Adaptive Fuzzy Segmentation of Magnetic Resonance Images

Dzung L. Pham, *Student Member, IEEE*, and Jerry L. Prince,* *Member, IEEE*

Abstract—An algorithm is presented for the fuzzy segmentation of two-dimensional (2-D) and three-dimensional (3-D) multispectral magnetic resonance (MR) images that have been corrupted by intensity inhomogeneities, also known as shading artifacts. The algorithm is an extension of the 2-D adaptive fuzzy C-means algorithm (2-D AFCM) presented in previous work by the authors. This algorithm models the intensity inhomogeneities as a gain field that causes image intensities to smoothly and slowly vary through the image space. It iteratively adapts to the intensity inhomogeneities and is completely automated. In this paper, we fully generalize 2-D AFCM to three-dimensional (3-D) multispectral images. Because of the potential size of 3-D image data, we also describe a new faster multigrid-based algorithm for its implementation. We show, using simulated MR data, that 3-D AFCM yields lower error rates than both the standard fuzzy C-means (FCM) algorithm and two other competing methods, when segmenting corrupted images. Its efficacy is further demonstrated using real 3-D scalar and multispectral MR brain images.

Index Terms— Clustering methods, fuzzy sets, image segmentation, magnetic resonance imaging.

I. INTRODUCTION

TISSUE classification is a necessary step in many medical imaging applications, including the quantification of tissue volumes, the detection of pathology, and computer integrated surgery. Classification of voxels exclusively into distinct classes, however, is difficult because of artifacts such as noise and partial volume averaging, where multiple tissues are present in a single voxel. To compensate for these artifacts, there has recently been growing interest in soft segmentation methods [1]–[3]. In soft segmentations, voxels may be classified into multiple classes with a varying degree of membership. The membership thus gives an indication of where noise and partial volume averaging have occurred in the image. Standard soft segmentation algorithms, however, cannot effectively compensate for intensity inhomogeneities, a common artifact in magnetic resonance (MR) images.

In MR images, intensity inhomogeneities are typically caused by nonuniformities in the RF field during acquisition, although other factors also play a role [4], [5]. Similar artifacts also occur in computed tomography images, due to beam hardening effects, as well as in microscopy and light photography, due to nonuniform illumination. The result is a shading effect where the pixel or voxel intensities of the same tissue class vary over the image domain. It has been shown that the shading in MR images is well modeled by the product of the original image and a smooth slowly varying gain field [6], [7]. Typically, corrupted images are segmented using either a two-step approach or a segmentation algorithm that simultaneously classifies the voxels while compensating for the shading effect.

In the two-step approach, the image is first corrected to remove intensity inhomogeneities. This correction is then followed by a standard segmentation algorithm that assumes no inhomogeneity is present. Numerous methods have been proposed in the literature to perform the correction step. Several research groups have used homomorphic filtering in an attempt to remove the multiplicative effect of the inhomogeneity [8]–[10]. It was shown in [10], however, that homomorphic filtering can sometimes distort an image rather than correct it. In their work, Dawant *et al.* [6] used manually selected reference points in the image to guide the construction of a spline correction surface. Meyer *et al.* [11] used an edge-based segmentation scheme to find uniform regions in the image, followed by a polynomial surface fit to those regions. Sled *et al.* [12] corrected the inhomogeneity by estimating a gain field that sharpens the histogram of the image. The latter three methods used either disconnected regions in the original image or a subsampled image to estimate the gain field, thereby not taking advantage of all available image data. In [13], Lee and Vannier corrected T_1-weighted scans using an extension of the fuzzy C-means (FCM) algorithm. Images were segmented into two classes, consisting of background and nonbackground and the mean of the nonbackground class was allowed to vary, using filtered estimates computed within a local neighborhood.

Several methods have also been proposed that simultaneously compensate for the shading effect while segmenting the image. These methods have the advantage of being able to use intermediate information from the segmentation while performing the correction. Markov random field-based algorithms have been proposed that account for inhomogeneities by allowing the centroids of each class to vary independently [14]–[16]. Unser [17] proposed an adaptive K-means algorithm that also allowed the centroids to vary independently, according

to a first-order regularization term. Only hard segmentations, however, were obtained with these methods.

Wells *et al.* [7] proposed an expectation-maximization algorithm that modeled the inhomogeneities as a bias field of the image logarithm. This method was later extended in [18]. The Wells *et al.* method iterates between a classification step and a step to estimate the inhomogeneity, using an approximate spatially invariant, low-pass filter. Because training data obtained through manual interaction is required to model the distributions of the tissue intensities, it is a supervised method [1]. Their method is capable of obtaining soft segmentations based on posterior probabilities.

In a recent letter [19], we presented some initial results on an unsupervised segmentation algorithm called the two-dimensional (2-D) adaptive fuzzy C-means algorithm (2-D AFCM), designed for segmenting 2-D scalar images corrupted by intensity inhomogeneities. Based on FCM [20], [21], the advantages of 2-D AFCM are that it automatically produces soft segmentations, it is robust to inhomogeneities, and it computes a smooth gain field based on all pixels in the image. Although this algorithm is suitable for the segmentation of MR images obtained using single or multislice acquisitions, it cannot be used in volumetric acquisitions where the inhomogeneities are three-dimensional (3-D) in nature or on multispectral data. In this paper, we generalize AFCM to 3-D multispectral images. Our generalization also allows for the adjustment of the hardness or fuzziness of the resulting segmentation and for the segmentation of data with ellipsoidal-shaped clusters. A novel algorithm is presented for computing the gain field, which typically yields a threefold improvement in speed over a standard multigrid approach, without reducing accuracy. This speed improvement is especially significant when working with large 3-D data sets.

It was shown in [19] that 2-D AFCM segments images corrupted by inhomogeneities as accurately as FCM segments uncorrupted images. The accuracy of the FCM segmentation itself, however, was not quantified. We provide in this paper several new results using simulated data that show that the segmentations obtained using FCM on uncorrupted images and AFCM on corrupted images are indeed accurate in terms of classification and modeling of partial volume effects. Moreover, we show that, under default initializations, AFCM's performance on corrupted 3-D images is superior to the performance of the methods presented in [12] and [15].

II. BACKGROUND

In this section, we give a brief overview of FCM and 2-D AFCM. FCM has been used with some success in the soft or fuzzy segmentation of MR images [22]–[24], as well as for the estimation of partial volumes [3]. It clusters data by computing a measure of membership, called the fuzzy membership, at each voxel for a specified number of classes. The fuzzy membership function, constrained to be between zero and one, reflects the degree of similarity between the data value at that location and the prototypical data value or centroid of its class. Thus, a high membership value near unity signifies that the data value at that location is close to the centroid for that particular class.

FCM is formulated as the minimization of the following objective function with respect to the membership functions u and the centroids $\mathbf{v}$ [1], [21]:

$$J_{\text{FCM}} = \sum_{j \in \Omega} \sum_{k=1}^{C} u_{jk}^q \|\mathbf{y}_j - \mathbf{v}_k\|^2. \qquad (1)$$

Here, Ω is the set of voxel locations in the image domain, q is a parameter that is constrained to be greater than one, u_{jk} is the membership value at voxel location j for class k such that $\sum_{k=1}^{C} u_{jk} = 1$, $\mathbf{y}_j$ is the observed (multispectral) image intensity at location j, and $\mathbf{v}_k$ is the centroid of class k. The total number of classes C is assumed to be known. The parameter q is a weighting exponent on each fuzzy membership and determines the amount of fuzziness of the resulting classification. For $q = 1$, J_{FCM} reduces to the classical within-group sum of the squared errors objective function and FCM becomes equivalent to the K-means or ISODATA clustering algorithms [21]. A commonly used value is $q = 2$ [22], [24], [25]. The operator $\| \cdot \|$ is any inner product norm on $\mathbb{R}^P$, where P is the number of channels or spectra in the image and $\| \cdot \| = \sqrt{\langle \cdot, \cdot \rangle}$. By specifying the appropriate norm, FCM can be applied to data that possess ellipsoidal shaped clusters, although typically the Euclidean norm is used.

The FCM objective function (1) is minimized when high membership values are assigned to voxels whose intensities are close to the centroid for its particular class and low membership values are assigned when the voxel intensity is far from the centroid. Taking the first derivatives of (1) with respect to u_{jk} and $\mathbf{v}_k$ and setting those equations to zero yields necessary conditions for (1) to be minimized. Iterating through these two necessary conditions leads to a grouped coordinate descent scheme for minimizing the objective function [20], [21]. This is the standard FCM algorithm. The resulting fuzzy segmentation can be converted to a hard or crisp segmentation by assigning each voxel solely to the class that has the highest membership value for that voxel. This is known as a *maximum membership segmentation*. The advantages of FCM are that it is unsupervised (i.e., it does not require training data) and it is robust to initial conditions when applied to data with well-separated clusters [26]. However, FCM assumes that the centroids of the image are spatially invariant, which is not true of images that have been corrupted by intensity inhomogeneities.

In order to preserve the advantages of FCM, we proposed the following objective function [19], [27] for segmenting 2-D scalar images possessing intensity inhomogeneities:

$$J_{\text{AFCM2D}} = \sum_{j \in \Omega} \sum_{k=1}^{C} u_{jk}^2 (y_j - g_j v_k)^2 + \lambda_1 \sum_{j \in \Omega} \sum_{r=1}^{2}$$
$$\cdot (D_r * g)_j^2 + \lambda_2 \sum_{j \in \Omega} \sum_{r=1}^{2} \sum_{s=1}^{2}$$
$$\cdot (D_r * D_s * g)_j^2 \qquad (2)$$

where y_j is the (scalar) pixel intensity, v_k is the centroid, g_j is an unknown gain field, D_r is a (known) finite difference operator along the rth dimension of the image. The notation $(D * g)_j$ refers to the convolution of g with the kernel D and taking the resulting value at the jth pixel. Equation (2) models the brightness variation of the inhomogeneity by multiplying the centroids by the gain field g_j. The last two terms are first- and second-order regularization terms used to ensure that g_j is spatially smooth and slowly varying. The finite difference operators act like derivatives, except that they are performed on a discrete domain. AFCM, like FCM, does not place any assumption of spatial smoothness on the membership functions u_{jk}. Note that because J_{AFCM2D} assumes scalar data, the norm operator in (1) does not come into play. In addition, values of $q \neq 2$ and 3-D images are not considered in [19].

Note the similarities between (1) and the AFCM objective function. The differences lie in the inclusion of the gain field g within the norm operator and the addition of two regularization terms on g. If we assume that the membership values u and the centroids $\mathbf{v}$ are known, then the gain field that minimizes J_{AFCM2D} is the field that makes the centroids close to the data, but is also slowly varying and smooth. Without the regularization terms, a gain field could always be found that would set the objective function to zero. If λ_1 and λ_2 are set sufficiently large, then the gain field is forced to be constant and the AFCM objective function essentially reduces to a special case of the standard FCM objective function. In [19], (2) was minimized by taking its first partial derivatives with respect to u, v, and g, and iterating through these three necessary conditions. The necessary condition on g yields a space-varying difference equation that was solved using a standard multigrid approach (see Section III-C).

III. ADAPTIVE FUZZY C-MEANS

In this section, we generalize the AFCM objective function to 3-D multispectral images and describe an algorithm for minimizing the objective function. We also describe an implementation that yields much faster results than the standard multigrid approach.

A. Objective Function

When working with multispectral MR data corrupted by intensity inhomogeneities, there are two possible assumptions one can make about the gain field: 1) the gain field is a scalar field and 2) the gain field is a vector field. The first assumption implies that the brightness variation in each component or spectrum of the acquired image is identical, while the second assumes that they can be different. In [4], it was found that for spin-echo acquisitions, the inhomogeneity was nearly identical for different pulse echo times (TE) when using a body coil, but not identical when using a head coil. This indicates that in different situations, either assumption might be appropriate. We therefore consider both cases.

If a scalar gain field is assumed, we define AFCM to be an algorithm that seeks to minimize the following objective function with respect to membership functions u, the centroids

$\mathbf{v}$, and the gain field g:

$$J_{\mathrm{AFCM}} = \sum_{j \in \Omega} \sum_{k=1}^{C} u_{jk}^q \|\mathbf{y}_j - g_j \mathbf{v}_k\|^2 + \lambda_1 \sum_{j \in \Omega} \sum_{r=1}^{R} (D_r * g)_j^2 + \lambda_2 \sum_{j \in \Omega} \sum_{r=1}^{R} \sum_{s=1}^{R} (D_r * D_s * g)_j^2. \quad (3)$$

This equation is applicable to 2-D images when $R = 2$ and to 3-D images when $R = 3$. For $R = 2$, $q = 2$, and scalar image data, (3) reduces to the 2-D AFCM objective function given in (2). If the gain field is assumed to be a vector field $\mathbf{g}_j$, then we use the following objective function:

$$J_{\mathrm{AFCMV}} = \sum_{j \in \Omega} \sum_{k=1}^{C} u_{jk}^q \|\mathbf{y}_j - \mathbf{G}_j \mathbf{v}_k\|^2 + \lambda_1 \sum_{i}^{P} \sum_{j \in \Omega} \sum_{r=1}^{R} (D_r * [\mathbf{g}]_i)_j^2 + \lambda_2 \sum_{i}^{P} \sum_{j \in \Omega} \sum_{r=1}^{R} \sum_{s=1}^{R} (D_r * D_s * [\mathbf{g}]_i)_j^2 \quad (4)$$

where $\mathbf{G}_j$ is a diagonal matrix whose entries are equal to the components of the P component vector $\mathbf{g}_j$ at each voxel j. The notation $[\mathbf{x}]_i$ means the ith component of the vector $\mathbf{x}$.

In practice, we have found in double-echo MR data that the scalar gain field assumption provides nearly identical results to the vector gain field assumption and is also faster, requiring fewer computations. Furthermore, the algorithm derived from the scalar case is notationally cleaner and therefore more easily explained. For these reasons, we focus mainly on the scalar assumption for the remainder of this paper. Equations for the vector case are similar and are provided in Appendix A.

The scalar gain field objective function J_{AFCM} in (3) can be minimized by taking the first derivatives of J_{AFCM} with respect to u_{jk}, $\mathbf{v}_k$, and g_j, setting them equal to zero, and iterating through these three necessary conditions for J_{AFCM} to be at a minimum. This yields the following algorithm (the equations for which are derived in Appendix B).

Algorithm 1—AFCM:
1) Provide initial values for the centroids $\mathbf{v}_k$, $k = 1, \cdots, C$ and set the gain field g_j equal to one for all $j \in \Omega$.
2) Compute membership functions as follows:

$$u_{jk} = \frac{\|\mathbf{y}_j - g_j \mathbf{v}_k\|^{-2/(q-1)}}{\sum_{l=1}^{C} \|\mathbf{y}_j - g_j \mathbf{v}_l\|^{-2/(q-1)}} \quad (5)$$

for all $j \in \Omega$ and $k = 1, \cdots, C$.
3) Compute new centroids as follows:

$$\mathbf{v}_k = \frac{\sum_{j \in \Omega} u_{jk}^q g_j \mathbf{y}_j}{\sum_{j \in \Omega} u_{jk}^q g_j^2}, \qquad k = 1, \cdots, C. \quad (6)$$

4) Compute a new gain field by solving the following space-varying difference equation for g_j:

$$\sum_{k=1}^{C} u_{jk}^q \langle \mathbf{y}_j \mathbf{v}_k \rangle = g_j \sum_{k=1}^{C} u_{jk}^q \langle \mathbf{v}_k, \mathbf{v}_k \rangle + \lambda_1 (H_1 * g)_j + \lambda_2 (H_2 * g)_j \quad (7)$$

where the convolution kernels H_1 and H_2 are given by

$$H_1 = \sum_{r=1}^{R} (D_r * \check{D}_r)_j \quad (8)$$

$$H_2 = \sum_{r=1}^{R} \sum_{s=1}^{R} \left((D_r * D_s) * (\check{D}_r * \check{D}_s) \right)_j \quad (9)$$

where $\check{D}$ is the mirror reflection of the finite difference operator D. H_1 and H_2 are given explicitly in Appendix C for 2-D and 3-D images.

5) If the algorithm has converged, then quit. Otherwise, go to Step 2.

We define convergence to be when the maximum change in the membership functions over all pixels between iterations is less than a given threshold value. In practice, we used a threshold value of 0.01. Methods for determining initial centroids in Step 1 are described in Section III-B. The solution to the difference equation in Step 4 is described in Section III-C.

B. Initial Centroids

AFCM requires an initial estimate of centroid values. Proper selection will generally improve accuracy and convergence of the algorithm. We propose two methods for automatically selecting initial centroids. The first method may be applied generally to all scalar data, while the second method is specific to multispectral MR images.

If the given data is scalar-valued, then one can apply the approach described in [19] and [27], where the modes of a critically smoothed kernel estimator of the image histogram are used to determine the initial centroids. The approach is essentially the same as the bump-hunting algorithm described by Silverman in [28]. Briefly, a kernel estimator of the histogram is smoothed in an iterative fashion until it possesses a number of modes equal to the desired number of classes C. These modes are then numerically computed, using first and second derivatives of the kernel estimator, and are used as initial centroids.

For multispectral data, manipulation of a multidimensional kernel estimator can be computationally prohibitive. In this case, one can obtain initial centroids by applying the approach described in [22]. This approach requires *a priori* knowledge of the approximate T_1, T_2, and proton spin density of the tissue classes being segmented. Most of these values for different tissue classes have been documented in the literature (cf. [29]). These values can then be used in an imaging equation derived for the corresponding pulse sequence (e.g., spin echo) to obtain expected intensity values. This rough initialization is normally sufficient for AFCM to yield good convergence properties.

C. Solution to the Gain Field

In Step 4 of AFCM, a new gain field is computed given the current values of the centroids and membership functions. This is the most time-consuming step in AFCM and deserves special attention in its numerical implementation. Because the difference equation (7) is space varying, the gain field cannot be found using standard frequency domain filters. The equation could be solved iteratively using the Jacobi or Gauss–Seidel schemes [30], [31], but these methods take a large number of iterations to converge. In [19] and [27] this equation was solved, using a standard multigrid algorithm at each iteration of AFCM (for a general overview of multigrid algorithms, see [17] or [31]). For 2-D images, this approach is sufficiently fast, but for large 3-D images, execution times can grow to several hours. We now describe a modified multigrid algorithm that yields significantly faster overall execution time, without loss of accuracy. Its premise is that during early iterations of AFCM, only an approximate solution to the gain field is required. Thus, a subsampled solution is used and later refined as the number of iterations increases.

To see how (7) can be solved efficiently, we first write it in matrix form. We begin by defining the functions f and w to be

$$w_j = \sum_{k=1}^{C} u_{jk}^q \langle \mathbf{v}_k, \mathbf{v}_k \rangle \quad (10)$$

$$f_j = \frac{1}{w_j} \sum_{k=1}^{C} u_{jk}^q \langle \mathbf{y}_j, \mathbf{v}_k \rangle. \quad (11)$$

Then, (7) can be written as

$$\mathbf{f_w} = \mathbf{Wm} + (\lambda_1 \mathbf{H}_1 + \lambda_2 \mathbf{H}_2)\mathbf{m} \quad (12)$$

where $\mathbf{f_w}$ and $\mathbf{m}$ have as their elements $f_j w_j$ and g_j, respectively, stacked in vector form, $\mathbf{W}$ is a diagonal matrix with w_j as its diagonal elements, and $\mathbf{H}_1$ and $\mathbf{H}_2$ are matrix versions of H_1 and H_2. Defining $\mathbf{A} = (\mathbf{W} + \lambda_1 \mathbf{H}_1 + \lambda_2 \mathbf{H}_2)$, we see that the objective is to find $\mathbf{m}$ that solves $\mathbf{Am} = \mathbf{f_w}$. If $\mathbf{A}$ is decomposed as $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$, where $\mathbf{D}$ is diagonal, $\mathbf{L}$ is lower triangular, and $\mathbf{U}$ is upper triangular, then the weighted Jacobi iteration is given by [31]

$$\mathbf{m} \leftarrow [(1 - \omega)\mathbf{I} + \omega \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})]\mathbf{m} + \omega \mathbf{D}^{-1}\mathbf{f_w} \quad (13)$$

where $\mathbf{I}$ is the identity matrix, and ω is a weighting parameter (in practice, set to 0.3 and not changed). To solve (12) efficiently, we apply Jacobi iterations in a multigrid-based scheme. Computational complexity is drastically reduced by replacing the iterations that would normally take place on a fine grid with iterations on a coarse grid.

The basis of a multigrid algorithm is the coarse grid correction scheme, where an estimate of the solution is refined by approximating the error of the estimate on a coarse grid, then updating the estimate with the error. If $\hat{\mathbf{m}}$ is an approximate solution to $\mathbf{Am} = \mathbf{f_w}$, the error is defined as $\mathbf{e} = \mathbf{m} - \hat{\mathbf{m}}$. Then the error itself satisfies

$$\mathbf{Ae} = \mathbf{f_w} - \mathbf{A}\hat{\mathbf{m}} \quad (14)$$

where $\mathbf{r} = \mathbf{f_w} - \mathbf{A}\hat{\mathbf{m}}$ is called the residual. Thus, the error $\mathbf{e}$ satisfies an equation of the same form as the equation for $\mathbf{m}$ and can be approximated by applying Jacobi iterations at a resolution lower than the original resolution of $\mathbf{m}$. The coarse grid correction scheme is summarized as follows.

Algorithm 2—Coarse Grid Correction Scheme:

1) Given an initial guess of the solution, perform ν_1 Jacobi iterations on $\mathbf{A}^l \mathbf{m}^l = \mathbf{f_w}^l$ and assign the result to $\hat{\mathbf{m}}^l$.
2) Compute a coarse grid residual $\mathbf{r}^{l+1}$ by computing $\mathbf{r}^l = \mathbf{f_w}^l - \mathbf{A}^l \hat{\mathbf{m}}^l$ followed by a REDUCE operation.
3) Solve for the coarse grid error by performing ν_1 Jacobi iterations on the error equation $\mathbf{A}^{l+1} \mathbf{e}^{l+1} = \mathbf{r}^{l+1}$.
4) Apply an EXPAND operator to the error and update the fine grid estimate $\hat{\mathbf{m}}^l \leftarrow \hat{\mathbf{m}}^l + \mathbf{e}^l$.
5) Perform ν_2 Jacobi iterations on $\mathbf{A}^l \mathbf{m}^l = \mathbf{f_w}^l$ with the updated $\hat{\mathbf{m}}^l$ as the initial estimate.

A REDUCE operator is a transformation from a high-resolution representation to a representation with half the resolution along each axis, and an EXPAND operator is a low-to-high resolution transformation. Our 3-D REDUCE operator is simply the replacement of each eight-pixel ($2 \times 2 \times 2$) neighborhood with the average of the neighborhood. Our 3-D EXPAND operator is the replication of each low-resolution pixel to an eight-pixel neighborhood. By iteratively applying these operators, a multigrid pyramid is formed, as illustrated in Fig. 1(a). The superscripts in Algorithm 2 are used to denote the pyramid level of the vector or operator. Level zero represents the original resolution of the image. The operator $\mathbf{A}$ at a pyramid level other than zero is determined simply by performing a REDUCE operation on the diagonal elements of $\mathbf{W}$. Since $\mathbf{H}_1$ and $\mathbf{H}_2$ represent spatially invariant operators, these are well-defined at any resolution.

A *V-cycle* is a recursion of the coarse grid correction upon itself to coarser resolutions. In other words, the error (14) also has its own error that can be solved using an embedded coarse grid correction and this is repeated for multiple resolutions. A full multigrid V cycle is a sequence of V cycles which is first initialized by repeatedly applying Step 2 of Algorithm 2 until the top of the multigrid pyramid is reached. It then performs a sequence of V cycles that increase in resolution levels until the bottom of the pyramid is reached. A four-level full multigrid V cycle is illustrated in Fig. 1(b). By the time the original resolution has been reached, the algorithm has typically converged to its final solution.

In [19] and [27] the gain field was computed by applying one full multigrid V cycle at each iteration of 2-D AFCM. For 3-D images, we propose a new faster method that takes advantage of the fact that during early iterations of AFCM, the estimates of the centroid and membership functions are poor and an exact solution to the gain field is not necessary. We define a truncated multigrid cycle at level L to be a full multigrid V cycle that terminates the first time the Lth pyramid level is reached. In Fig. 1(b), the termination points of a truncated multigrid cycle are shown as open circles. For a truncated multigrid cycle at level $L > 0$, the estimated gain field is an approximation of the final solution on a coarse grid but it can be computed quickly. The implementation
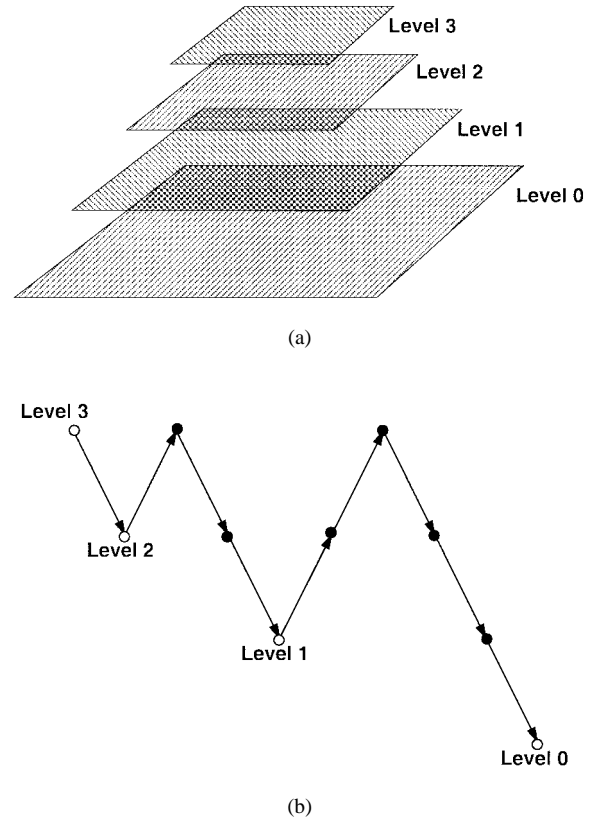

(a)

(b)

Fig. 1. Multigrid. (a) A four-level multigrid pyramid. (b) A full multigrid V cycle.

of AFCM, using a truncated full multigrid cycle, proceeds as follows.

Algorithm 3—AFCM Using Truncated Multigrid Cycle:

1) Set the size of the multigrid pyramid to some value K. Set $L = K - 2$.
2) Run the entire AFCM algorithm until convergence, using a truncated multigrid cycle at level L to solve for the gain field at each iteration.
3) If $L > 0$, decrease L by one. Using the most recent values of u, $\mathbf{v}$, and g as initial values, go to Step 2. Otherwise, if $L = 0$, terminate.

This modified multigrid algorithm greatly increases the speed of AFCM during its early iterations. As the number of iterations increases, the truncation level reduces toward the original resolution and the iterations become slower. If a result is required quickly, one can terminate Algorithm 3 at some value of $L > 0$. This provides an approximation of the final solution and, as we show in Section IV, the approximation error decreases rapidly as the resolution increases.

IV. RESULTS

AFCM was implemented in C on a Silicon Graphics O2 system with an R10000 processor running IRIX 6.3. It has been tested on both real MR data as well as simulated MR brain images obtained from the Brainweb simulated brain database at the McConnell Brain Imaging Center of the Montreal Neurological Institute, McGill University [32]. Simulated brain data sets of varying noise, inhomogeneity, and
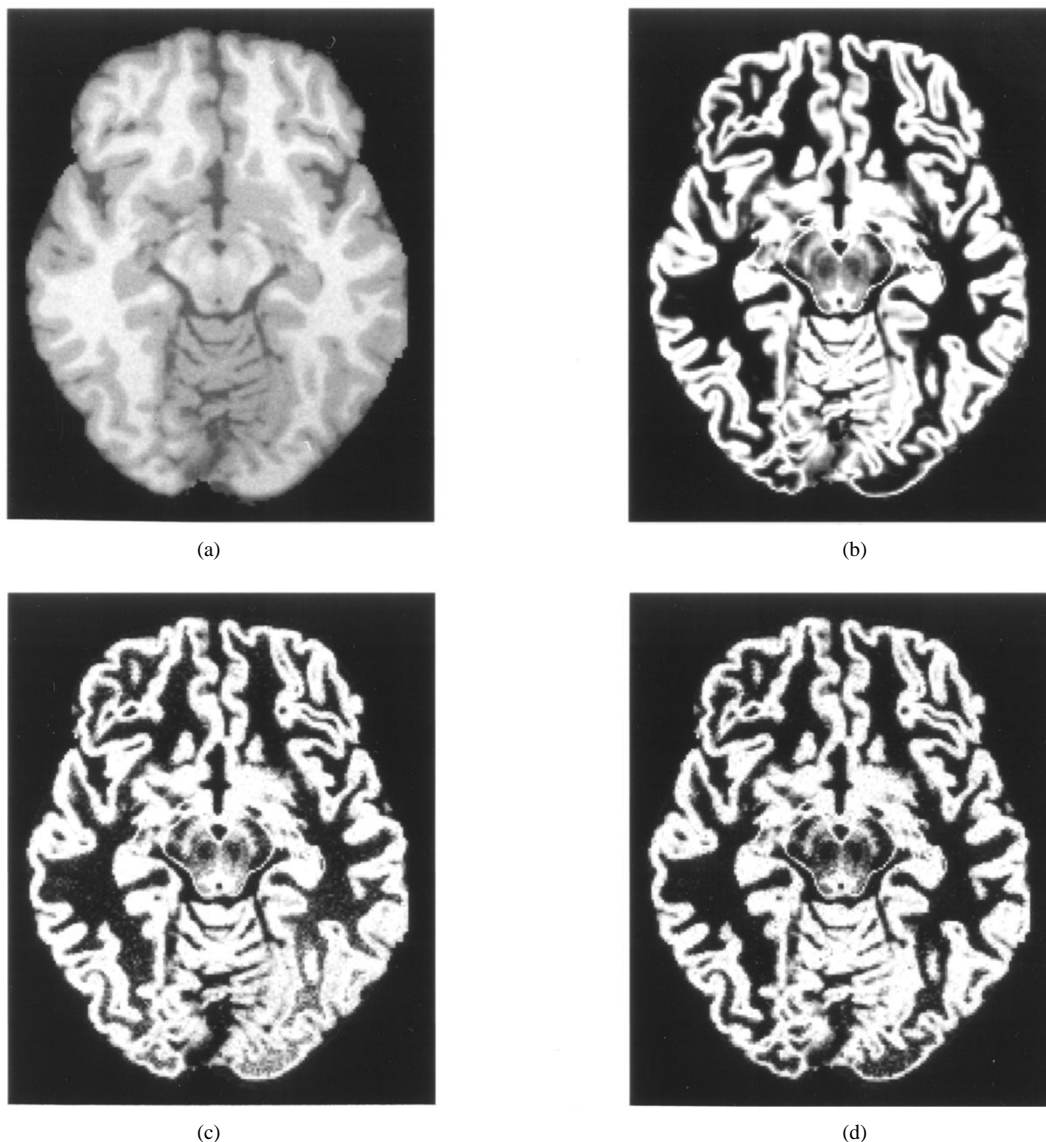
Fig. 2. FCM and AFCM membership functions. (a) Simulated MR phantom. (b) GM partial volume truth model. (c) FCM GM membership function. (d) TM-AFCM GM membership function.

contrast are available on the World Wide Web at the website listed in the references section. The inhomogeneity in these data sets was simulated by multiplying the image by an RF field recovered from an actual MR scan [32].

In this section, we present the application of AFCM only to 3-D brain images. For 2-D results, readers are referred to [19]. Extracranial tissues were removed from all images prior to applying any segmentation algorithm. For the real MR data, this was performed using a semiautomated technique described in [33]. In all results that follow, the value of q was set to two and the standard Euclidean distance norm was used. We denote the AFCM results computed with the full multigrid V cycle as FM-AFCM and the results computed with the truncated multigrid V cycle as TM-AFCM. Using FM-AFCM, execution times for a 3-D T_1-weighted MR data set with 1-mm cubic voxels are typically between 45 min and 3 h. Using TM-AFCM, execution times are between 10 min and 1 h. Quantitative evaluation indicates that this

speed increase is not achieved at the cost of segmentation accuracy.

A. Visual Evaluation of Performance on Simulated Data

Fig. 2 shows the results of applying FCM and AFCM on a Brainweb simulated MR brain image. This brain image was simulated with T_1-weighted contrast, 1-mm cubic voxels, 3% noise, and 40% image intensity inhomogeneity. The number of tissue classes was assumed to be three, corresponding to gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF) tissue classes. Background pixels were ignored. Fig. 2(a) shows a slice from the simulated data set and Fig. 2(b) shows the true partial volume model of the GM tissue class that was used to generate the simulated image. Fig. 2(c) and (d) shows the GM membership function obtained by applying FCM and TM-AFCM, respectively, to the 3-D data set. Because of the shading effect present in the data, the FCM membership function deteriorates near the bottom of the
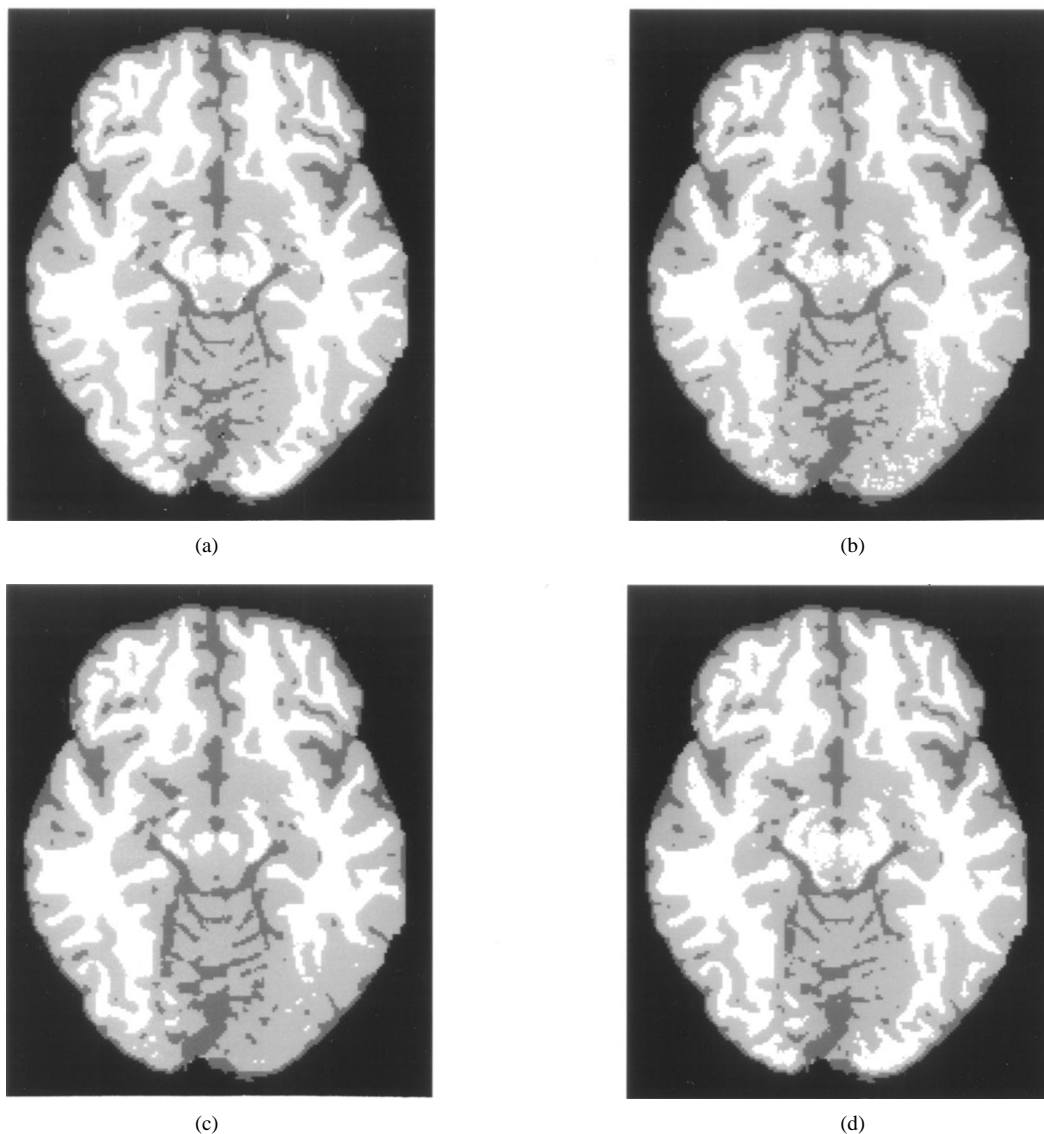
Fig. 3. Comparison of hard segmentations. (a) Truth model. (b) FCM max membership segmentation. (c) AMRF segmentation. (d) TM-AFCM max membership segmentation.

image. The AFCM result, however, shows less speckling at the bottom of the image and is very similar to the true partial volume image. Both results do, however, show some overall grain because of the effects of noise and because no constraint on spatial smoothness is placed on the membership functions in FCM or AFCM.

Fig. 3 shows the results of three different segmentation algorithms applied to the same data set described in the previous example. Fig. 3(a) shows the true hard segmentation of the simulated data. CSF is labeled as dark gray, GM as light gray, and WM as white. Fig. 3(b)–(d) shows the maximum membership segmentation produced by FCM, the segmentation produced by the adaptive Markov random field (AMRF[1]) method used in [15] and [33], and the maximum membership segmentation produced by TM-AFCM, respectively. Clearly, the AFCM segmentation is most similar to the truth model. Both the FCM and AMRF results segment much of the WM as GM near the bottom of the image. The AMRF segmentation is

[1] This method is also very similar to the one described in [16].

TABLE I
ERROR MEASURES FROM SIMULATED DATA RESULTS

Method	Error measure					
	0% MSE	20% MSE	40% MSE	0% MCR	20% MCR	40% MCR
FCM	0.0194	0.0272	0.0517	3.988%	5.450%	9.046%
FM-AFCM	0.0210	0.0242	0.0251	4.171%	4.322%	5.065%
TM-AFCM	0.0210	0.0214	0.0244	4.168%	4.322%	4.938%
EM1	0.0437	0.0491	0.0770	6.344%	7.591%	13.768%
EM2	0.0335	0.0391	0.0587	4.242%	5.638%	9.604%
AMRF				3.876%	4.795%	6.874%
MNI-FCM	–	–		4.979%	4.970%	5.625%
TM1-AFCM	–	–	–	4.156%	4.363%	4.994%
TM2-AFCM	–	–	–	4.125%	4.760%	5.560%
TM3-AFCM	–	–		3.998%	5.410%	8.292%

also spatially smoother than the other methods. This is because it takes into account pixel dependency, while both FCM and AFCM classify pixels independently.

B. Quantitative Evaluation of Performance on Simulated Data

Table I summarizes error measures resulting from applying the FCM, FM-AFCM, TM-AFCM, and the AMRF algorithms

to Brainweb simulated T_1-weighted data sets (1-mm cubic voxels, 3% noise) with varying levels of inhomogeneity. Errors were also computed from applying the unsupervised EM algorithm for finite Gaussian mixture models [34]. EM1 refers to the standard model and EM2 refers to the model where variances and mixture coefficients are assumed equal. The latter model was used, since it is conceptually and algorithmically similar to FCM, iterating between estimating the mean of each tissue class and computing a soft segmentation. In addition, error measures were also computed for a segmentation obtained by first applying the $N3$ inhomogeneity correction software [12] obtained from the Montreal Neurological Institute, then applying FCM. The results of this method are given in the row labeled MNI-FCM. Errors for intermediate results of TM-AFCM at each resolution level are shown in the rows labeled TML-AFCM, which stands for truncated multigrid AFCM at level L. Two error measures were used. The first measure was the mean squared error (MSE) between the true GM partial volume and the GM fuzzy membership function. The second error measure was the misclassification rate (MCR), defined as the number of pixels misclassified by the algorithm divided by the total number of pixels in the image. For FM-AFCM and TM-AFCM, the parameters λ_1 and λ_2 were fixed to a default value of 2×10^4 and 2×10^5, respectively. Default parameters were also used for all other segmentation methods.

Columns 1–3 show the MSE resulting from segmenting data sets with 0, 20, and 40% inhomogeneity, respectively. Similarly, columns 4–6 show the MCR for the same respective data sets. The MSE columns show that AFCM is capable of estimating partial volume coefficients with a reasonable accuracy, even in the presence of inhomogeneities. MSE errors for the EM methods were computed based on their posterior probability estimates. These probabilities tend to be too hard for T_1-weighted data, however, and this is reflected in the higher errors. Although none of these methods explicitly model partial volume effects, it has been shown in previous work that FCM can behave similarly to partial volume estimation approaches [35].

The MCR columns show that as the inhomogeneity is increased, the errors for all methods also increase. However, the AFCM methods are much more robust to increased inhomogeneity than the other methods, with TM-AFCM achieving slightly lower errors than FM-AFCM. In the case of 40% inhomogeneity, AFCM provides an improvement of nearly 50% over FCM, nearly 30% over the MRF methods, and over 10% over the MNI-FCM method. At zero inhomogeneity, both the FCM and AMRF methods perform slightly better than AFCM, while AMRF yields the lowest error. This is expected since the AMRF method provides some smoothing of noise, while FCM and AFCM do not. The increase in error of AFCM over FCM in the zero inhomogeneity case is due to the additional freedom of the gain field. This effect is also seen in the errors resulting from the MNI-FCM method. One could easily reduce the error by increasing the regularization terms if the amount of inhomogeneity was known to be low. The difference in error is small, however, and, overall, AFCM performs well on images of varying inhomogeneity, without the need for modifying the regularization parameters. Note that

TABLE II
MISCLASSIFICATION RATE AS A FUNCTION OF λ_1 AND λ_2

λ_1	λ_2					
	5×10^3	1×10^4	5×10^4	1×10^5	5×10^5	1×10^6
5×10^3	5.659	5.194	4.566	4.350	5.070	5.479
1×10^4	5.385	5.073	4.550	4.342	5.071	5.479
5×10^4	4.763	4.686	4.441	4.295	5.105	5.479
1×10^5	4.540	4.489	4.352	4.260	5.111	5.480
5×10^5	4.241	4.244	4.292	4.403	5.178	5.487
1×10^6	4.482	4.491	4.596	4.739	5.280	5.496

one can potentially achieve much lower errors in each of the AFCM, AMRF, and MNI-FCM methods if more information about the inhomogeneity is known *a priori*, thereby allowing some tailoring of their parameters.

Results of the TM-AFCM algorithm using the Lth pyramid level estimate show that in the 20 and 40% inhomogeneity cases the error is reduced rapidly as L is decreased. Because the gain field is slowly varying and spatially smooth, the difference in error between the level one and level zero results is small. However, in the zero inhomogeneity case, the error is actually increased. This is, again, because each increase in resolution adds additional degrees of freedom in how the multiplier field can vary. Note also that the TM-AFCM algorithm actually produces lower errors than FM-AFCM. We speculate that the slow increase in resolution provided by FM-AFCM allows it to avoid local minima in the AFCM objective function better than TM-AFCM. Further comparisons between the two approaches are discussed in Section IV-F.

C. Variation of Regularization Parameters

Table II shows the MCR after applying TM-AFCM to the simulated data set (1-mm cubic voxels, 3% noise, 20% inhomogeneity) when varying the regularization parameters λ_1 and λ_2. The 20% inhomogeneity was used because this is the default setting of the Brainweb database and is likely to be the level of inhomogeneity encountered in most T_1-weighted MR acquisitions. The table shows that even if the parameters are varied by an order of magnitude higher or lower than the optimal values, AFCM will still yield lower errors than FCM. For those values tested, the error was rarely significantly larger than the FCM error. However, the lowest error obtained was never lower than the FCM error on the zero inhomogeneity image (see Table I). Selecting larger values for λ_1 and λ_2 generally leads to more conservative results, since AFCM reduces to FCM in this case.

D. Correction of Inhomogeneities

Fig. 4 shows the results of using AFCM to correct the inhomogeneity in an actual 3-D T_1-weighted MR image data set. Fig. 4(a) shows a slice from the original data set. Fig. 4(b) shows the same slice after correction by AFCM. The correction was obtained by multiplying the original image by the reciprocal of the estimated gain field. The corrected image does not exhibit the left to right shading present in the original image. Fig. 4(c) shows the computed gain field for that slice. The gain field is actually computed everywhere in the image domain but, for visual purposes, it has been masked by the brain area. Note the bright area on the upper left quadrant of the image has been captured by the gain field.
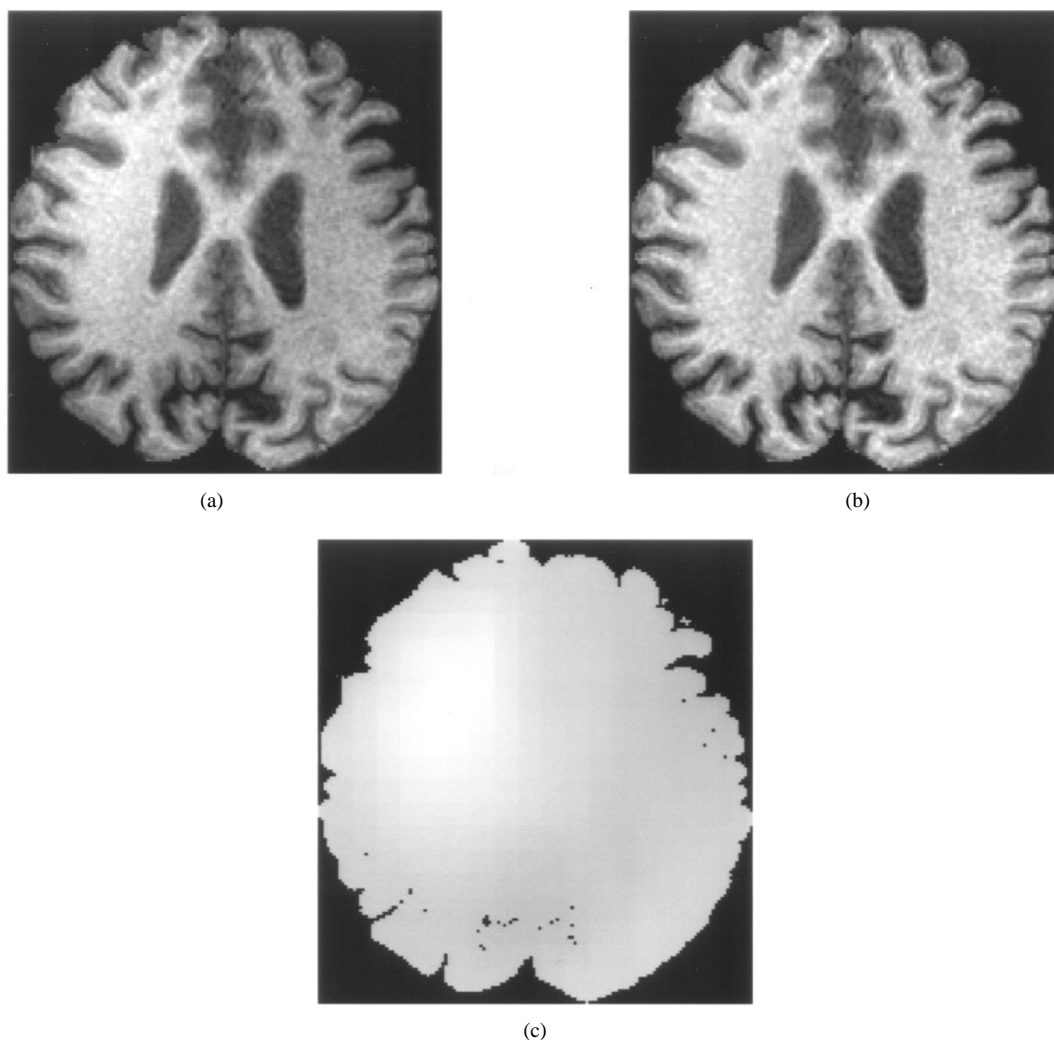
(a)

(b)

(c)

Fig. 4. Correction of inhomogeneity using TM-AFCM. (a) Slice from original MR image. (b) MR slice after AFCM correction. (c) Gain field computed using AFCM.

Fig. 4(d) and (e) shows histograms of the slice before and after the correction has been performed. On a typical histogram of an uncorrupted MR image, three modes are present corresponding to (from left to right) CSF, GM, and WM. The original histogram in Fig. 4(d), however, exhibits an additional mode around an intensity of 80 that corresponds to the bright WM on the upper left of the image slice. The corrected histogram does not possess this additional mode and also shows a significant improvement in contrast between the modes corresponding to GM and WM.

E. Multispectral Data

Fig. 5 shows the results of FCM and TM-AFCM when applied to a spin-echo [T_2-weighted and proton spin density (PD) weighted] multispectral MR data set. Fig. 5(a) and (b) shows a PD-weighted and the corresponding T_2-weighted slice, respectively, from the data set. Fig. 5(c) and (d) shows the GM membership functions computed by FCM and AFCM, respectively. One can see that the FCM membership function has a noticeable fading on the left side. There is also an increase in noise in the FCM membership function on the right side of the image. The AFCM membership function, however, is markedly cleaner from noise and does not exhibit the same

fading. Fig. 5(e) and (f) shows the contour of where the GM membership function is equal to the white matter membership function, overlayed on the PD-weighted slice. The inhomogeneity has the effect of shifting the boundaries between tissue classes. On the upper right-hand side of the image, the FCM contour has shifted inward toward the center of the image, while on the left of the image, the contour has shifted outward. The AFCM contour however, conforms to the GM-WM boundary, as seen on the original images much more accurately. Note that the ring of gray matter outside the brain region is due to errors in the brain extraction preprocessing step.

The vector gain field version of AFCM (see Appendix A) was also applied to the same data set. Although differences between the vector gain field segmentation and scalar gain field segmentation were negligible in this case and in most other data to which we have applied AFCM, the PD-weighted component of the estimated vector gain field typically possessed greater variation. This topic requires further investigation.

F. Comparisons of Truncated and Full Multigrid Approaches

In this section, the behavior of the AFCM objective function under the full and truncated multigrid approaches is examined. The objective function was evaluated at each iteration using
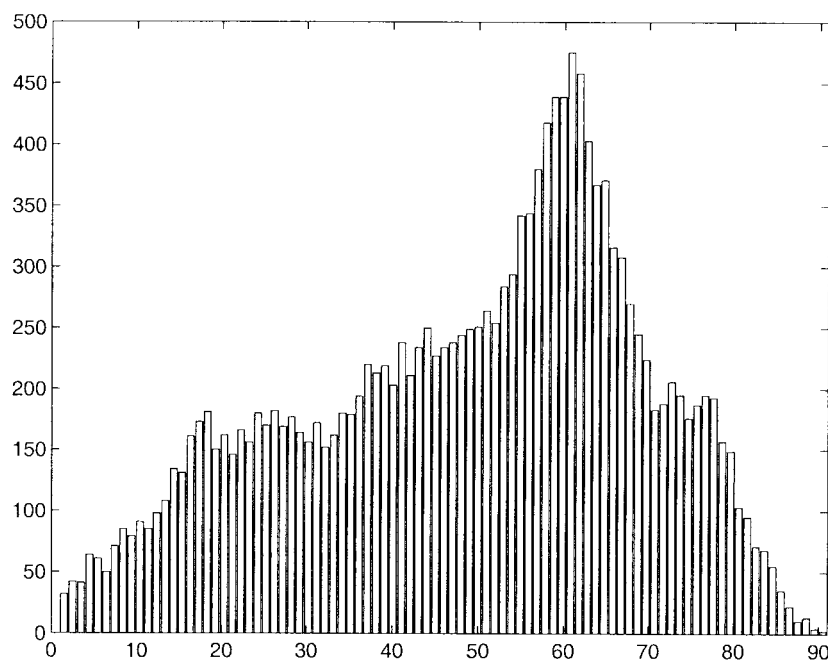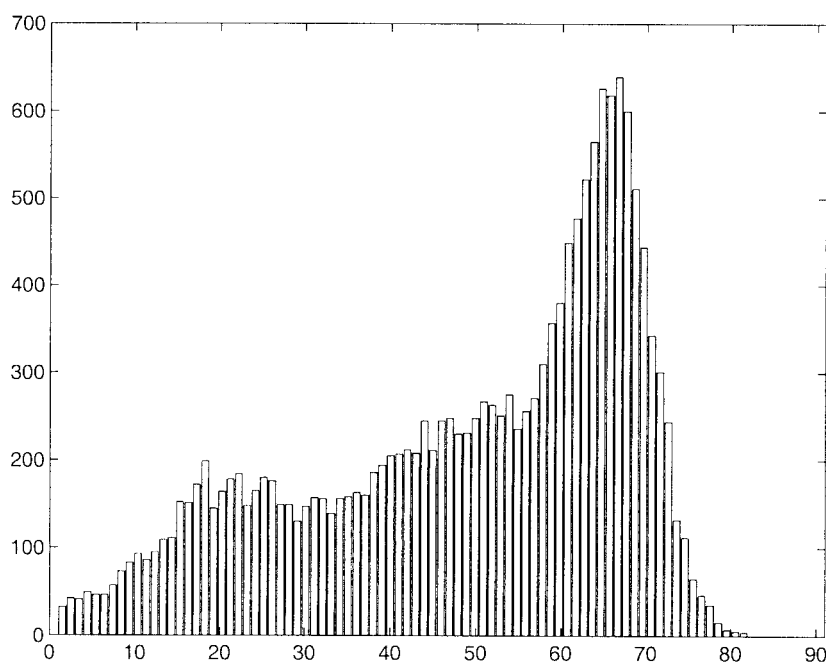
Fig. 4. (*Continued.*) Correction of inhomogeneity using TM-AFCM. (d) Histogram of slice before correction. (e) Histogram after correction.

both approaches for the McGill phantom data set with 3% noise and 20% inhomogeneity, as well as a real 3-D T_1-weighted MR data set with 1-mm cubic voxels. Fig. 6(a) and (b) plots the objective function over time for these two data sets, respectively. Each data point marker on the plots represents the value of the objective function taken after each iteration of AFCM. The full multigrid AFCM curves possess uniform spacing between each marker, since each iteration requires equal computational expense. On the other hand, the truncated multigrid results show increased spacing between markers as it moves from coarser resolution levels to finer levels. At the finest resolution, both methods require an equal amount of time for each iteration.

Both plots show that during early iterations of AFCM, the objective function decreases much more rapidly using the truncated multigrid approach. This is because the truncated method quickly computes an approximation of the gain field, allowing for more iterations of AFCM to take place and
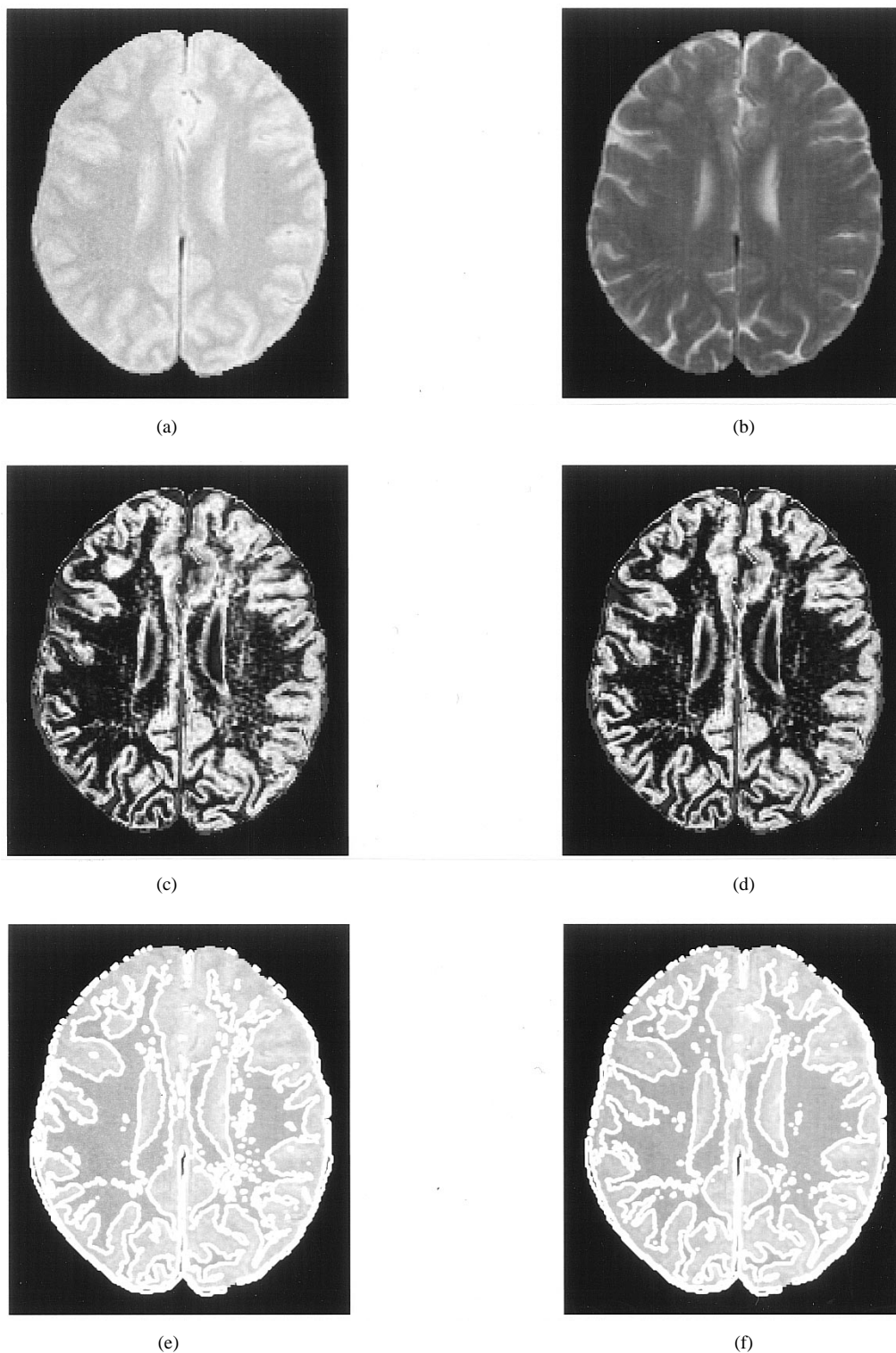
Fig. 5. FCM versus AFCM for double-echo MR data. (a) Slice from PD-weighted MR image. (b) Slice from T_2-weighted MR image. (c) FCM GM membership function. (d), TM-AFCM GM membership function. (e) FCM isocontour superimposed on PD-weighted image. (f) TM-AFCM isocontour superimposed on PD-weighted image.

yielding better estimates of the membership functions and centroids. In the case of Fig. 6(b), the objective function began to flatten horizontally during the later iterations at level three, but upon switching to level two, it resumed a steeper slope. The truncated multigrid approach also achieves a lower minimization of the objective function, even though both methods use the same stopping criterion (see Section III-A). This corroborates the results in Section IV-B, where the truncated multigrid approach achieved slightly lower segmentation errors than the full approach. For both examples shown, decrease in the objective function was monotonic for both full and truncated multigrid approaches. Note that the data set used
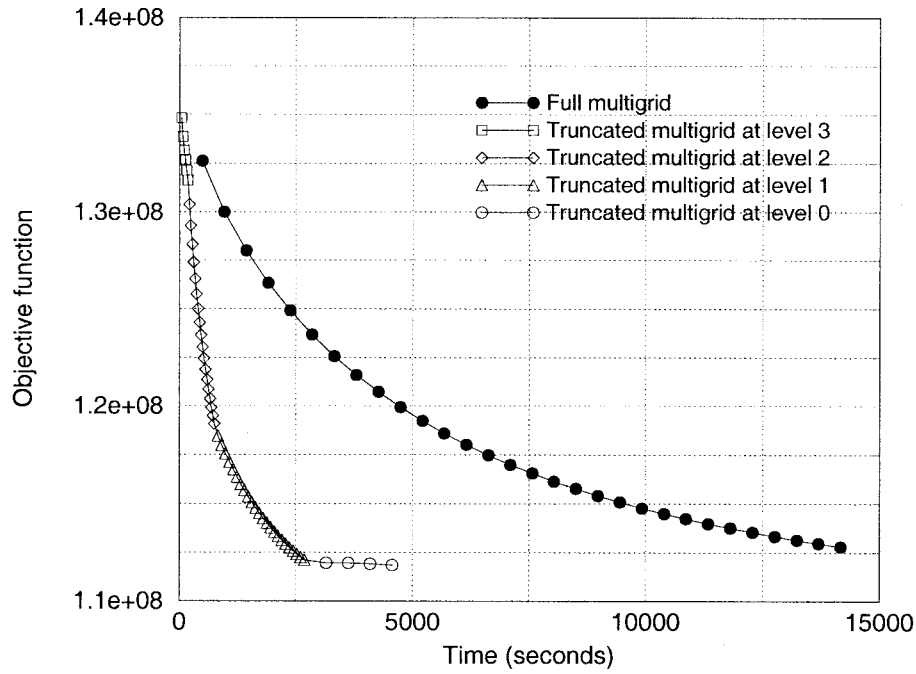
(a)
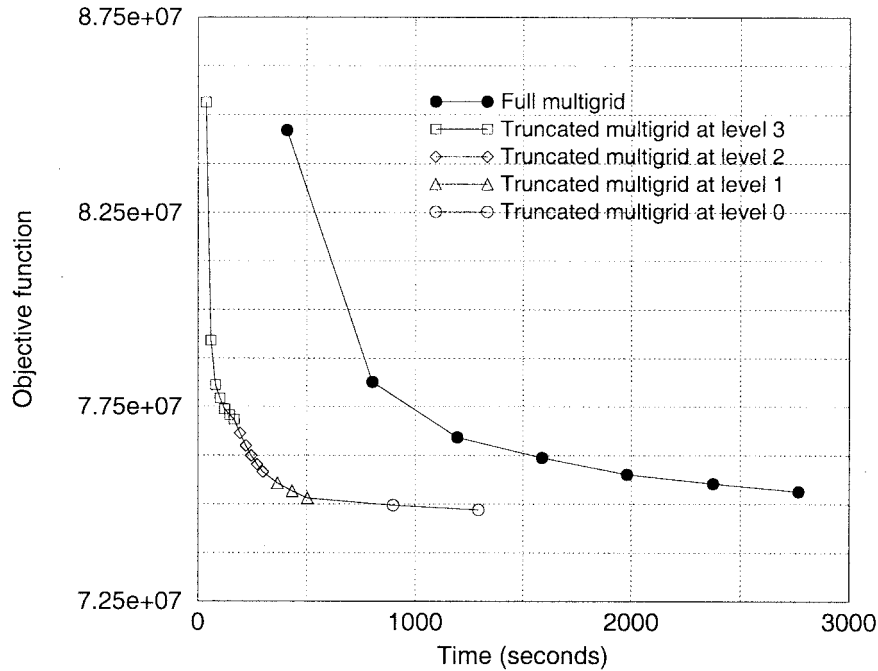
(b)

Fig. 6.   Plots of objective function over time for FM-AFCM and TM-AFCM: application to (a) phantom data with 3% noise, 20% inhomogeneity and (b) real T_1-weighted MR data set.

to generate Fig. 6(b) possessed lower levels of inohomogeneity than the phantom data and, thus, fewer iterations of AFCM were required.

V. DISCUSSION

It was shown in Section IV-B that AFCM was more robust to intensity inhomogeneities than two other established methods. Because the gain field is computed at every point in the image, AFCM has a potential advantage over these methods and other similar methods that use subsampling or regression techniques to compute the gain field (cf. [11], [12], [14], [16]). Although the inhomogeneities are typically slowly varying, there can be specific regions in the image where the inhomogeneities vary more quickly. This can be difficult to characterize when the field is subsampled. The key to the robustness of AFCM lies in the spatially varying difference equation (7), which governs how the gain field is estimated.

To better understand the action of (7) it is helpful to view it as a smoothing filter. If the membership functions and centroids are assumed to be known, then (7) can be shown to be a necessary condition for the solution of the variational problem

$$\arg \min_m \sum_{j \in \Omega} w_j (f_j - g_j)^2 + \lambda_1 \sum_{j \in \Omega} \sum_r^R (D_r * g_j)^2$$

$$+ \lambda_2 \sum_{j \in \Omega} \sum_{r=1}^R \sum_{s=1}^R (D_r * D_s * g_j)^2 \qquad (15)$$

where the functions w and f were defined in Section III-C. We refer to the first term in (15) as the data term and the last two terms as the smoothing terms. Thus, the difference equation governing the gain field can be seen as a smoothing of the function f where w is a weighting function that controls where the gain field should be close to f. In areas of the image where w_j is large relative to λ_1 and λ_2, the data term dominates and g_j is forced to be equal or close to f_j. On the other hand, where w_j is small, then the smoothing terms dominate and g_j is forced to be equal or close to the value of its neighbors.

The function f is essentially a function of the ratios of the observed intensity to the centroids. This is more easily seen when considering the special case when $u_{jb} = 1$ for some $b \in \{1, \cdots, C\}$, and y_j is a scalar. In this case, $f_j = y_j/v_b$, the observed intensity divided by the centroid. In an ideal image, the intensity of each class would be constant and equal to the centroid. Thus, any gain field effect could be computed using this ratio. The membership functions weight the ratio according to which centroid the observed intensity is closest.

The function w_j is small when the membership functions are low for all classes, implying that the intensity is an outlier. This corresponds to either partially volume-averaged pixels or pixels originating from unconsidered tissue classes. The smoothing terms therefore dominate in these areas of the image. The function w_j also has a dependency on the magnitude of the centroid vectors. This property originates from the AFCM objective function (3) and the desire to minimize $\|\mathbf{y}_j - g_j \mathbf{v}_k\|$. When the magnitude of a centroid is large, small variations in g_j result in a larger normed difference than when the centroid is small. Thus, a greater weight is necessary on the data term to equalize the differences. The reason why this dependence is necessary is because classes with higher intensity will locally possess a higher signal-to-noise ratio than classes with lower intensity. Thus, gain field estimates from areas of the image with higher intensity can be trusted more than estimates from areas of low intensity.

The spatially varying nature of (7) simplifies dealing with boundary conditions. In MR images, it is common to attempt to segment an object such as a brain surrounded by empty space. Thus, boundary conditions can become a problem with spatially invariant filtering methods since it is undesirable to blur the background with the object. In [8] and [9] it was necessary to fill the background with the average intensity of the object, while in [7] a predefined region of interest was required in certain cases. In AFCM, when a background tissue class with a centroid approximately equal to zero is used, w_j

is automatically set to zero outside the object of interest. Thus, the background plays little or no role in the data term of (15).

It is possible to give a statistical interpretation to (15) (cf. [36]). This interpretation might prove beneficial if an optimal selection of the parameters λ_1 and λ_2 is desired given the data. We showed in Section IV-C, however, that AFCM is fairly insensitive to the selection of these parameters and, in practice, one can normally use a set of standard fixed parameters.

Several variations on the objective function (3) are possible, which lead to algorithms similar to AFCM. An obvious alternative would be to allow the gain field to multiply the observed intensities $\mathbf{y}_j$, rather than the centroids $\mathbf{v}_k$. This leads to a slightly more computationally complex difference equation for the solution of the gain field. Another alternative formulation that was suggested by a reviewer replaces the expression $\|\mathbf{y}_j - g_j \mathbf{v}_k\|$ in the objective function with $\|\mathbf{y}_j/\mathbf{v}_k - g_j [1, \cdots, 1]^T\|$, where the division is performed component-wise. A potential advantage of this formulation is that the resulting weights w_j would depend exclusively on the membership functions. We have found in initial experiments that this approach can provide reasonable results, although convergence is slower and the resulting membership functions behave differently than those produced by FCM or AFCM. Further investigation is necessary to fully determine the benefits and disadvantages of this approach.

VI. CONCLUSIONS AND FUTURE WORK

We have presented an algorithm for obtaining fuzzy segmentations of images that have been corrupted by intensity inhomogeneities. The algorithm is fully automated, except for the initial specification of some parameters. In the experiments performed, AFCM yielded accurate segmentations in the presence of intensity inhomogeneities and can be directly substituted into current methodologies that require: 1) hard segmentations; 2) soft segmentations; 3) gain field estimates; or 4) inhomogeneity corrected images. We are currently using AFCM in conjunction with deformable surface algorithms for the reconstruction of the human cerebral cortex from MR images [23].

Several areas require further research. One disadvantage of AFCM is that it tends to look for clusters of the same shape and size. AFCM can, however, be extended to find differently shaped and sized clusters in the same manner as FCM [37], [38]. This extension would take advantage of the covariance characteristics of each tissue, potentially resulting in improved segmentation accuracy for multispectral images. However, because of the increased complexity that would be involved, issues such as sensitivity to initial conditions and speed of convergence may arise. Another approach to this problem is to use AFCM as a preprocessing step to a finite Gaussian mixture model [34] analysis of the data, where AFCM is used to correct the inhomogeneity and the fuzzy membership functions are treated as initial posterior probabilities for an expectation-maximization algorithm [25], [26]. This approach leads to a statistically optimal partitioning of the data as long as the assumption of a Gaussian mixture model is correct.

Because AFCM does not place any contextual constraints on the membership functions it can be sensitive to excessive noise and other artifacts, such as inaccuracies in the extracranial tissue removal. Further research on incorporating spatial information through Markov random fields or atlas information may alleviate this sensitivity.

Additional validation studies are required if the method is to be used for clinical studies. Furthermore, the convergence properties of the algorithm require further investigation although, in practice, we have found it to be very stable and to always converge. Finally, although the smoothness parameters are fairly robust to selection, a method for optimally selecting their value would prove beneficial in ensuring accurate performance.

APPENDIX A
AFCM ALGORITHM FOR VECTOR GAIN FIELDS

In Step 2 of AFCM, the membership function computation of (5) is replaced by

$$u_{jk} = \frac{\|\mathbf{y}_j - \mathbf{G}_j\mathbf{v}_k\|^{-2/(q-1)}}{\sum_{l=1}^{C} \|\mathbf{y}_j - \mathbf{G}_j\mathbf{v}_l\|^{-2/(q-1)}},$$

$$j \in \Omega, k = 1, \cdots, C. \qquad (16)$$

In Step 3, the centroid computation of (6) is replaced by

$$[\mathbf{v}_k]_i = \frac{\sum_j u_{jk}^q [\mathbf{G}_j\mathbf{y}_j]_i}{\sum_{j \in \Omega} u_{jk}^q [\mathbf{g}_j]_i^2}, \qquad k = 1, \cdots, C, \ i = 1, \cdots, P.$$

$$(17)$$

In Step 4, the difference (7) governing the gain field is replaced by

$$[\mathbf{y}_j]_i \sum_{k=1}^{C} u_{jk}^q [\mathbf{v}_k]_i$$

$$= [\mathbf{g}_j]_i \sum_{k=1}^{C} u_{jk}^q [\mathbf{v}_k]_i^2 + \lambda_1 (H_1 * [\mathbf{g}]_i)_j + \lambda_2 (H_2 * [\mathbf{g}]_i)_j,$$

$$i = 1, \cdots, P. \qquad (18)$$

Note that in this case, the difference equation must be solved for each component of the data.

APPENDIX B
DERIVATION OF AFCM EQUATION

In this section, we derive the scalar gain field AFCM equations. The derivation in the vector gain field case is analogous. The derivation of (5) is almost identical to the derivation of the membership function equation in the standard FCM algorithm [20], [21]. We first rewrite the objective function using a Lagrange multiplier κ_j to enforce the constraint that $\sum_{k=1}^{C} u_{jk} = 1$. Ignoring the regularization terms, since these will be zero when the partial derivative with respect to u_{jk}

is taken, this yields

$$J_{\text{AFCM}} = \sum_{j \in \Omega} \left(\sum_{k=1}^{C} u_{jk}^q \|\mathbf{y}_j - g_j\mathbf{v}_k\|^2 + \kappa_j \left(1 - \sum_{k=1}^{C} u_{jk} \right) \right).$$

Taking the partial derivative with respect to u_{jk} and setting the result equal to zero yield

$$u_{jk} = \left(\frac{\kappa_j}{q\|\mathbf{y}_j - g_j\mathbf{v}_k\|^2} \right)^{1/(q-1)}. \qquad (19)$$

Substituting u_{jk} into the constraint equation on the membership functions results in

$$\kappa_j = \frac{q}{\sum_{k=1}^{C} \|\mathbf{y}_j - g_j\mathbf{v}_k\|^{-2}}.$$

Substituting the value of κ_j back into (19) and rearranging yields (5).

The derivation of (6) proceeds as follows. Any real, finite-dimensional inner product norm must satisfy $\|\mathbf{f}\| = \mathbf{f}^T \mathbf{Q}\mathbf{f}$ for some positive definite matrix $\mathbf{Q}$. Substituting this identity into (3), taking the partial derivative with respect to the centroids $\mathbf{v}$, and setting the result to zero yields

$$-2 \sum_{j \in \Omega} u_{jk}^q g_j \mathbf{Q}\mathbf{y}_j + 2 \sum_{j \in \Omega} u_{jk}^q g_j^2 \mathbf{Q}\mathbf{v}_k = 0, \qquad k = 1, \cdots, C.$$

Using the fact that $\mathbf{Q}$ is positive definite and rearranging leads to (6).

To derive (7) we use the following lemma

$$\frac{\partial}{\partial g_j} \sum_j [H * g]_j^2 = 2(H * \breve{H}) * g_j$$

which was proven in [39]. Taking the partial derivative of (3) with respect to the gain field g_j yields

$$- 2 \sum_{k=1}^{C} u_{jk}^q \langle \mathbf{y}_j, \mathbf{v}_k \rangle + 2g_j \sum_{k=1}^{C} u_{jk}^q \langle \mathbf{v}_k, \mathbf{v}_k \rangle$$

$$+ 2\lambda_1 [H_1 * g]_j + 2\lambda_2 [H_2 * g]_j = 0.$$

Dividing through by 2 and rearranging yields (7).

APPENDIX C
CONVOLUTION MASKS

Standard finite differences were used for computing the convolution kernels H_1 and H_2 used in the difference equation (7). For 2-D images, the resulting kernels are

$$H_1 = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

$$H_2 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & -8 & 2 & 0 \\ 1 & -8 & 20 & -8 & 1 \\ 0 & 2 & -8 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

For 3-D images, we provide the kernels in slices, with the corresponding slice numbers denoted in the superscript.

All kernels are symmetric. Note that for data with extremely anisotropic voxels, asymmetric kernels may be desirable

$$H_1^{1,3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$H_1^2 = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 6 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

$$H_2^{1,5} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$H_2^{2,4} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 2 & -12 & 2 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$H_2^3 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & -12 & 2 & 0 \\ 1 & -12 & 42 & -12 & 1 \\ 0 & 2 & -12 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

ACKNOWLEDGMENT

REFERENCES

[1] J. C. Bezdek, L. O. Hall, and L. P. Clarke, "Review of MR image segmentation techniques using pattern recognition," *Med. Phys.,* vol. 20, pp. 1033–1048, 1993.

[2] J. K. Udupa and S. Samarasekera, "Fuzzy connectedness and object definition: Theory, algorithms and applications in image segmentation," *Graph. Models Image Processing,* vol. 58, no. 3, pp. 246–261, 1996.

[3] M. E. Brandt, T. P. Bohan, L. A. Kramer, and J. M. Fletcher, "Estimation of CSF, white and gray matter volumes in hydrocephalic children using fuzzy clustering of MR images," *Comput. Med. Imag. Graph.,* vol. 18, pp. 25–34, 1994.

[4] A. Simmons, P. S. Tofts, G. J. Barker, and S. R. Arridge, "Sources of intensity nonuniformity in spin echo images at 1.5T," *Magnetic Resonance Med.,* vol. 32, pp. 121–128, 1994.

[5] B. R. Condon, J. Patterson, *et al.,* "Image nonuniformity in magnetic resonance imaging: Its magnitude and methods for its correction," *Br. J. Radiol.,* vol. 60, pp. 83–87, 1989.

[6] B. M. Dawant, A. P. Zijidenbos, 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.

[7] W. M. Wells, W. E. L. Grimson, R. Kikins, and F. A. Jolesz, "Adaptive segmentation of MRI data," *IEEE Trans. Med. Imag.,* vol. 15, pp. 429–442, Aug. 1996.

[8] B. Johnston, M. S. Atkins, B. Mackiewich, and M. Anderson, "Segmentation of multiple sclerosis lesions in intensity corrected multispectral MRI," *IEEE Trans. Med. Imag.,* vol. 15, pp. 154–169, Apr. 1996.

[9] K. O. Lim and A. Pfefferbaum, "Segmentation of MR brain images into cerebrospinal fluid and white and gray matter," *J. Comput. Assisted Tomogr.,* vol. 13, pp. 588–593, 1989.

[10] 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.

[11] C. R. Meyer, H. B. Peyton, and J. Pipe, "Retrospective correction of intensity inhomogeneities in MRI," *IEEE Trans. Med. Imag.,* vol. 14, pp. 36–41, Mar. 1995.

[12] 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.

[13] S. K. Lee and M. W. Vannier, "Post-acquisition correction of MR inhomogeneities," *Magnetic Resonance Med.,* vol. 36, pp. 276–286, 1996.

[14] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Trans. Signal Processing,* vol. 40, pp. 901–914, Apr. 1992.

[15] M. X. H. Yan and J. S. Karp, "An adaptive bayesian approach to three-dimensional MR brain segmentation," in *Proc. XIVth Int. Conf. Information Processing Medical Imaging,* 1995, pp. 201–213.

[16] J. C. Rajapakse, J. N. Giedd, and J. L. Rapoport, "Statistical approach to segmentation of single-channel cerebral MR images," *IEEE Trans. Med. Imag.,* vol. 16, pp. 176–186, Apr. 1997.

[17] M. Unser, "Multigrid adaptive image processing," in *Proc. IEEE Conf. Image Processing (ICIP'95),* 1995, vol. I, pp. 49–52.

[18] R. Guillemaud and M. Brady, "Estimating the bias field of MR images," *IEEE Trans. Med. Imag.,* vol. 16, pp. 238–251, June 1997.

[19] D. L. Pham and J. L. Prince, "An adaptive fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities," *Pattern Recognit. Lett.,* vol. 20, pp. 57–68, 1998.

[20] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-sparated clusters," *J. Cybern.,* vol. 3, pp. 32–57, 1973.

[21] J. C. Bezdek, "A convergence theorem for the fuzzy ISODATA clustering algorithms," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. PAMI-2, pp. 1–8, Jan. 1980.

[22] D. L. Pham, J. L. Prince, A. P. Dagher, and C. Xu, "An automated technique for statistical characterization of brain tissues in magnetic resonance imaging," *Int. J. Pattern Recognit. Artificial Intell.,* vol. 11, no. 8, pp. 1189–1211, 1997.

[23] C. Xu, D. L. Pham, and J. L. Prince, "Finding the brain cortex using fuzzy segmentation, isosurfaces, and deformable surfaces," in *Proc. XVth Int. Conf. Information Processing Medical Imaging (IPMI'97),* 1997, pp. 399–404.

[24] 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.

[25] I. Gath and A. B. Geva, "Fuzzy clustering for the estimation of the parameters of the components of mixtures of normal distributions," *Pattern Recognit. Lett.,* vol. 9, pp. 77–86, 1989.

[26] J. W. Davenport, J. C. Bezdek, and R. J. Hathaway, "Parameter estimation for finite mixture distributions," *Comput. Math. Appl.,* vol. 15, pp. 810–828, 1988.

[27] D. L. Pham and J. L. Prince, "An adaptive fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities," in *Proc. SPIE Medical Imaging'98: Image Processing,* San Diego, CA, Feb. 21–27, 1998.

[28] B. W. Silverman, *Density Estimation for Statistics and Data Analysis.* London, U.K.: Chapman & Hall, 1993.

[29] P. A. Bottomley, T. H. Foster, R. E. Argersinger, and L. M. Pfeifer, "A review of normal tissue hydrogen NMR relaxation times and relaxation mechanisms from 1–100 MHz: Dependence on tissue type, NMR frequency, temperature, species, excision, and age," *Med. Phys.,* vol. 11, pp. 425–448, 1984.

[30] W. H. Press and B. P. Flannery, *et al., Numerical Recipes in C.* Cambridge, U.K.: Cambridge Univ. Press, 1988.

[31] W. L. Briggs, *A Multigrid Tutorial.* New York: Soc. Indus. Appl. Math., 1987.

[32] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, and A. C. Evans. (1997). BrainWeb: Online interface to a 3D MRI simulated brain database. *Neuroimage,* vol. 5, no. 4. Available: http://www.bic.mni.mcgill.ca/brainweb, 1997.

[33] 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. Assisted Tomogr.,* vol. 22, no. 5, pp. 827–837, 1998.

[34] 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.

[35] D. L. Pham and J. L. Prince, "Partial volume estimation and the fuzzy *c*-means algorithm," in *Proc. Int. Conf. Image Processing (ICIP'98),* 1998, vol. III, pp. 819–822.

[36] S. J. Reeves, "Optimal space-varying regularization in iterative image restoration," *IEEE Trans. Image Processing,* vol. 3, pp. 319–324, May 1994.

[37] D. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proc. IEEE CDC 2,* 1979, pp. 761–766.

[38] M. P. Windham, "Geometrical fuzzy clustering algorithms," *Fuzzy Sets Syst.,* vol. 10, pp. 271–279, 1983.

[39] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I—Theory," *IEEE Trans. Signal Processing,* vol. 41, pp. 821–833, Feb. 1993.