandor and R. Leahy, "Toward automated labeling of the cerebral cortex using a deformable atlas," in *Proc. Information Processing and Medical Imaging (IPMI'95)*, 1995, pp. 127–138.
- [35] T. Kapur, W. E. L. Grimson, W. M. Wells III, and R. Kikinis, "Segmentation of brain tissue from magnetic resonance images," *Med. Image Anal.*, vol. 1, pp. 109–127, 1996.
- [36] F. Kruggel and G. Lohman, "Automatic adaptation of the stereotactical coordinate system in brain MRI datasets," in *Proc. Information Processing and Medical Imaging (IPMI'97)*, 1997, pp. 471–476.
- [37] H. E. Cline, W. E. Lorensen, R. Kikinis, and F. Jolesz, "Three dimensional segmentation of MR images of the head using probability and connectivity," *J. Comput. Assist. Tomogr.*, vol. 14, pp. 1037–1045, 1990.
- [38] M. E. Brummer, R. M. Mersereau, R. L. Eisner, and R. R. J. Lewine, "Automatic detection of brain contours in MRI data sets," *IEEE Trans. Med. Imag.*, vol. 12, pp. 153–166, June 1993.
- [39] K. V. Leemput, F. Maes, D. Vandermeulen, and P. Suetens, "Automated model-based bias field correction of MR images of the brain," *IEEE Trans. Med. Imag.*, vol. 18, pp. 885–896, Oct. 1999.
- [40] —, "Automated model-based tissue classification of MR images of the brain," *IEEE Trans. Med. Imag.*, vol. 18, pp. 897–908, Oct. 1999.
- [41] A. 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, pp. 716–724, Dec. 1994.