A new fuzzy c-means method with total variation regularization for segmentation of images with noisy and incomplete data

Yanyan He\textsuperscript{a}, M. Yousuff Hussaini\textsuperscript{a}, Jianwei Ma\textsuperscript{a,d,n}, Behrang Shafei\textsuperscript{b}, Gabriele Steidl\textsuperscript{c}

\textsuperscript{a} Department of Mathematics, Florida State University, Tallahassee, FL, USA
\textsuperscript{b} Fraunhofer ITWM, Image Processing Department, Kaiserslautern, Germany
\textsuperscript{c} Department of Mathematics, University of Kaiserslautern, Germany
\textsuperscript{d} Institute of Applied Mathematics, Harbin Institute of Technology, Harbin, China

\textbf{A R T I C L E  I N F O}

Article history:
Received 10 August 2011
Received in revised form 23 January 2012
Accepted 7 March 2012
Available online 23 March 2012

Keywords:
Fuzzy c-means
Multi-class labeling
Sparsity-promoting method
Alternating direction method of multipliers
MRI segmentation
Noisy and incomplete data

\textbf{A B S T R A C T}

The objective function of the original (fuzzy) c-mean method is modified by a regularizing functional in the form of total variation (TV) with regard to gradient sparsity, and a regularization parameter is used to balance clustering and smoothing. An alternating direction method of multipliers in conjunction with the fast discrete cosine transform is used to solve the TV-regularized optimization problem. The new algorithm is tested on both synthetic and real data, and is demonstrated to be effective and robust in treating images with noise and missing data (incomplete data). © 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Image segmentation is a process of partitioning an image into non-empty, pairwise disjoint segments so that the pixels in the same segment share some characteristics, such as intensity, texture or color. It is an important step in many applications of image processing, including medical imaging, machine learning, remote sensing, robotics, etc. In the last few decades, segmentation algorithms of different types have been developed [1,2], e.g., mean shift segmentation [3], fuzzy c-means (FCM) [4], and active appearance models [5], active contour models/snakes [6] and, active shape models [7]. Generally, most contour-based methods (e.g., snakes) work well for images with obvious edges and close-form boundaries, while intensity-based models (e.g., FCM) are suitable for images with highly variable intensity or complex segmentation boundaries. In this paper, we focus on improvement of the FCM approach and apply it to MRI brain image segmentation, which plays an important role in neurosurgical planning and clinical diagnosis.

The FCM method developed by Bezdek et al. [4] has been widely used for image segmentation. It is easy to implement, robust to blurring, applicable to multispectral data and requires no assumptions on the probability density function of the data. However, the original FCM method is sensitive to noise and incomplete data because it considers only the intensity of the image and does not take the spatial context and boundary connections into consideration. It has been improved in different ways to overcome this disadvantage. One approach is to pre-process the image before applying FCM, see, e.g., [8,9]. Szilágyi et al. [8] apply a linearly-weighted averaging filter to the original image to smooth the noise. Cai et al. [9] use a local neighbor averaged image according to the local similarity. This approach labels the pixels associating them with their neighborhood such that spatially close pixels tend to get homogeneous labeling. However, these smoothing filters have two main disadvantages—loss of important image information (such as discontinuous edges) and poor balance between smoothing and clustering. Another approach to make the FCM algorithm more robust to noise is to add a regularization term to the objective function, which takes into account the influence of spatial information of the local neighborhood. In some papers [10,11], the Euclidean distance between the pixel and a class centroid is replaced by a new dissimilarity index, which is the weighted sum of the distances within a neighborhood, while in others [12–15], regularization terms are introduced to constrain the behavior of the membership function retaining the same computations for centroids as in the original FCM algorithm. Li and Mukaidono [12] apply regularization by the entropy of the membership function,
while Miyamoto and Umayahara [13] regularized the FCM by a quadratic term. Both of these regularizers restrict the admissible membership function within the space of smooth functionals. However, similar to the original FCM algorithm, they are only based on the image intensity. Pham [14] proposes a new regularization term to incorporate spatial information where the term discourages unlikely or undesirable configurations in the membership functions. Hou et al. [15] exploit a moving-average filter as the regularizer which constrains the membership function by the windowed average of local neighborhoods.

MR imaging has to cope with intensity inhomogeneities visible as slowly varying shadow artifacts in the final image if disregarded in the reconstruction process. For field inhomogeneity correction based MRI reconstruction methods, see [16] and the references therein. For images with intensity nonuniformities, several segmentation methods were proposed ranging from two-step algorithms which first remove intensity inhomogeneities followed by some standard segmentation algorithm to adapted methods from various fields. With respect to this paper, we mention only some modified FCM approaches which take the intensity inhomogeneities into account. In general, the intensity nonuniformity (intensity inhomogeneity) is modeled as a multiplicative bias field (grain field) (sometimes after taking the logarithm as an additive field). In Pham et al. [23], the unknown grain field was incorporated within the minimizing FCM functional and appropriate grain field regularizing terms were added to correct the smooth nonuniformity within the segmentation process. This method was further improved by Cao et al. [24] for M-FISH images by using another regularization term. An adaptive spatial FCM clustering algorithm for MRI images corrupted by noise and intensity nonuniformity artifacts based on a dissimilarity index that allows spatial interactions between image pixels was proposed by Liew et al. [25].

Recently, sparsity-promoting regularization has been widely applied to inverse problems in applied mathematics and information theory. It incorporates prior knowledge of the sparsity of the solution into the optimization or minimization process. A straightforward way to measure the sparsity of the solution uses the $\ell_0$-‘norm’, i.e., the number of nonzero entries. Unfortunately, optimization with the $\ell_0$-‘norm’ is a NP-hard problem. Alternatively, the $\ell_1$-norm or $\ell_p$-quasinorms ($0 < p < 1$) can be considered to measure sparsity. The sparsity can be interpreted as spatial sparsity of the signal, or as coefficient sparsity in a proper dictionary or as a sparse gradient, i.e., the image is sparse under the total-variation (TV) operator. The gradient sparsity-promoting norm was recently also explored for multi-class image partitioning and labeling, e.g., in the convex relaxation approaches of the Potts model [26] proposed in [27–30]. Note that these methods require a priori information on the segmented images as, e.g., of the segment prototypes.

To exploit the sparse gradient property of MRI images [31], we propose a sparsity promoting method called total variation (TV) regularized FCM (TVFCM), and employ an alternating direction method of multipliers (ADMM) to solve a part of the optimization problem. Within the ADMM algorithm, we apply the fast cosine transform of type II (DCT-II). The TV regularizer works effectively on gradient-sparse images for removing spurious oscillations while preserving sharp edges. More precisely, Strong and Chan [32] showed that for piecewise constant features, the change in image intensity due to the TV regularizer is inversely proportional to the scale of image feature, which explains why the noise (which can be thought as small–scaled image features) is removed while the image edges (whose scales tend to be large relative to noise) is preserved. With this nice property, TV-based algorithms are capable of processing images with a sparse gradient property involving noise or incomplete data. However, our TV regularized methods are not suited to handle the above-mentioned smoothly and slowly varying intensity inhomogeneities since it would introduce the typical staircasing effects. We want to mention that such inhomogeneities can be corrected in the image acquisition step from the MRI data, see [16] such that the final image does not show these artifacts (which would be our method of choice). Otherwise, we have to perform one of the above-mentioned preprocessings or to use another regularizer as, e.g., in [33]. In the present work, we apply this TV regularizer on membership functions to incorporate the spatial information and boundary connection in the MRI images, and a regularization parameter $\mu$ is introduced to control the trade-off between smoothing and clustering. Our motivation is to make the segmentation regions and boundaries well connected.

The structure of this paper is as follows: after introducing the general notation in Section 2, the original FCM method is reviewed in Section 3. Since we need it for our numerical comparisons, the moving-average FCM algorithm is briefly explained in Section 4. The new TVFCM model with a corresponding minimization algorithm is presented in Section 5. Finally, Section 6 illustrates the performance of the algorithm on noisy images and missing data images, where both synthetic and real-world images are involved. Conclusions are given in Section 7.

2. General setting

We consider gray-value images $F : \{1, \ldots, n_1\} \times \{1, \ldots, n_2\} \rightarrow \mathbb{R}$ columnwise reshaped as $f := \text{vec}(F) : \Gamma \rightarrow \mathbb{R}$ where $\Gamma := \{1, \ldots, n\}$ and $n = n_1 n_2$. When dealing with such reshaped images, it is useful to employ an exact notation to work with the tensor product of matrices. The tensor product of $A \in \mathbb{R}^{n_1 \times n_2}$ and $B \in \mathbb{R}^{m_1 \times m_2}$ is given by the matrix

$$A \otimes B := \begin{pmatrix} a_{11} B & \cdots & a_{1n_2} B \\ \vdots & \ddots & \vdots \\ a_{n_1 1} B & \cdots & a_{n_1 n_2} B \end{pmatrix} \in \mathbb{R}^{n_1 m_1 \times n_2 m_2}.$$  

The tensor product is associative and distributive, but not commutative. Further, the relations

$$(A \otimes B)^T = A^T \otimes B^T \text{ and } (A \otimes B)(C \otimes D) = AC \otimes BD$$

hold true. Using the reshaping operator vec again, we can verify that

$$\text{vec}(AFB^T) = (B \otimes A)\text{vec}(F).$$

For our TVFCM method, we need two ingredients, namely the discrete TV functional as the regularizer which keeps the boundaries between the segments small and the FCM approach to assign the solution into the optimization or minimization process. A straightforward way to measure the sparsity of the solution uses the $\ell_0$-‘norm’, i.e., the number of nonzero entries. Unfortunately, optimization with the $\ell_0$-‘norm’ is a NP-hard problem. Alternatively, the $\ell_1$-norm or $\ell_p$-quasinorms ($0 < p < 1$) can be considered to measure sparsity. The sparsity can be interpreted as spatial sparsity of the signal, or as coefficient sparsity in a proper dictionary or as a sparse gradient, i.e., the image is sparse under the total-variation (TV) operator. The gradient sparsity-promoting norm was recently also explored for multi-class image partitioning and labeling, e.g., in the convex relaxation approaches of the Potts model [26] proposed in [27–30]. Note that these methods require a priori information on the segmented images as, e.g., of the segment prototypes.

To exploit the sparse gradient property of MRI images [31], we propose a sparsity promoting method called total variation (TV) regularized FCM (TVFCM), and employ an alternating direction method of multipliers (ADMM) to solve a part of the optimization problem. Within the ADMM algorithm, we apply the fast cosine transform of type II (DCT-II). The TV regularizer works effectively on gradient-sparse images for removing spurious oscillations while preserving sharp edges. More precisely, Strong and Chan [32] showed that for piecewise constant features, the change in image intensity due to the TV regularizer is inversely proportional to the scale of image feature, which explains why the noise (which can be thought as small–scaled image features) is removed while the image edges (whose scales tend to be large relative to noise) is preserved. With this nice property, TV-based algorithms are capable of processing images with a sparse gradient property involving noise or incomplete data. However, our TV regularized methods are not suited to handle the above-mentioned smoothly and slowly varying intensity inhomogeneities since it would introduce the typical staircasing effects. We want to mention that such inhomogeneities can be corrected in the image acquisition step from the MRI data, see [16] such that the final image does not show these artifacts (which would be our method of choice). Otherwise, we have to perform one of the above-mentioned preprocessings or to use another regularizer as, e.g., in [33]. In the present work, we apply this TV regularizer on membership functions to incorporate the spatial information and boundary connection in the MRI images, and a regularization parameter $\mu$ is introduced to control the trade-off between smoothing and clustering. Our motivation is to make the segmentation regions and boundaries well connected.

The structure of this paper is as follows: after introducing the general notation in Section 2, the original FCM method is reviewed in Section 3. Since we need it for our numerical comparisons, the moving-average FCM algorithm is briefly explained in Section 4. The new TVFCM model with a corresponding minimization algorithm is presented in Section 5. Finally, Section 6 illustrates the performance of the algorithm on noisy images and missing data images, where both synthetic and real-world images are involved. Conclusions are given in Section 7.
and 
\[
D_N := \begin{pmatrix}
-1 & 1 \\
-1 & 1 \\
\vdots & \vdots \\
-1 & 1 \\
0 & 0 & \cdots & 0
\end{pmatrix} \in \mathbb{R}^{N \times N}
\]
denotes the forward difference matrix which takes the mirrored (Neumann) boundary conditions into account with a zero row. Note that the discrete TV functional can be rewritten with the \( \ell_1 \) norm as
\[
TV(f) := ||\nabla f||_{l^1}, \quad |\nabla f| := (|\nabla f|)^2_{l^1}.
\]
Now let us turn to our segmentation task. Let \( c \in \mathbb{N}, c < n \) denote the number of segments or clusters labeled by \( Q = \{1, \ldots, c\} \). Segmentation aims to find segment prototypes \( v := \{v_1, v_2, \ldots, v_c\} \) together with a labeling function \( l : \Gamma \rightarrow Q \). Although this problem can be tackled directly, see, e.g. [35] and the references therein, the problem relaxes if we instead of the labeling function a labeling vector \( u : \Gamma \rightarrow [0, 1]^c \) which assigns to each image point \( j \in \Gamma \) a c-dimensional vector \( u(j) := (u_k(j))_{k=1}^c \), with
\[
\sum_{k=1}^c u_k(j) = 1 \quad \text{and} \quad u_k(j) \geq 0 \quad \forall k = 1, \ldots, c,
\]
and we apply
\[
l(j) := \text{argmax}_{k=1, \ldots, c} u_k(j),
\]
as the labeling function. In the following, we also use the notation \( u_k := (u_k(j))_{j=1}^n \) and \( u := (u_k)_{k=1}^c \in \mathbb{R}^n \).

3. The FCM algorithm

The FCM algorithm can be considered as an optimization problem with weighted dissimilarity terms constituting the objective function, which is defined for \( m > 1 \) as
\[
f(u, v) = \sum_{j=1}^n \sum_{k=1}^c u_k(j) ||f(j) - v_k||^p + \sum_{k=1}^c TV(u_k), \tag{4}
\]
\[
\sum_{k=1}^c u_k(j) = 1 \quad \forall j, \quad u_k(j) \geq 0 \quad \forall j, \quad \forall k = 1, \ldots, c.
\tag{5}
\]
The original FCM method with \( f(j) - v_k \) and \( p = 2 \) was proposed in [4]. For \( p = 1 \), which results in weighted median computations, we refer to [36], for \( p > 1 \) to [37] and for \( 0 < p < 1 \) to [38]. Using the Lagrange multiplier method, we may obtain a local extremum for \( f \) (local minimum or saddle point) if we update \( u \) and \( v \) alternatingly according to the algorithm
\[
u_k^{(i+1)} := \sum_{j=1}^n \frac{u_k^{(i)}(j) f(j)}{\sum_{j=1}^n u_k^{(i)}(j)} f(j),
\]
\[
u_k^{(i+1)}(j) := \left( \sum_{j=1}^n \frac{(f(j) - v_k^{(i+1)}(j))^2}{(f(j) - v_k^{(i)}(j))^2} \right)^{-1},
\]
where the update of \( u \) is done in the above way if \( f(j) - v_k^{(i+1)} \neq 0 \) for all \( k = 1, \ldots, c \). If there exists \( k_0 \in \{1, \ldots, c\} \) with \( f(j) - v_{k_0}^{(i+1)} = 0 \), then any nonnegative update \( u_k^{(i+1)}(j), k = 1, \ldots, c \) fulfilling the first constraint in (5) and \( u_k^{(i+1)}(j) = 0 \) if \( f(j) - v_{k_0}^{(i+1)} \neq 0 \) can be chosen.

It was proved in [39] that there exists a subsequence of \((u^{(i)}, v^{(i)})_{i=1}^\infty\) which converges to a local minimizer or a saddle point of \( f \) if \( f \) contains at least \( c \) different gray values.

4. The moving-average FCM algorithm

A moving-average filter was introduced by Hou et al. [15] as the regularizer that measures the local variation of the membership function. We include it briefly for comparison as it is effectively similar to the first order derivative constraint. The approach aims to minimize
\[
f(u, v) = \sum_{j=1}^n \sum_{k=1}^c u_k^2(j) f(j) - v_k^2 \beta \sum_{j=1}^n (u_k(j) - \Pi_k(j))^2,
\]
where \( \beta > 0 \) is the regularization parameter and \( \Pi_k(j) \) is the average of the fuzzy membership of the pixels in the neighborhood. Solving the optimization problem by the Lagrange multiplier method, the same updating equation for \( v \) as in (4) is obtained since the regularizer is not related to \( v \). The updating equation for \( u \) in step (i) becomes
\[
u_k^{(i+1)}(j) := \frac{1 - \sum_{j=1}^n \frac{|f(j) - v_k^{(i)}(j)|^p}{|f(j) - v_k^{(i+1)}(j)|^p} + \beta \sum_{j=1}^n \left( \frac{1}{|f(j) - v_k^{(i)}(j)|^p} + \frac{1}{|f(j) - v_k^{(i+1)}(j)|^p} \right)^{-1}}{\sum_{j=1}^n \frac{|f(j) - v_k^{(i)}(j)|^p}{|f(j) - v_k^{(i+1)}(j)|^p} + \beta \sum_{j=1}^n \left( \frac{1}{|f(j) - v_k^{(i)}(j)|^p} + \frac{1}{|f(j) - v_k^{(i+1)}(j)|^p} \right)^{-1}}.
\]
where \( v_k := v_k^{(i+1)} \). If \( \beta = 0 \), Hou's method reduces to the original FCM method with \( m=2 \). If \( \beta \) is sufficiently large, the updated membership function is dominated by its neighbors; thereby oversmoothed or blurred images occur.

5. The TVFCM algorithm

5.1. The TVFCM model

To make the FCM algorithm more robust with respect to noise and missing data, we suggest to minimize instead of the FCM functional the following TV regularized functional:
\[
f(u, v) = \mu \sum_{j=1}^n \sum_{k=1}^c (u_k(j) f(j) - v_k)^2 + \sum_{k=1}^c TV(u_k), \tag{7}
\]
\[
\sum_{k=1}^c u_k(j) = 1 \quad \forall j, \quad u_k(j) \geq 0,
\]
\[
\sum_{k=1}^c \sum_{j=1}^n (\nabla u_k(j))^2 + \cdots + (\nabla v_k(j))^2)^{1/2}
\]
as suggested in [28] could be used. We have also implemented this version, but have not seen advantages for our example images. We like to emphasize that \( TV(u_k) \) is the same as applying the discrete total variation to the \( c \) reshaped matrices
\[
U_k := \begin{pmatrix}
u_{k,1} & \ldots & \nu_{k,n+1} \\
\vdots & \ddots & \vdots \\
\nu_{k,2n} & \ldots & \nu_{k,3n}
\end{pmatrix}
\]
Our TVFCM functional (7) is bi-convex meaning that it is convex in \( v \) for fixed \( u \) and conversely. We propose to find a local minimizer of (7) by alternating the minimization with respect to \( v \) while fixing \( u \) and one iteration step of an ADMM algorithm applied to the TVFCM functional with respect to \( u \) while fixing \( v \). The ADMM algorithm which solves (7) with respect to \( u \) for fixed \( v \) is explained in the next subsection. In contrast to the FCM approach, where \( m > 1 \) is required, we can also use \( m=1 \) here which gives the usual c-means algorithm, see [40] and then the TVCM algorithm. In the following, we focus on the case \( m=1 \).
since in this case the ADMM updating step for \( u \) will be very fast. As described after the following remark, the approach is possible for arbitrary \( m \geq 1 \) so that we retain the notation TVFCM also for the case \( m = 1 \). In summary, our TVFCM algorithm for \( m = 1 \) reads as follows:

**TVFCM Algorithm:** Initialization: nonnegative vectors \( u^{(0)}_1 \in \mathbb{R}^{cn} \) with \( \sum_{j=1}^n u^{(0)}_k(j) = 1, \; k = 1, \ldots, n \). \( d^{(0)} \in \mathbb{R}^{2cn}, \; b^{(0)}_0 \in \mathbb{R}^m, \; b^{(0)}_x \in \mathbb{R}^{2cn} \) and \( \gamma > 0 \).

For \( i = 0, 1, \ldots \) repeat until a convergence criterion is reached

1. For \( k = 1, \ldots, c \), compute
   \[
   u^{(i+1)}_k = \sum_{j=1}^n \frac{u^{(i)}_k(j)f(j)}{\sum_{j=1}^n u^{(i)}_k(j)}.
   \]

2. For \( k = 1, \ldots, c \), set \( x^{(i+1)}_k = \left( (f(j) - d^{(i+1)})(j) \right)_{j=1}^n \) and find \( w^{(i+1)}_k \) as the solution \( x \) of the linear system
   \[
   (\nabla^2 + I_n)x = \nabla^2 u^{(i)}_k(j) + (w^{(i)}_k(j) - d^{(i+1)} - \mu^2)_{j=1}^n
   \]
   by using the DCT-II, see next remark.

3. For \( k = 1, \ldots, c \), set \( g_k = b^{(i+1)}_k + \nabla w^{(i+1)}_k \) and perform for \( j = 1, \ldots, n \) the coupled shrinkage
   \[
   d^{(i+1)}_{k,j} = \begin{cases} 0 & \text{if } |g_k(j)| \leq \gamma, \\ g_k(j) \left( 1 - \gamma |g_k(j)| \right) & \text{otherwise}. \end{cases}
   \]

4. For \( j = 1, \ldots, n \), set \( h(j) = b^{(i+1)}(j) + w^{(i+1)}(j) \in \mathbb{R}^c \) and compute \( u^{(i+1)}_j \) as the orthogonal projection of \( h(j) \) onto the orthogonal \( \nabla w^{(i+1)}_k \) in steps 3 and 5 must only be computed once.

**Remark.** Consider the matrix
\[
\nabla^2 + I_n = D_{n}^T D_n + D_{n}^T D_n \otimes I_n + I_n.
\]

The matrix \( D_{n}^T D_n \) can be diagonalized by the DCT-II matrix \( C_n \), i.e.,
\[
D_{n}^T D_n = C_n^T \text{diag}(q_n) C_n, \quad q_n = \left( 4 \sin^2 \frac{2j}{2n} \right)_{j=0}^{n-1},
\]

with the orthonormal DCT-II matrix
\[
C_n = \sqrt{\frac{2}{n}} \left( e^{j2k+1} \right)^{n-1} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix},
\]

where \( \varepsilon_j = 1/\sqrt{2} \) if \( j = 0 \) and \( \varepsilon_j = 1 \) otherwise, see [41,42]. The multiplication with the DCT-II matrix can be realized by the FFT with only \( O(n \log n) \) arithmetic operations. Now straightforward computation gives
\[
\nabla^2 + I_n = (C_n^T \otimes C_n)(A + I_n)(C_n \otimes C_n),
\]

where \( A := I_n \otimes \text{diag}(q_n) + \text{diag}(q_n) \otimes I_n \). Consequently, we obtain
\[
(A + I_n)^{-1} = (C_n^T \otimes C_n)(A + I_n)^{-1}(C_n \otimes C_n).
\]

Therefore step 2 in our TVFCM algorithm can be realized in a fast way. We like to emphasize that we do not work with tensor products in our numerical computations, but with matrix–matrix operations based on (2).

If we want to avoid the numerical solution of a linear system of equations in step 2, the so-called primal dual hybrid gradient method (PDHG) and its relatives [43–45] could be used as an alternative to the ADMM algorithm. This would, in particular, make sense if we want to use \( m > 1 \) in our TVFCM functional as in the usual FCM algorithm since, in this case, the linear system cannot be solved by using the DCT.

Finally, we want to mention that in all our experiments there was an \( \varepsilon > 0 \) such that \( \sum_{k=1}^n u^{(i)}_k(j) \geq \varepsilon \) for all \( k = 1, \ldots, c \). In case this is not fulfilled for some \( k \), we suggest that the number of labels be reduced by one and the algorithm restarted.

5.2. The ADMM within TVFCM

The ADMM can be used to solve constrained minimization problems of the form
\[
\min_{x \in \mathbb{R}^n} \{ G_1(x) + G_2(y) \} \text{ subject to } Ax = y, \quad (8)
\]

with proper, closed, convex functions \( G_1 : \mathbb{R}^M \rightarrow \mathbb{R} \cup \{ + \infty \} \), \( G_2 : \mathbb{R}^l \rightarrow \mathbb{R} \cup \{ + \infty \} \) and \( A \in \mathbb{R}^{M \times N} \). The ADMM splits the problem into the following smaller subproblems which have to be solved alternatingly:

**General ADMM:** Initialization: \( y^{(0)}, b^{(0)} \in \mathbb{R}^l \) and \( \gamma > 0 \).

For \( i = 0, \ldots \) repeat until a convergence criterion is reached:
\[
x^{(i+1)} = \arg\min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2\gamma} b^{(i)}(x - Ax) + \frac{1}{2} ||y^{(i+1)} - y^{(i)}||_2^2 \right\}, \quad (9)
\]

\[
y^{(i+1)} = \arg\min_{y \in \mathbb{R}^l} \left\{ \frac{1}{2\gamma} b^{(i+1)}(y - Ax^{(i+1)}) + \frac{1}{2} ||y^{(i+1)} - y^{(i)}||_2^2 \right\},
\]

\[
b^{(i+1)} = b^{(i)}(x - Ax^{(i+1)}) - y^{(i+1)}.
\]

For the above problem (8), the ADMM coincides with the alternating split Bregman method [46] and with the Douglas–Rachford splitting method applied to the dual problem of (8), see [47–50]. For any starting values and any \( \gamma > 0 \), the ADMM sequence \( (b^{(i)}) \) converges to some \( b \), where \( (1/\gamma)b \) is a solution of the dual problem of (8) and the sequence \( (x^{(i)}) \) converges to a solution \( x \) of (8) if (9) has a unique solution, see, e.g., [50]. Moreover, \( Ax = y \) holds true.

To apply the general ADMM to our segmentation setting, we rewrite our TVFCM functional in (7) with \( s_k = (f(j) - u_k^j)^2 \)

\[
J(u,v) = \mu \sum_{k=1}^N \left\{ \langle u_k, s_k \rangle + \sum_{k=1}^n \| \nabla u_k \|_1 \right\},
\]

and the whole problem with \( w = (w_k)^N_{k=1} \in \mathbb{R}^n \) and \( d = (d_k)^N_{k=1} \in \mathbb{R}^{2cn} \) as

\[
\min_{w \in \mathbb{R}^n, d \in \mathbb{R}^{2cn}} \left\{ \mu \sum_{k=1}^N \left\langle w_k, s_k \right\rangle + \sum_{k=1}^n \| d_k \|_1 + \sum_{j=1}^n \| s(j) \|_1 \right\},
\]

subject to \( \nabla w_k = d_k \), \( w_k = u_k \),

with the indicator function \( I_S \) of \( S \) defined by
\[
I_S(x) = \begin{cases} 0 & \text{if } x \in S, \\ \infty & \text{otherwise,} \end{cases}
\]

\[
S = \left\{ x \in \mathbb{R}^c : \sum_{k=1}^c x_k = 1, x \geq 0 \right\}. \quad (10)
\]
We use
\[
G_1(w) = \mu \sum_{k=1}^{C} \langle w_k, s_k \rangle = \mu \langle w, s \rangle ,
\]
\[
G_2(d, u) = \sum_{k=1}^{C} \left\| d_k \right\|_1 + \sum_{j=1}^{n} t_j(u(j))
\]
and
\[
Aw = \begin{pmatrix} d \\ u \end{pmatrix}, \quad A = \begin{pmatrix} I_c \otimes \nabla \\ I_n \end{pmatrix}.
\]
in the general ADMM and obtain the following algorithm for our problem:

**ADMM for (7) with fixed segment prototypes:**

Initialization: \(d_0, u_0 \in \mathbb{R}^{2cn}, b_d(0) \in \mathbb{R}^{2cn}, b_u(0) \in \mathbb{R}^{2n}, \) and \(\gamma > 0\).

For \(i = 0, \ldots\), repeat until a convergence criterion is reached:

\[
w^{i+1} = \begin{pmatrix} w_0 \\ v_0 \end{pmatrix} \text{argmin}_{w \in \mathbb{R}^m} \left\{ G_1(w) + \frac{1}{2\gamma} \left\| \begin{pmatrix} b_d(0) \\ b_u(0) \end{pmatrix} - Aw \right\|_2^2 \right\},
\]

\[
d^{i+1} = \begin{pmatrix} d_0 \\ u_0 \end{pmatrix} \text{argmin}_{d, u} \left\{ G_2(d, u) + \frac{1}{2\gamma} \left\| \begin{pmatrix} b_d(0) \\ b_u(0) \end{pmatrix} + Aw^{i+1} - d \right\|_2^2 \right\},
\]

\[
b^{i+1} = \begin{pmatrix} b_d(i) \\ b_u(i) \end{pmatrix} + Aw^{i+1} - \begin{pmatrix} d^{i+1} \\ u^{i+1} \end{pmatrix}.
\]

We have to comment on the first two steps of the algorithm: Note that, in the second step, we can minimize separately over \(d\) and \(u\).

**Computation of \(w^{i+1}\):** Due to the differentiability of the functional in the first step of the above algorithm, its minimizer can be obtained by setting the gradient of the functional to zero. Then the minimizer is given by the solution of the linear system of equations
\[
A^T A w = A^T \left( \begin{pmatrix} d(0) \\ u(0) \end{pmatrix} - \begin{pmatrix} b_d(0) \\ b_u(0) \end{pmatrix} \right) - \mu \gamma s.
\]

Now
\[
A^T A = (I_c \otimes \nabla^T, I_n) \left( I_c \otimes \nabla \right) = (I_c \otimes \nabla^T \nabla) + I_n.
\]

This results in the second step of our TVFCM algorithm.

**Computation of \(d^{i+1}\):** The minimization over \(d = (d_k)_{k=1}^{C}\) can be performed separately for every \(k = 1, \ldots, C\) by solving
\[
d_k^{i+1} = \arg\min_{d_k \in \mathbb{R}^{2cn}} \left\{ \left\| d_k \right\|_1 + \frac{1}{2\gamma} \left\| d_k - (B_{d,k}^{(0)} + \nabla w_k^{i+1}) \right\|_2^2 \right\}.
\]

The minimizer \(d_k^{i+1} = (d_k^{i+1}(j))_{j=1}^{2}, \) \(d_k^{i+1}(j) \in \mathbb{R}^2, \) is given by the so-called coupled shrinkage of \(g_k\) with threshold \(\gamma\), see, e.g., [50], defined componentwise by
\[
d_k^{i+1}(j) = \begin{cases} \frac{B_{d,k}^{(0)}}{\gamma}, & \text{if } |g_k(j)| \leq \gamma, \\ g_k(j) \left(1 - \frac{\gamma}{|g_k(j)|}\right), & \text{otherwise}. \end{cases}
\]

**Computation of \(u^{i+1}\):** Finally, we are interested in the solution of the subproblem
\[
u^{i+1} = \arg\min_{u \in \mathbb{R}^n} \left\{ \sum_{j=1}^{n} t_j(u(j)) + \frac{1}{2\gamma} \left\| u - (b_u^{(0)} + w^{i+1}) \right\|_2^2 \right\}.
\]

This problem can be solved separately for every \(j = 1, \ldots, n\), i.e., with \(h(j) = b_u^{(0)} + w^{i+1}(j)\). We are looking for
\[
u^{i+1}(j) = \arg\min_{u(j) \in \mathbb{R}} \left\{ t_j(u(j)) + \frac{1}{2\gamma} \left(u(j) - h(j)\right)^2 \right\}.
\]

In other words, we have to find the orthogonal projection of \(h(j)\) onto \(S\) which can be done, for example, as in [51].

6. Numerical results

The qualitative and quantitative comparison of the effectiveness of the proposed TVFCM algorithm versus the original FCM algorithm and the moving-average FCM algorithm is provided in the case of synthetic and real images. The segmentation accuracy (SA) is defined as
\[
\text{SA} = \frac{\text{#correctly classified pixels}}{\text{#all pixels}},
\]
and the segmentation accuracy of the target pixels (target accuracy) is defined as
\[
\text{TA} = \frac{\text{#correctly classified target pixels}}{\text{#all target pixels}}.
\]

Images corrupted by white Gaussian noise and images with incomplete data are used to test the robustness of the algorithms. Note that the images were normalized before the computation to have intensities within \([0,1]\). The parameters of the moving average FCM and the TVFCM are chosen empirically according to the best results. The regularization parameter \(\mu\) in our TVFCM algorithm was chosen by hand to get the best SA for the simulated data and the best visual quality for the real-world data. An automatic adaptation of the regularization parameter, e.g., along the lines of [23] is beyond the scope of this paper.

6.1. Synthetic two-strip image

The first synthetic test image of 128 \(\times\) 128 pixels contains two strips with intensity 0 and 90 (Fig. 1). It is corrupted by white Gaussian noise with the standard deviation \(\sigma = 20\) (noise level: 22\%) as shown in Fig. 1a. The FCM, moving-average FCM and TVFCM are applied for segmenting this image and the different segments of the image are shown in the same figure with different gray values. The SA of the FCM method is 0.9874 (Fig. 1b), which

![Image 1](image1.png)

**Fig. 1.** Segmentation of noisy two strip image. (a) The image corrupted by white Gaussian noise with \(\sigma = 20\). (b) FCM segmented image. (c) Moving-average FCM segmented image with \(\beta = 0.2\). (d) TVFCM segmented image with \(\mu = 0.3, \gamma = 1\).

![Image 2](image2.png)

**Fig. 2.** Segmentation of image with incomplete data. (a) Two strips with 20\% missing data. (b) FCM segmented image. (c) Moving-average FCM segmented image with \(\beta = 3.5\). (d) TVFCM segmented image with \(\mu = 0.3, \gamma = 1\).
shows that FCM is sensitive to noise. The moving-average FCM [15] and our TVFCM have $SA = 1$, i.e., every pixel is classified correctly so that the algorithms turn out to be robust to Gaussian noise. Both the moving-average FCM and TVFCM take one iteration to attain $SA = 1$ if the FCM results were used as initial values. Note that for the two-strip image $SA = TA$.

In Fig. 2, we test the methods for the two-strip image (with intensity 100 and 200) with incomplete data by randomly assigning 20% of the pixels zero intensity (Fig. 2a). Fig. 2b shows that the FCM algorithm cannot recover the missing data completely and it has $SA = 0.9092$. Since the moving-average FCM takes spatial information into account, it recovers most of the missing data. However, it does not preserve the sharp edges (boundaries) shown in Fig. 2c. The segmentation accuracy for the moving-average FCM is $SA = 0.9941$. Fig. 2d shows that the TVFCM method can totally recover the missing data. We further remark that by ADMM with fixed segment prototypes, the two stripes are also recovered perfectly but with about 110 iterations where as our TVFCM needed only 30 iterations. In this experiment, the result of the FCM is used as the initial data for both the moving-average FCM and TVFCM. Note that the moving-averaged FCM is iterated until numerical convergence; the SA result can be slightly better if we stop at an earlier iteration step.

A comparison of the segmentation accuracy of FCM, moving-average FCM and TVFCM on the two-strip image with different levels of Gaussian noise and on the two-strip image with different levels of incomplete data is presented in Table 1. As expected, a decrease in the accuracy of both FCM and moving-average FCM occurs when the level of noise and incomplete data increases. However, TVFCM recovered the two strips completely from the

Table 1
A comparison of segmentation accuracy of FCM, moving-average FCM, and the proposed TVFCM on the two-stripe image with different levels of noise and incomplete data.

<table>
<thead>
<tr>
<th>Noise/missing level</th>
<th>FCM</th>
<th>Moving-average FCM</th>
<th>TVFCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian noise 33%</td>
<td>0.9358</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Gaussian noise 44%</td>
<td>0.8704</td>
<td>0.9996</td>
<td>1</td>
</tr>
<tr>
<td>30% data missing</td>
<td>0.8692</td>
<td>0.9894</td>
<td>1</td>
</tr>
<tr>
<td>40% data missing</td>
<td>0.8365</td>
<td>0.9908</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 3. Segmentation results on annulus. (a) Original image. (b) The same image with 20% data and a line missing. (c) Mask defining the target for the computation of $TA$. (d) FCM result. (e) Moving-average FCM, $\beta = 3$. (f) TV-preprocessed FCM. (g) ADMM with fixed segment prototypes, $\mu = 1.35$, $\gamma = 1$. (h) TVFCM, $\mu = 1.35$, $\gamma = 1$.

Fig. 4. Data term, TV term, objective function in the iteration procedure (the first row) and segmentations results (the second row) for annulus with incomplete data ($\gamma = 1$). The input image is shown in Fig. 3b. (a) $\mu = 1$. (b) $\mu = 10$. The horizontal coordinates in the first row denote the index of iterations.
noisy image with 33% Gaussian noise ($\sigma = 40$) and with 40% missing data.

6.2. Synthetic image with annulus

The second synthetic test image of size $128 \times 128$ is composed of an annulus of intensity 200 while the rest of the area has an intensity of 100 (Fig. 3a). The image with incomplete data (Fig. 3b) is generated by randomly assigning 20% of the pixels and a radial line zero intensity. This test image is treated by FCM, moving-average FCM, TV-preprocessed FCM, ADMM with fixed segment prototypes and TVFCM methods. To create a mask defining the target for the computation of $TA$, we set all foreground pixels of the annulus (Fig. 3a) to 1 and all background pixels to 0. Then we applied a morphological dilation with a square structuring element of size 4 with the MATLAB command `imdilate(binary_annulus, strel('square', 4))` such that also errors near the boundary but outside the original annulus are taken into account by the $TA$. Fig. 3d shows that the FCM algorithm does not recover the annulus well ($SA = 0.9461$, $TA = 0.8333$). Using the FCM results as the initial solution for the moving-average FCM gives the segmentation accuracy $SA = 0.9935$ and $TA = 0.9883$ (Fig. 3e). It is also interesting to compare TVFCM with the result of TV-preprocessing (using ROF denoising followed by FCM). For the two-strip image, TV-preprocessing FCM recovers the two strips as completely as TVFCM. In the case of the annulus, Fig. 3f shows that TV-preprocessing makes the edges smoother; however, the segmentation with $SA = 0.9966$ and $TA = 0.9894$ is slightly worse compared to $SA = 0.9968$ and $TA = 0.9900$ for TVFCM with parameters $\mu = 1.35$, $\gamma = 1$ (Fig. 3h). For the example of the annulus shown in Fig. 3g, the ADMM with fixed segment prototypes ($SA = 0.9914$, $TA = 0.9734$) was not able to achieve the accuracy of the TVFCM.

Finally, we are interested in the role of the parameter $\mu$. For small $\mu$, the TV term initially dominates the minimization procedure. However, it decreases as the iterations increase (Fig. 4a). This results in a too smooth segmentation with a closed inner hole. For large $\mu$, the dominating data term decreases (Fig. 4b), so that the segmentation result is still noisy.

6.3. Phantom data

The segmentation and target accuracy of FCM, moving-average FCM and TVFCM on phantom data (simulated MR images) with different levels of noise from the Brain web database [52] is shown in Table 2. The mask defining the target for the computation of $TA$ is created in the same way as in the annulus example above (Fig. 5g). The phantom data of size $181 \times 217 \times 181$ ($X \times Y \times Z$) was simulated with T1 modality, 1 mm slice thickness, different levels of noise and uniform intensity. One slice in the $XY$-plane with $Z = 90$ (Fig. 5a) is selected to test the robustness of the algorithms. Fig. 5b is the slice of noisy phantom with only four clusters (white matter, grey matter, cerebrospinal fluid and background). The true classification is shown in Fig. 5c. The segmentation results of these algorithms in Fig. 5b is presented in Fig. 5d–g.

The results in Table 2 and Fig. 6 show the target and/or segmentation accuracy for different noise levels. Note that, in Fig. 6, every second noise-level between 1 and 49 ($1:2:49$) is plotted while, due to space limitations, in Table 2 only every second noise-level between 1 and 49 ($1:2:49$) is listed. For the first noise levels, the plot in Fig. 6 is decreasing as expected. At the higher noise levels, there appear some increasing steps for all three algorithms. This can happen due to the randomness of the highly disturbing noise and the fact that the algorithms only find

---

**Table 2**

Comparisons of the segmentation and target accuracy of FCM, moving-average FCM, and TVFCM on phantom data with different levels of noise.

<table>
<thead>
<tr>
<th>Noise level (%)</th>
<th>FCM $SA$</th>
<th>FCM $TA$</th>
<th>MoAve $SA$</th>
<th>MoAve $TA$</th>
<th>TVFCM $SA$</th>
<th>TVFCM $TA$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.9849</td>
<td>0.9712</td>
<td>0.9855</td>
<td>0.9724</td>
<td>0.9859</td>
<td>0.9731</td>
</tr>
<tr>
<td>7</td>
<td>0.9539</td>
<td>0.9159</td>
<td>0.9708</td>
<td>0.9445</td>
<td>0.9719</td>
<td>0.9465</td>
</tr>
<tr>
<td>11</td>
<td>0.8826</td>
<td>0.7764</td>
<td>0.9522</td>
<td>0.9089</td>
<td>0.9596</td>
<td>0.9231</td>
</tr>
<tr>
<td>15</td>
<td>0.8180</td>
<td>0.6533</td>
<td>0.8886</td>
<td>0.7878</td>
<td>0.9465</td>
<td>0.9881</td>
</tr>
<tr>
<td>19</td>
<td>0.7740</td>
<td>0.5694</td>
<td>0.8375</td>
<td>0.6905</td>
<td>0.9329</td>
<td>0.8722</td>
</tr>
<tr>
<td>23</td>
<td>0.7505</td>
<td>0.5247</td>
<td>0.8084</td>
<td>0.6349</td>
<td>0.9217</td>
<td>0.8509</td>
</tr>
<tr>
<td>27</td>
<td>0.7270</td>
<td>0.4800</td>
<td>0.7882</td>
<td>0.5966</td>
<td>0.9021</td>
<td>0.8136</td>
</tr>
<tr>
<td>31</td>
<td>0.7158</td>
<td>0.4587</td>
<td>0.7735</td>
<td>0.5684</td>
<td>0.8851</td>
<td>0.7811</td>
</tr>
<tr>
<td>35</td>
<td>0.7019</td>
<td>0.4320</td>
<td>0.7710</td>
<td>0.5638</td>
<td>0.8572</td>
<td>0.7280</td>
</tr>
<tr>
<td>39</td>
<td>0.6989</td>
<td>0.4263</td>
<td>0.7641</td>
<td>0.5506</td>
<td>0.8554</td>
<td>0.7245</td>
</tr>
<tr>
<td>43</td>
<td>0.6872</td>
<td>0.4042</td>
<td>0.7502</td>
<td>0.5242</td>
<td>0.8513</td>
<td>0.7168</td>
</tr>
<tr>
<td>47</td>
<td>0.6821</td>
<td>0.3943</td>
<td>0.7447</td>
<td>0.5136</td>
<td>0.8185</td>
<td>0.6543</td>
</tr>
</tbody>
</table>

![Fig. 5](image-url) Segmentation results on phantom data with 3% noise. (a) One slice of phantom with 3% noise. (b) One slice of phantom with 3% noise only four clusters. (c) Mask defining the target for the computation of $TA$. (d) True classification. (e) FCM result. (f) Moving-average FCM, $\beta = 0.01$. (g) TVFCM, $\mu = 290$, $\gamma = 0.1$. 
The results become smoother with decreasing the segments have smoother boundaries. The segmented result in Fig. 7d–f. The TVFCM results have less speckling and shown in Fig. 7c and from TVFCM with FCM results as initial FCM results (Fig. 7b), the result of the moving average FCM is Fig. 7ft o.

local optima. However, best segmentation results are obtained by the TVFCM on the phantom data at all the noise levels.

6.4. Real MRI image

Fig. 7 shows the application of TVFCM to a real functional MRI image [53]. The real image is presented in Fig. 7a. Compared to FCM results (Fig. 7b), the result of the moving average FCM is shown in Fig. 7c and from TVFCM with FCM results as initial condition in Fig. 7d–f. The TVFCM results have less speckling and the segments have smoother boundaries. The segmented result in Fig. 7d corresponds to TVFCM with the parameters $\mu = 120$, $\gamma = 0.1$. Fig. 7e to parameters $\mu = 60$, $\gamma = 0.1$ and Fig. 7f to parameters $\mu = 20$, $\gamma = 0.1$. The boundaries of the segments in the results become smoother with decreasing $\mu$.

7. Summary

This paper combines the idea of the FCM approach with those of TV-regularized multi-class labeling within one algorithm. It updates the segment prototypes and the labels in an alternating way by performing an averaging step to obtain the segment centroids and one step of an ADMM to improve the label matrix.

The test images demonstrate that the proposed segmentation approach—TVFCM works properly on gradient-sparse images.

References

Yanyan He has been a PhD student since 2007 at the Department of Mathematics, Florida State University, USA.

M. Yousuff Hussaini received the Ph.D. degree in engineering from the University of California, Berkeley, in 1970. He is currently the Sir James Lighthill Professor of mathematics and computational science and engineering and the TMC Eminent Scholar Chair in High Performance Computing with the Florida State University (FSU), Tallahassee. Before coming to FSU, he was the Director of the Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA. His research interests include numerical algorithms for solving partial differential equations and parallel computation on high-performance computers; computational fluid dynamics; nanomaterials and nanofluids; optimal control and optimization under uncertainty; uncertainty quantification; and compressed sensing, imaging, and visualization.

Jianwei Ma received the Ph.D. degree from Tsinghua University, Beijing, China, in 2002. He was a research fellow and visiting scientist at the University of Cambridge, University of Grenoble, EPFL, Ecole des Mines de Paris, and Florida State University. He was an associate professor with the School of Aerospace, Tsinghua University, China. Since July 2011, he has been a full professor and director of the Institute of Applied Mathematics, Harbin Institute of Technology, China. His research interests include sparse transform, image processing, compressed sensing, remote sensing and seismic exploration.

Behrang Shafei is a Ph.D. student at Image Processing group of the Fraunhofer ITWM, Kaiserslautern, Germany.

Gabriele Steidl received her Ph.D. and Habilitation in Mathematics from the University of Rostock, Germany, in 1988 and 1991, respectively. From 1993 to 1996, she held a position as assistant professor of Applied Mathematics at the University of Darmstadt, Germany. From 1996 to 2010, she was full professor at the Department of Mathematics and Computer Science, University of Mannheim, Germany. Since 2011, she is Professor at the Department of Mathematics at the University of Kaiserslautern, Germany. She was a visiting researcher at the University of Debrecen, Hungary, the University of Zu¨rich, Switzerland, the Banach Center Warszaw, Poland, the ENS Cachan and the University Paris-East, France. Her research interests include numerical analysis and applications in image processing.