Image Segmentation Based on Adaptive Cluster Prototype Estimation

Alan Wee-Chung Liew, *Member, IEEE*, Hong Yan, *Senior Member, IEEE*, and N. F. Law, *Member, IEEE*

***Abstract*—An image segmentation algorithm based on adaptive fuzzy c-means (FCM) clustering is presented in this paper. In the conventional FCM clustering algorithm, cluster assignment is based solely on the distribution of pixel attributes in the feature space, and does not take into consideration the spatial distribution of pixels in an image. By introducing a novel dissimilarity index in the modified FCM objective function, the new adaptive fuzzy clustering algorithm is capable of utilizing local contextual information to impose local spatial continuity, thus exploiting the high inter-pixel correlation inherent in most real-world images. The incorporation of local spatial continuity allows the suppression of noise and helps to resolve classification ambiguity. To account for smooth intensity variation within each homogenous region in an image, a multiplicative field is introduced to each of the fixed FCM cluster prototype. The multiplicative field effectively makes the fixed cluster prototype adaptive to slow smooth within-cluster intensity variation, and allows homogenous regions with slow smooth intensity variation to be segmented as a whole. Experimental results with synthetic and real color images have shown the effectiveness of the proposed algorithm.**

***Index Terms*—Fuzzy clustering, image segmentation, prototype adaptation, spatial continuity.**

I. Introduction

IMAGE segmentation is an important low-level preprocessing step for many computer vision problems [1], [2]. Many algorithms for image segmentation using fuzzy techniques have been proposed [3]–[6]. One popular technique involves using the fuzzy c-means (FCM) algorithm [7], or variants of it, to compute the membership values for different classes before the final segmentation. The FCM algorithm assigns membership value to a data sample based on its proximity to the cluster prototypes in the feature space. In the FCM-based segmentation algorithm, feature vectors are assumed to be independent of each other and independent of their spatial coordinates.

However, real-world images usually have strong correlation between neighboring pixels. Adjacent pixels in an object are generally not independent of each other. Thus, the incorporation of local spatial interaction between adjacent pixels in the fuzzy clustering process can produce more meaningful classification, as well as help to resolve classification ambiguities due to overlap in intensity value between clusters or noise corruption. In addition, the intensity of objects or regions in an image usually varies with image location, due to illumination and/or object geometry. Such objects or regions cannot be adequately represented by cluster prototypes of constant value.

A popular method to introduce local spatial context into pixel classification is relaxation labeling [8]–[10]. However, relaxation labeling requires the initial labeling probabilities of each pixel to be available. This requires a supervised classification of labeled pixels to obtain the initial statistics and to construct the compatibility coefficients between neighboring pixels for the chosen neighborhood configuration [8], [9]. Then, the labeling probabilities of each pixel are updated iteratively based on the compatibility coefficients. In [10], Hsiao proposed an unsupervised relaxation labeling algorithm for image segmentation. The method is essentially a two-stage process, i.e., unsupervised K-means clustering is used to estimate the initial labeling probabilities, followed by probabilistic relaxation to update the labeling probabilities. The important point to note in [8]–[10] is that the initial labeling on each pixel is based on the pixel attributes only, without considering the spatial correlation between neighboring pixels. This is also clearly brought out in [10], which states that "The weakness of classifying pixels based solely upon feature space distribution is that the formation of clusters in the feature space does not take into consideration the spatial distribution of points in image." The spatial correlation between neighboring pixels is only added later through the process of relaxation labeling. Therefore, relaxation labeling can be viewed as a *postprocessing* step that attempts to incorporate local spatial continuity into the final solution.

In contrast to postprocessing, there are several attempts to incorporate local spatial continuity directly into the clustering-based segmentation process. Tolias and Panas [11] proposed a fuzzy clustering-based segmentation algorithm that takes into account local spatial context and with spatially varying cluster prototypes. The cluster prototypes at a particular location are estimated from the pixels within a local window. The spatial context between neighboring pixels is taken into account by an ad-hoc modification of the membership values through the addition or subtraction of a small constant. In [12], [13], Liew *et al.* proposed a spatial FCM algorithm that incorporates local spatial context directly into the fuzzy objective functional formulation. However, there is no adaptation of the cluster prototypes to smooth intensity variation of homogenous regions.

Manuscript received October 7, 2001; revised September 29, 2003, July 7, 2004, and October 20, 2004. This work was supported by a strategic research grant (Project 7 001 556) from the City University of Hong Kong.

A. W.-C. Liew was with the Department of Computer Engineering and Information Technology, the City University of Hong Kong, Hong Kong. He is now with the Department of Computer Science and Engineering, The Chinese University of Hong Kong.

H. Yan is with the Department of Computer Engineering and Information Technology, the City University of Hong Kong, Hong Kong, and also with the School of Electrical and Information Engineering, University of Sydney, NSW 2006, Australia.

N. F. Law is with the center for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong.

Digital Object Identifier 10.1109/TFUZZ.2004.841748

In this paper, we proposed an adaptive FCM segmentation algorithm that: i) takes into account the high interpixel correlation, and ii) allows spatial adaptation of the cluster prototype to slow intensity variation in a homogenous region. For point one, our method incorporates the local spatial correlation between neighboring pixels *directly* into the clustering algorithm. In the estimation of the pixel's cluster memberships, not only the pixel's attributes but also the spatial relationship between it and its neighbors are taken into consideration. Although point one has appeared in our previous work [12], we provide a deeper analysis and interpretation of the idea in this paper. Some further improvements have also been made in this work. For example, the definition of the dissimilarity index given by (6) is now more general than the original formulation in [12], and while the steepness parameter σ was selected by experimentation in [12], it is determined here based on the image characteristic. For point two, although the idea of adapting to smooth intensity variation in an image has been adopted in [17], the goal there is to account for global intensity nonuniformity occurring over the entire image, which is a common artifact in MRI imaging [19]. The homogenous regions are still assumed to be of uniform intensity, i.e., the cluster prototypes in [17] are assumed to be constant. In this work, we are instead concerned with intensity variation within each homogeneous region, i.e., within each cluster. A further difference between this work and our work in [17] is the introduction of an additional relabeling step in the clustering algorithm due to the cluster prototypes adaptation.

This paper is divided into six sections. In Section II, we describe the image segmentation problem and point out the deficiencies in the conventional clustering-based approach. In Section III, we describe in detail how the local spatial continuity constraint and prototype adaptation can be incorporated into the FCM algorithm. Section IV describes some implementation issues and gives a pseudo-code description of the algorithm. Experimental results are given in Section V. Finally, Section VI draws some conclusions.

II. Image Segmentation by Fuzzy Cluster Analysis

The task of image segmentation can be stated as the partition of an image into a number of nonoverlapping regions, each with distinct properties. Using this definition, an image A can be modeled as the union of c homogenous regions A_k,

$$A = \cup_{k=1}^{c} A_k \tag{1}$$

where each homogenous region is specified by the representative properties $v_k(x, y)$, e.g., intensity, and an additive, zero mean random noise component $n_k(x, y)$, i.e., $A_k(x, y) = v_k(x, y) + n_k(x, y)$, for $(x, y) \in$ region A_k. Each homogenous region can consist of a single connected component or a group of possibly disjoint connected components of pixels with similar property.

Let $S = \{s(\underline{x})\}$, where $\underline{x} = (x, y)$, be the set of feature vectors, i.e., intensity value or color value associated with an image defined in the domain I. The conventional FCM algorithm [7] is formulated as the minimization of the objective functional J_{FCM} with respect to the membership values U and cluster prototypes v,

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

subject to

$$\sum_{k=1}^{c} u_{k,\underline{x}} = 1 \qquad \forall \underline{x} \in I$$

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

$$d_{k,\underline{x}}^{2} = \|s(\underline{x}) - v_k\|^2. \tag{3}$$

The FCM objective function is minimized when high membership values are assigned to pixels close to a cluster prototype, and low membership values are assigned when they are far from the prototype. After FCM clustering, each pixel will be associated with a membership value for each class. By assigning the pixel to the class with the highest membership value, a segmentation of the image can be obtained.

Although the membership allows a pixel to deviate from the cluster prototype, the FCM segmentation algorithm implicitly assumes that the cluster prototype of each class has a constant value. For the image model (1), this implies that $v_k(\underline{x}) = v_k$. Obviously, this assumption would only be valid on a very restricted set of images consisting of regions of constant intensities. Moreover, each pixel in the FCM algorithm is assumed to be independent of every other pixels and spatial correlation between adjacent pixels is not considered.

III. Proposed Methods

In view of the shortcomings of conventional FCM segmentation algorithm, we propose a modified FCM segmentation algorithm that explicitly: i) Takes into account the local spatial correlation between adjacent pixels, and ii) allows the cluster prototype to adapt to slow intensity variation in a homogenous region. Objective one is achieved by replacing the distance metric of (3) by a novel dissimilarity index as described in Section III-A. For objective two, a multiplicative field is added to each cluster prototype to model the intensity variation of each homogenous region. A relabeling procedure is also incorporated to merge adjacent regions that are homogenous. These are described in Sections III-B and III-C, respectively. Iterative updating of the variables to minimize the modified objective functional is given in Section III-D.

A. Local Spatial Continuity

The idea of incorporating local spatial context in FCM is to consider the influence of neighboring pixels on the pixel of interest during classification. Let $\aleph_\rho$ denote the chosen local neighborhood configuration with respect to a center pixel ρ. If

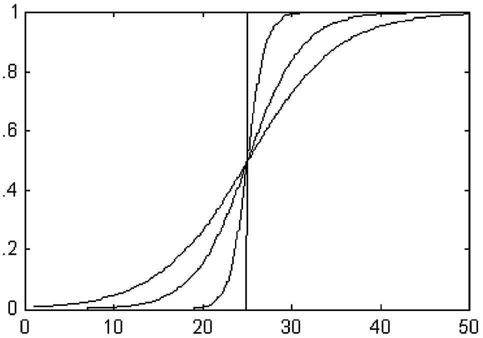


Fig. 1. Shape of the weighting factor λ centered at $\mu = 25$, with σ varied from 1, 3 to 5.

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

$$\partial_{\underline{x},\underline{y}} = \mathrm{dist}(s(\underline{x}), s(\underline{y})), \quad s(\underline{y}) \in \aleph_{\underline{x}} \tag{4}$$

$$d_{k,\underline{x}} = \mathrm{dist}(s(\underline{x}), v_k) \tag{5}$$

where $\aleph_{\underline{x}}$ is the neighborhood of $s(\underline{x})$ and v_k is the prototype of the kth cluster. The first distance metric $\partial_{\underline{x},\underline{y}}$ measures the dissimilarity of the pixel $s(\underline{x})$ and its neighbor $s(\underline{y})$. The second distance metric $d_{k,\underline{x}}$ measures the affinity of the center pixel $s(\underline{x})$ to the cluster prototype v_k. Now, if the distance $\partial_{\underline{x},\underline{y}}$ is small (i.e., $s(\underline{x})$ and $s(\underline{y})$ are similar), $d_{k,\underline{x}}$ should be greatly influenced by $d_{k,\underline{y}}$. Otherwise, $d_{k,\underline{x}}$ should be largely independent of $d_{k,\underline{y}}$. Taking all pixels in $\aleph_{\underline{x}}$ into account, we define a dissimilarity index $D_{k,\underline{x}}$ which measures the dissimilarity between $s(\underline{x})$ and the kth cluster prototype v_k, as

$$D_{k,\underline{x}} = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y} \in \aleph_{\underline{x}}} \left[d_{k,\underline{x}}^2 \lambda_{\underline{x},\underline{y}} + d_{k,\underline{y}}^2 (1 - \lambda_{\underline{x},\underline{y}}) \right] \tag{6}$$

where $|\aleph_{\underline{x}}|$ is the cardinality of the neighborhood configuration, and $\lambda(\partial_{\underline{x},\underline{y}}) = \lambda_{\underline{x},\underline{y}}, 0 \leq \lambda_{\underline{x},\underline{y}} \leq 1$, is the weighting factor controlling the degree of influence of the neighboring pixels $s(\underline{y}) \in \aleph_{\underline{x}}$ on the center pixel $s(\underline{x})$. One possible weighting factor is given by

$$\lambda(\partial) = \frac{1}{1 + e^{-(\partial - \mu)/\sigma}} \tag{7}$$

where the parameters μ and σ specify the displacement of λ from zero, and the steepness of λ, respectively. Fig. 1 shows the shape of the weighting factor λ centered at $\partial = \mu(\mu = 25)$, when σ is set to 1, 3, and 5. As σ increases, the transition becomes less steep.

The parameter μ can be viewed as the average "background randomness" of the homogeneous region with respect to the chosen neighborhood $\aleph_{\underline{x}}$. Its role can be easily understood by considering the asymptotic case when σ goes to zero. When $\sigma \rightarrow 0, \lambda$ approaches a two-state function, i.e., its value approaches zero or one, depending on whether $\partial < \mu$ or $\partial > \mu$. When $\partial < \mu$, i.e., the dissimilarity between $s(\underline{x})$ and $s(\underline{y})$ is below the random fluctuation of pixel value within a homogenous region, the cluster affinity of the center pixel $s(\underline{x})$ is almost completely determined by the cluster affinity of its neighboring pixel $s(\underline{y})$ as $\lambda \approx 0$. On the other hand, if $\partial > \mu$, the dissimilarity between $s(\underline{x})$ and $s(\underline{y})$ is significant such that the pixel $s(\underline{x})$ is unlikely to belong to the same homogenous region as that of pixel $s(\underline{y})$. In such situation, $\lambda \approx 1$ and the cluster affinity of pixel $s(\underline{x})$ is independent of that of pixel $s(\underline{y})$. Based on the above reasoning, and assuming that majority of $\aleph_{\underline{x}}$ in a typical image fall on homogeneous regions, a reasonable estimate for the parameter μ is

$$\mu = \frac{1}{|I|} \sum_{\underline{x} \in I} \partial_{\mathrm{av}}(\underline{x}) \tag{8}$$

where

$$\partial_{\mathrm{av}}(\underline{x}) = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y} \in \aleph_{\underline{x}}} \partial_{\underline{x},\underline{y}} \tag{9}$$

We note that it is also possible to make μ adaptive to different regions of the image when it is known that certain region is noisier than other region. This can be done by computing (8) over a local window rather than over the entire image.

The steepness parameter σ in (7) controls the degree of influence of the neighboring pixels on the center pixel. As can be seen in Fig. 1, when σ is large, a larger dissimilarity between the center pixel and its neighbor is needed before the influence of the neighboring pixels is turned off, i.e., large ∂ is needed before λ reaches 1. Clearly, σ should be chosen carefully such that the random fluctuation is smoothed out while important image structures are preserved during clustering, i.e., we let $\lambda(\partial) \approx 1$ when ∂ between two pixels is due to genuine structures, such as object boundaries or edges in the image. We estimate σ as follows. From the $\partial_{\mathrm{av}}(\underline{x})$ computed over the image data, we take ∂_t to be equal to the 95 percentile of $\partial_{\mathrm{av}}(\underline{x})$. Then, we let $\lambda(\partial_t) = 0.8$ and solve for σ using (7).

The dissimilarity index $D_{k,\underline{x}}$ effectively smoothes the cluster assignment of $s(\underline{x})$ by the cluster assignment of its neighboring pixels adaptively. When $s(\underline{x})$ is along a genuine edge, its value will be very different from that of its neighbors, reflecting that they do not belong to the same class. Hence, ∂ will be large and $\lambda \rightarrow 1$ for all its neighbors. In this case, $D_{k,\underline{x}} \approx d_{k,\underline{x}}^2$, and the neighboring influence is turned off. When $\aleph_{\underline{x}}$ falls on a step boundary, $s(\underline{x})$ is only affected by those neighboring pixels in the same class (i.e., neighboring pixels on the same step level as $s(\underline{x})$). When $s(\underline{x})$ is on a smooth region and is affected by all its neighbors, the degree of influence of each neighbor $s(\underline{y})$ on $s(\underline{x})$ is given by the weight λ, which is a function of the dissimilarity $\partial_{\underline{x},\underline{y}}$ between $s(\underline{x})$ and $s(\underline{y})$. In this way, the novel dissimilarity index $D_{k,\underline{x}}$ enables spatial interaction between neighboring pixels to be adaptive to image content, which helps to preserve the sharpness of the memberships for pixels along region boundaries or genuine edges.

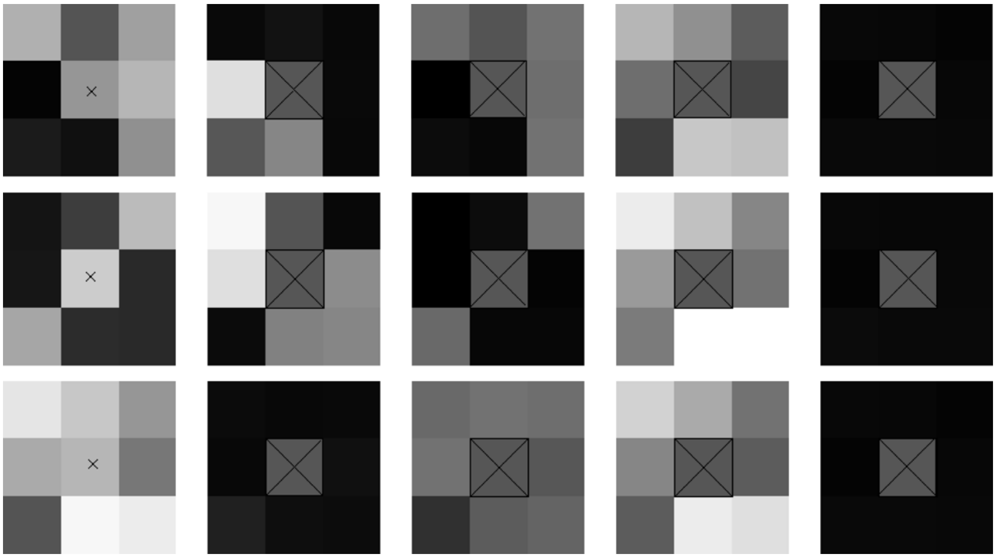


Fig. 2. Three scenarios where the N_8 neighborhood window is located on a -45° region boundary (first row), on a line inclined at $+45^\circ$ (middle row), and in a homogenous region (last row). The first column shows the pixel intensity within the window, where the center pixel is marked by a cross. The second column shows the $\partial_{\underline{x},\underline{y}}$ for the eight neighborhood pixels. The third column shows the corresponding $1-\lambda_{\underline{x},\underline{y}}$ value. The fourth and fifth columns show each of the summation term in (10) that corresponds to each of the neighborhood pixel for the first and the second class, respectively. See text for explanation.

We illustrate the effects of the new dissimilarity index $D_{k,\underline{x}}$ using an N_8 neighborhood. The $D_{k,\underline{x}}$ for the N_8 neighborhood is given by

$$D_{i,r,s} = \frac{1}{8}\sum_{l_1=-1}^{1}\sum_{l_2=-1}^{1}\left[d_{i,r,s}^2\lambda_{l_1,l_2}^{r,s} + d_{i,r+l_1,s+l_2}^2(1-\lambda_{l_1,l_2}^{r,s})\right] \quad (l_1,l_2)\neq(0,0). \tag{10}$$

Fig. 2 shows three scenarios where the N_8 neighborhood window is located on a -45° region boundary (first row), on a line inclined at $+45^\circ$ (middle row), and in a homogenous region (last row). The image consists of two classes, with centroids at 1 and 5, respectively. The center pixel (the pixel of interest for membership computation) is marked by a cross and belongs to the second class. The first column shows the pixel intensity within the window. The second column shows the $\partial_{\underline{x},\underline{y}}$, which measures the dissimilarity of the center pixel $s(\underline{x})$ and its neighbors $s(\underline{y})$. The value at the center of the window is irrelevant (irrelevant location is marked by a crossed box in all windows) and the values at the other locations are set to be the corresponding $\partial_{\underline{x},\underline{y}}$. We see that pixels with similar intensity as the center pixel have low $\partial_{\underline{x},\underline{y}}$, and vice versa. Column three shows the corresponding $1-\lambda_{\underline{x},\underline{y}}$ value for the neighborhood pixels. We see that neighborhood pixels that are not in the same class as that of the center pixel have low values, indicating that they have minimal influence in the computation of $D_{k,\underline{x}}$. The fourth and fifth columns show each of the summation term in (10) that corresponds to each of the neighborhood pixels for the first and the second class, respectively. Since the center pixel belongs to class two, the entries in column four all have high values, and the entries in column five all have low values. It is interesting to note that even though the three lower left corner neighborhood pixels of the first case (i.e., first row) belong to class one, they do not give low value in the corresponding entries in column four or high value in the corresponding entries in column five. This is due to their contributions being suppressed by the low $1-\lambda_{\underline{x},\underline{y}}$ values. The same is true of the class 1 neighborhood pixels in case two. These examples clearly illustrate the ability of the new dissimilarity index $D_{k,\underline{x}}$ to adapt to the local image content.

The spatial continuity constraint also has a noise suppression capability due to the adaptive smoothing operation. Random noise would either increase or decrease the distance of the center pixel to cluster prototype and the distance of the neighboring pixel to cluster prototypes randomly. When the weighted average of these distances is taken using (6), the effect of random noise is smoothed out. Another observation is that the incorporation of local spatial continuity inherently takes into account the local spatial ordering of the data. This is in contrast to the classical FCM clustering algorithm, where each pixel is treated as an independent instance, without considering the local ordering relationship between pixels.

B. *Adaptive Cluster Prototype*

In conventional FCM clustering, the prototype v_k of each cluster is assumed constant. For image segmentation, this implies that an image is consisted of piecewise constant or planar regions. This assumption is clearly inadequate for many real world images. A homogenous region with nonconstant intensity could be overly segmented into many smaller regions.

In order to allow better segmentation of nonplanar regions, we make the cluster prototypes adaptive to slow intensity variation within a cluster via the addition of a multiplicative field. Let $\vartheta_{k,\underline{x}}$ be the adaptive cluster prototype for the kth cluster at spatial index $\underline{x}$, then $\vartheta_{k,\underline{x}} = w_{k,\underline{x}}v_k$, where $w_{k,\underline{x}}$ is the spatially adaptive multiplicative field for the kth cluster and v_k is the original k-th FCM cluster prototype. With this modification, the original distance metric of (3) now becomes

$$d_{k,\underline{x}}^2 = \|s(\underline{x}) - w_{k,\underline{x}}v_k\|^2. \tag{11}$$

We model the slow varying multiplicative field $w_{k,\underline{x}}$ as a smooth cubic B-spline surface. The normalized cubic B-spline basis $N_{\mathrm{i},4}$ with knots $\lambda_i,\ldots,\lambda_{i+4}$ is given by [14]

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

where

$$(x-c)_+^k = \begin{cases}(x-c)^k, & \text{if } x \geq c\\ 0, & \text{if } x < c\end{cases} \qquad (13)$$

Specifically, we construct $w_{k,\underline{x}} = w_k(x,y)$ by using the tensor products of 1D cubic B-spline bases, i.e.,

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

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

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

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

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

By using the tensor product spline representation, the computation of the multiplicative field $w_k(x,y)$ becomes that of finding the set of B-spline coefficients $\{\alpha_{ij}^k\}$. Let the x and y dimensions be divided into t_x and t_y intervals, respectively. Then, the number of B-spline coefficients to be computed is $(t_x+3)\times(t_y+3)$. Since the multiplicative field is smooth and slowly varying, the number of intervals needed is small. This implies the number of unknown B-spline coefficients to be estimated is also very small.

C. Image Region Relabeling

Adaptation of the cluster prototype to the region's intensity variation results in a better fit to the region's profile, i.e., the pixel-to-centroid distance of (11) would be reduced regardless of whether a region is assigned to the correct class or not. So, if a homogenous region was wrongly segmented into multiple regions due to variation in its intensity profile, adaptation of the cluster prototype would not necessarily correct for the over-segmentation. To overcome this problem, a region relabeling procedure is needed.

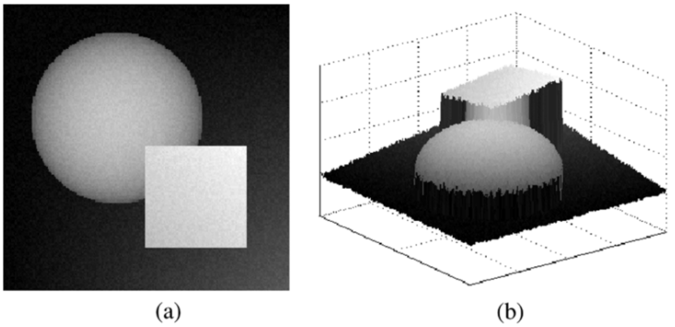


Fig. 3. Simulated image with additive noise (three object classes, i.e., two objects and one background). (a) Intensity plot of the image. (b) 3-D view of the image showing intensity variation within each object class.

Given an image A segmented by the proposed adaptive FCM algorithm, the reconstructed image $\hat{A}$ can be obtained from the fuzzy partition matrix U, the bias field W and the cluster prototypes v by $\hat{A}(\underline{x}) = \sum_{k=1}^{c} u_{k,\underline{x}}w_{k,\underline{x}}v_k$. Since prototype adaptation results in a better fit to the region's profile, region boundaries due to over-segmentation but do not correspond to genuine edges in A will not have a significant transition in the reconstructed image $\hat{A}$. By retaining only those regions having boundaries correspond to significant transitions in the reconstructed image $\hat{A}$, and merging adjacent regions that do not have boundaries with significant transitions, homogenous region within an object can be segmented into one contiguous region.

To detect the significant transitions in the reconstructed image, we first compute the gradient magnitude ζ of $\hat{A}$, i.e., $\zeta = \sqrt{(d\hat{A}/dx)^2 + (d\hat{A}/dy)^2}$. The gradient magnitude is then thresholded to remove insignificant gradient. The surviving gradient magnitude can then be used to determine whether the boundary pixels between two regions are significant or not. When majorities of the boundary pixels in a region are insignificant, the region is merged with the adjacent region with which it shares most of its boundary. This is done by relabeling its class label such that the two regions combine into one region. When only part of the shared region boundary is insignificant, the region is modified by only retaining the class label within the area enclosed by the significant boundary pieces joined together by straight lines. The class label of the discarded region fragments is changed such that the fragments merged to their adjacent regions, respectively.

The relabeling procedure is performed at each iteration after cluster prototype adaptation is activated. To obtain the initial class label, each pixel is assigned to the cluster with the largest membership value. The relabeling process affects the membership computation (and hence the final segmentation result) by influencing the pixel-to-centroid distance of (11), i.e., adaptation of the cluster prototype v_k by $w_{k,\underline{x}}$ in the distance metric (11) is only performed if the current pixel $s(\underline{x})$ is of label k. Otherwise, $w_{k,\underline{x}}$ is set to 1 (no adaptation).

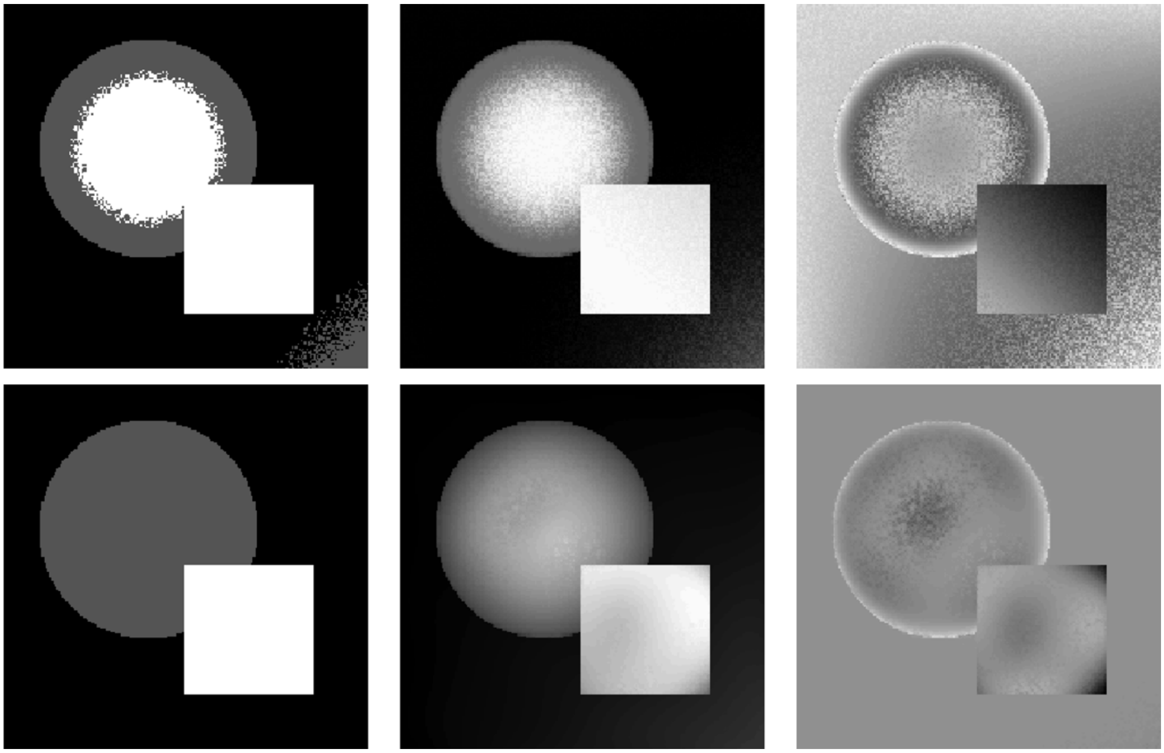


Fig. 4. Top row: Conventional FCM segmentation. 13.77% of the pixels are misclassified. Bottom row: ASFCM segmentation. Perfect segmentation is achieved. The first column shows the segmentation results, second column shows the reconstructed images, and third column shows the residual images (top: $\mathrm{MSE} = 61.99$, bottom: $\mathrm{MSE} = 12.82$), of the two algorithms, respectively.

D. Iterative Optimization

The objective function for the proposed adaptive FCM algorithm is given by

$$J_{\text{new}} = \sum_{\underline{x} \in I} \sum_{k=1}^{c} u_{k,\underline{x}}^{m} D_{k,\underline{x}} + \beta \sum_{k=1}^{c} \eta(w_k(x,y)) \tag{17}$$

with $D_{k,\underline{x}}$ defined by (6) and (11), subject to

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

where the regularizing term in (17) is given by

$$\eta(w_k(x,y)) = \iint \left\{ \left[\frac{\partial^2 w_k(x,y)}{\partial x^2} \right]^2 + 2 \left[\frac{\partial^2 w_k(x,y)}{\partial x \partial y} \right]^2 + \left[\frac{\partial^2 w_k(x,y)}{\partial y^2} \right]^2 \right\} dx\, dy. \tag{19}$$

The regularizing term of (19) minimizes the thin plate energy of the spline surfaces $w_k(x,y)$ and ensures that the estimation corresponds to a smoothing spline surface fitting [14]. The regularization parameter β controls the fidelity of the fit to the data and the smoothness of the estimated field.

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

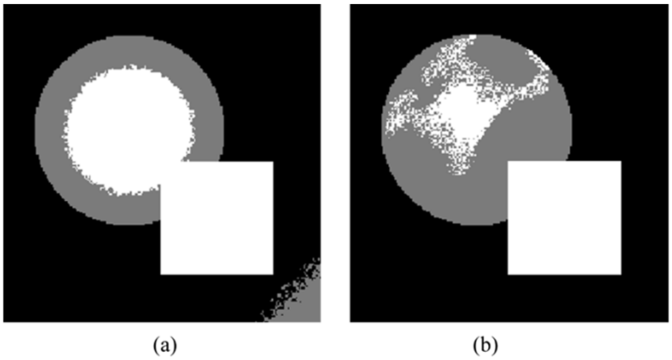


Fig. 5. (a) FCM segmentation with just the spatial continuity constraint but without cluster prototype adaptation (classification error is 13.55%). (b) FCM segmentation with cluster prototype adaptation but without the spatial continuity constraint (classification error is 5.53%).

$$L(u) = J_{\text{new}}(u) + \sum_{\underline{x} \in I} \delta_{\underline{x}} \left(1 - \sum_{k=1}^{c} u_{k,\underline{x}} \right) \tag{20}$$

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

$$u_{k,\underline{x}} = \left[\sum_{l=1}^{c} \left(\frac{D_{k,\underline{x}}}{D_{l,\underline{x}}} \right)^{1/m-1} \right]^{-1} \tag{21}$$

TABLE I
MEAN-SQUARE-ERROR (MSE) BETWEEN THE RECONSTRUCTED IMAGE AND THE ORIGINAL NOISE FREE IMAGE, AND THE PERCENTAGE CLASSIFICATION ERROR (IN BRACKET) FOR DIFFERENT PARAMETER SETTINGS. IN COMPARISON, THE MSE AND THE PERCENTAGE CLASSIFICATION ERROR FOR THE FCM ALGORITHM IS 61.99 AND 13.77%, RESPECTIVELY

β \ n	3	4	5	6	7
1 ×10^3	18.32 (1.29%)	14.59 (1.30%)	14.10 (0.01%)	14.86 (0.14%)	17.43 (1.30%)
1 ×10^4	18.31 (1.29%)	14.59 (1.30%)	14.10 (0.00%)	14.86 (0.14%)	17.41 (1.28%)
1 ×10^5	18.31 (1.27%)	14.56 (1.28%)	14.08 (0.00%)	14.85 (0.14%)	17.44 (0.75%)
1×10^6	18.23 (0.00%)	12.84 (0.00%)	12.82 (0.00%)	12.79 (0.00%)	17.16 (0.26%)
1×10^7	18.01 (0.02%)	15.18 (0.00%)	16.90 (0.00%)	17.54 (0.00%)	18.08 (0.01%)
1×10^8	17.25 (0.01%)	18.79 (0.01%)	25.99 (0.08%)	36.51 (0.71%)	36.42 (0.68%)

for all $k, \underline{x}$. Differentiating J_{new} with respect to v_k and setting the result to zero yields

$$v_k = \frac{\sum_{\underline{x}\in I} u_{k,\underline{x}}^m g_k(\underline{x})}{\sum_{\underline{x}\in I} u_{k,\underline{x}}^m h_k(\underline{x})} \tag{22}$$

where

$$g_k(\underline{x}) = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y}\in\aleph_{\underline{x}}} [\lambda_{\underline{x},\underline{y}} s(\underline{x}) w_k(\underline{x}) + (1-\lambda_{\underline{x},\underline{y}}) s(\underline{y}) w_k(\underline{y})] \tag{23}$$

$$h_k(\underline{x}) = \frac{1}{|\aleph_{\underline{x}}|} \sum_{\underline{y}\in\aleph_{\underline{x}}} \left[\lambda_{\underline{x},\underline{y}} w_k^2(\underline{x}) + (1-\lambda_{\underline{x},\underline{y}}) w_k^2(\underline{y})\right] \tag{24}$$

for all $k = 1$ to c.

In deriving the first derivative of J_{new} with respect to the B-spline coefficients $\{\alpha_{ij}^k\}$, we ignore the spatial interactions between neighboring pixels since the spline surface to be estimated is already very smooth. To find an expression for $\{\alpha_{ij}^k\}, u_{k,\underline{x}}$, and v_k are held constant. Then the modified J_{new} is differentiated with respect to α_{pq}^k. By setting the result to zero, the following set of linear equations are obtained:

$$\begin{aligned}\sum_{i=-3}^{g}\sum_{j=-3}^{h} \alpha_{ij}^k L_k(p,q,i,j) - b_k(p,q) \\ + \beta \sum_{i=-3}^{g}\sum_{j=-3}^{h} \alpha_{ij}^k[\omega_1(p,q,i,j) \\ + 2\omega_2(p,q,i,j) + \omega_3(p,q,i,j)] = 0\end{aligned} \tag{25}$$

for all $p = -3,\ldots,g$ and $q = -3,\ldots h$, where

$$L_k(p,q,i,j) = v_k^2 \sum_{(x,y)\in I} u_{k,\underline{x}}^m N_{p,4}(x) M_{q,4}(y) N_{i,4}(x) M_{j,4}(y) \tag{26}$$

$$b_k(p,q) = v_k \sum_{(x,y)\in I} u_{k,\underline{x}}^m s(\underline{x}) N_{p,4}(x) M_{q,4}(y) \tag{27}$$

$$\omega_1(p,q,i,j) = \int N_{i,4}^{''}(x) N_{p,4}^{''}(x)\, dx \int M_{j,4}(y) M_{q,4}(y)\, dy \tag{28}$$

$$\omega_2(p,q,i,j) = \int N_{i,4}^{'}(x) N_{p,4}^{'}(x)\, dx \int M_{j,4}^{'}(y) M_{q,4}^{'}(y)\, dy \tag{29}$$

$$\omega_3(p,q,i,j) = \int N_{i,4}(x) N_{p,4}(x)\, dx \int M_{j,4}^{''}(y) M_{q,4}^{''}(y)\, dy \tag{30}$$

and the single and double prime in N and M denote first and second derivatives, respectively.

As the resultant system of equations is linear, the B-spline coefficients $\{\alpha_{ij}^k\}$ can be solved efficiently using the direct least squares error approach. By using the ordering $t = (p+3) \times (h+4) + (q+4)$ and $u = (i+3) \times (h+4) + (j+4)$, we rearrange $L_k(p,q,i,j), \omega_l(p,q,i,j), l = 1,2,3$, into square matrices L_k and $\{\Omega_l\}$ with indexes t and u. When $g = h$, i.e., the number of knots in the x and y dimensions are equal, L_k is a symmetric matrix with $L_k = L_k^T$. In addition, the local support property of (16) allows the summations in (26) to be done over a small region in the image domain. The matrices $\{\Omega_l\}$ need to be calculated only once. When the x and y dimensions of the image are equal, symmetry can also be observed in $\{\Omega_l\}$. The B-spline values $N_{i,4}(x)$ can be evaluated in a numerically stable way using the recurrence relation

$$\begin{aligned} N_{i,l+1}(x) &= \frac{x-\lambda_i}{\lambda_{i+l}-\lambda_i} N_{i,l}(x) + \frac{\lambda_{i+l+1}-x}{\lambda_{i+l+1}-\lambda_{i+1}} N_{i+1,l}(x) \\ N_{i,1}(x) &= \begin{cases} 1, & \text{if } x \in [\lambda_i, \lambda_{i+1}) \\ 0, & \text{if } x \notin [\lambda_i, \lambda_{i+1}) \end{cases} \end{aligned} \tag{31}$$

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

$$N_{i,k+1}^{'}(x) = k\left\{\frac{N_{i,k}(x)}{\lambda_{i+k}-\lambda_i} - \frac{N_{i+1,k}(x)}{\lambda_{i+k+1}-\lambda_{i+1}}\right\}. \tag{32}$$

With L_k and $\{\Omega_l\}$, the set of $(g+4) \times (h+4)$ equations from (27) can be expressed in matrix notation, $A_k\theta_k = b_k$, where A_k is the $(g+4) \times (h+4)$ by $(g+4) \times (h+4)$ sparse matrix given by $L_k + \beta(\Omega_1 + 2\Omega_2 + \Omega_3), \theta_k$ is the vector of B-spline coefficients $\{\alpha_{ij}^k\}$ arranged using the ordering u, and b_k is the vector obtained from (27) using the ordering t. The B-spline coefficients $\{\alpha_{ij}^k\}$ are obtained as $\theta_z = VS^+U^Tb_z$ using SVD [15], where S^+ is a diagonal matrix with diagonal elements given by the reciprocal of that in S, if it is greater than a small tolerance, or zero otherwise, and U and V are column orthogonal and square orthogonal matrices, respectively.

IV. PROPOSED ALGORITHM

The solution to the objective function (17) involves a local optimization procedure. Hence, proper initialization of the proposed algorithm is important for the convergence of the result to the desired solution. Whenever the values of the object class are approximately known, they should be used as the initial cluster prototypes. When such knowledge is not available, the initial

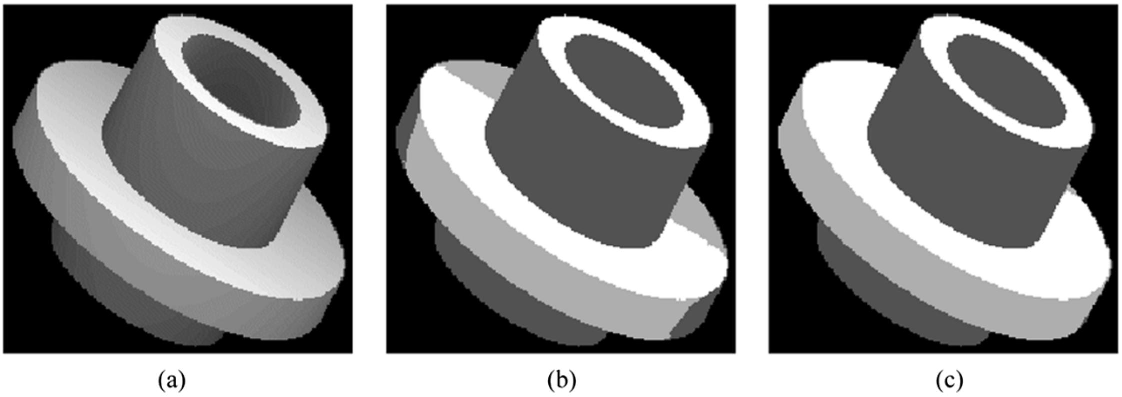


Fig. 6. (a) Original image. (b) FCM segmentation. (c) ASFCM segmentation with spatial continuity constraint and prototype adaptation.

prototypes could be estimated as regions of dense data in the feature space as follows. First, we compute the smooth histogram for each dimension, and record the location of the prominent peaks that are above a certain threshold. The intersections of these locations specify the possible concentration of data in the multidimension space. The data density within a local region around these intersections is computed, and the c intersections with the highest data density are chosen as the c initial cluster prototypes. The initial multiplicative field for each cluster is set to be a constant field of unity value.

The choice of neighborhood configuration $\aleph_\rho$ can be tailored to the specific problem and the topology of the spatial interaction. The local neighborhood configuration we used is the simple symmetrical eight-point neighborhood. Each neighbor has equal contribution, i.e., equal weight, to the dissimilarity measure in (6).

As with the conventional FCM algorithm, our algorithm iterates to the final solution by updating the cluster prototypes, membership values, and multiplicative fields alternatingly. Thus, when the membership values and cluster prototypes are changing rapidly during the first few iterations, the multiplicative fields cannot be updated in a stable manner. We therefore only activate the adaptation of the cluster prototypes when the L_∞ change in membership value has settled down sufficiently (i.e., when $||U^{t+1} - U^t|| < 0.01$). The procedures for carrying out the adaptive cluster prototype FCM-based image segmentation can now be stated as the following.

1) Adaptive Cluster Prototype FCM Segmentation Algorithm:

1) Set the number of clusters c. Set $m = 2$. Choose a value for the spline smoothness weighting coefficient β. Set the number of splines knots in the x and y dimensions. Set the maximum number of iterations *ITMAX*. Initialize the multiplicative fields $\{w_k(\underline{x})\}, k = 1, \ldots, c$, to unity. Set the flag *ADAPT* to false.

2) Obtain initial estimates of the cluster centroids v_k as outlined previously.

3) Compute $\lambda_{\underline{x},y}$ using (7)–(9).

4) Compute the initial membership for every pixel using (21).

5) Compute the regularizing matrix $\Omega = \Omega_1 + 2\Omega_2 + \Omega_3$ using (28)–(30).

6) Repeat for $t = 1$ to *ITMAX* or until maximum change in membership value is less than a small threshold ε.

i) When maximum change in membership value is less than 0.01 and *ADAPT* is false, set *ADAPT* to true.

ii) When *ADAPT* is true, update $\{w_k(\underline{x})\}$ by solving for the B-spline coefficients $\{\alpha_{ij}^k\}$.

iii) Update the fuzzy cluster centroids using (22).

iv) If *ADAPT* is true, perform region relabeling.

v) Update the membership values using (21).

7) Perform a final hard classification by assigning the pixel to the cluster with the highest membership value.

V. Image Segmentation Experiments

To illustrate the performance of the proposed algorithm, the ASFCM algorithm is run on an artificial three-class image consists of two objects in a nonuniform background, as shown in Fig. 3. The image is corrupted with additive Gaussian noise. For comparison, the same image is also segmented using the conventional FCM algorithm. The parameter settings are: $c = 3, m = 2, n = 5, \beta = 1 \times 10^6$. Looking at the first column of Fig. 4, it can be seen that the proposed ASFCM algorithm performs superior to the FCM algorithm for this test image. The three regions were resolved perfectly by the ASFCM algorithm, whereas the FCM leads to erroneous segmentation. The second column in Fig. 4 shows the reconstructed images obtained from the FCM and ASFCM, respectively, where the reconstructed image r is given by $r(\underline{x}) = \sum_{k=1}^{c} u_{k\underline{x}} w_{k,\underline{x}} v_k$. The residual images, i.e., difference between the reconstructed image and the noise free test image, are shown in the last column of Fig. 4. It can be seen that the residual image of ASFCM is much more uniform than that of FCM and most of the intensity variations in the objects were successfully captured in the reconstructed image.

In order to see the contribution of each of the two modifications, we perform segmentation using only the spatial continuity constraint or using only prototype adaptation. The results are shown in Fig. 5. Both results show an improvement over that of conventional FCM. The classification error is 13.77% for FCM, 13.55% with spatial continuity only and 5.53% with prototype adaptation only. We see that the spatial continuity constraint helps to smooth out noise, but by itself it is not capable of handling intensity variations in the objects. On the other hand, prototype adaptation was able to capture some of the intensity

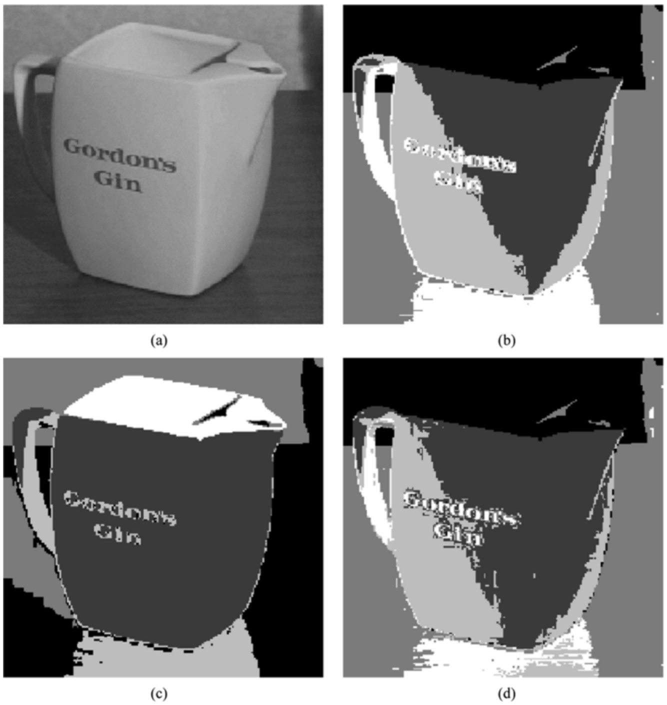


Fig. 7. (a)"Teapot" image. (b) FCM segmentation. (c) ASFCM segmentation. (d) Gustafson and Kessell's fuzzy clustering algorithm [16].

variations within the objects (i.e., the nonuniform background and the square object were segmented correctly), but it still fails to segment the sphere object correctly. Since the spatial continuity constraint has a spatial regularization effect on the membership assignment, it helps to facilitate prototype adaptation. Thus, by using both spatial continuity constraint and prototype adaptation, all objects can be segmented correctly (see Fig. 4).

We also investigate the sensitivity of the proposed algorithm to different parameter settings by varying the regularization parameter β and the number of spline intervals n (we set the number of spline intervals in both x and y directions to be equal). The performance of the algorithm is measured in terms of the mean squared reconstruction error between the reconstructed image and the original noise free image, and the percentage classification error. The test image of Fig. 3 is used in the experiment. The results in Table I indicated that good performance can still be achieved even if β is varied over several orders of magnitude. In addition, only a small number of spline intervals are needed for the prototype adaptation. A final point to note is that the proposed algorithm with the different parameter combinations all performs significantly better than the original FCM algorithm.

Fig. 6 shows the segmentation of a synthetic object where each homogenous region has nonuniform intensity. The segmentation using the FCM algorithm is shown in Fig. 6(b). We see that due to intensity overlap between different regions, some parts of the object are segmented incorrectly. In comparison, the ASFCM algorithm produced a much better segmentation in Fig. 6(c) due to adaptation of the prototypes. We also perform segmentation on some color images. To correspond to perceptual color difference, the RGB color image is first transformed to the CIELAB color space [18]. The clustering is then done in the CIELAB space using the L, a, b components as features. Prototype adaptation is only done on the luminance component L, but not on the chrominance components a and b since they are insensitive to illumination variation. Fig. 7 shows the segmentation of a color image "teapot" into five clusters. Fig. 7(a) is the original "teapot" image. Fig. 7(b), (c), and (d) are the segmentation results obtained from the conventional FCM, ASFCM, and the FCM clustering algorithm of Gustafson and Kessell with a fuzzy covariance matrix (gkFCM) [16], respectively. In gkFCM, the pixel-to-centroid distance is weighted by the fuzzy covariance matrix. The covariance-weighted distance allows the FCM algorithm to handle elliptical clusters and therefore provides some degree of adaptation to within cluster variation. However, such adaptation is generally not adequate to model the smooth intensity variation in a homogenous region. Moreover, there is no constraint on the smoothness of the variation since inter-pixel spatial correlations are ignored in the feature space. From Fig. 7, we see that the ASFCM is able to produce better segmentation than the other algorithms, as it has managed to segment the body of the teapot as a whole. The gkFCM actually produce poorer result than the conventional FCM algorithm, i.e., the segmentation is noisier. We think that this is due to the lack of smoothness constraint on the within cluster adaptation offered by the covariance-weighted distance. Fig. 8 presents additional color segmentation results using the proposed algorithm.

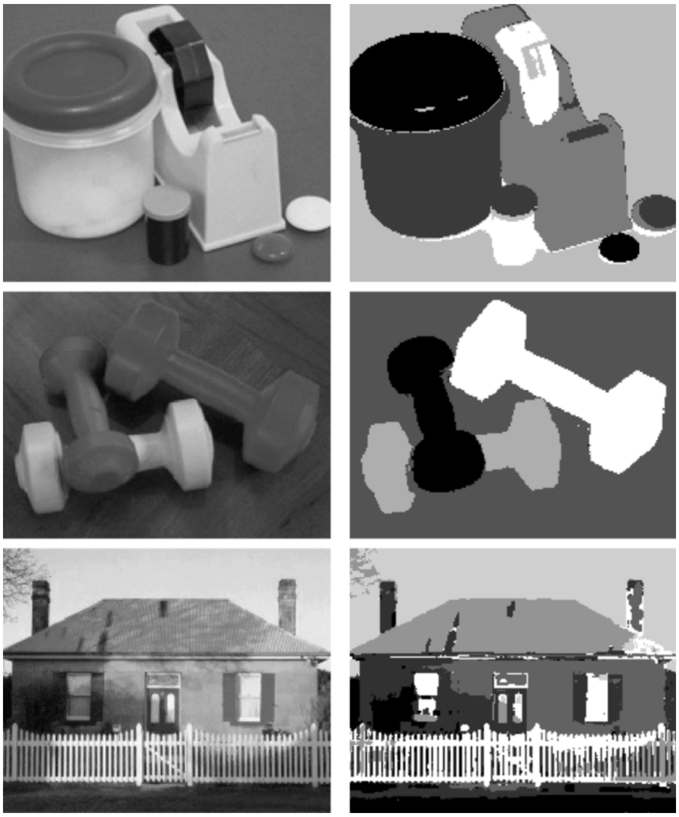


Fig. 8. Some color segmentation results using the proposed algorithm.

Although it is possible to use the covariance-weighted norm of [16] in place of the Euclidean norm in our algorithm, we found that the extra computation incurred using the covariance-weighted norm does not result in any noticeable difference in the final clustering. The cluster prototype adaptation allows the prototype to closely trace the intensity profile of the homogenous region, so that the deviation of the data from the prototype becomes more spherically distributed than that without prototype adaptation. The local spatial continuity constraint is also able to handle elliptical clusters to some extent [13].

VI. Conclusion

A fuzzy clustering-based segmentation algorithm is proposed in this paper. The proposed algorithm is able to take into consideration the high correlation between neighboring pixels, i.e., local spatial continuity, as well as being adaptive to smooth intensity variation within homogenous regions. The local spatial continuity is enforced through the use of a novel dissimilarity index $D_{i,r,s}$, which measures the dissimilarity between a pixel and a cluster prototype under the influence of its local neighborhood. The dissimilarity index is adaptive to the image content within the neighborhood window. This implies that if the window is in a nonhomogeneous region, the influence of the neighboring pixels on the center pixel is suppressed. Otherwise, the center pixel is smoothed by its neighboring pixels during the computation of memberships and cluster prototypes. The cluster prototypes are made adaptive spatially by the use of a multiplicative field for each cluster, where each field is constructed using splines. With the ability to take into account the image's intensity variation within cluster and the inter-pixel spatial correlation, the proposed ASFCM algorithm has been shown to perform more meaningful image segmentation compared to the conventional FCM and several FCM-based algorithms.

References

[1] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recogn.*, vol. 26, no. 9, pp. 1277–1294, 1993.

[2] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *CVGIP*, no. 29, pp. 100–132, 1985.

[3] S. K. Pal, R. A. King, and A. A. Hashim, "Automatic grey level thresholding through index of fuzziness and entropy," *Pattern Recog. Lett.*, vol. 1, pp. 141–146, Mar. 1983.

[4] F. Wang, "Fuzzy classification of remote sensing images," *IEEE Trans. Geosci. Rem. Sensing*, vol. 28, no. 2, pp. 194–201, Mar. 1990.

[5] H. D. Cheng, Y. H. Chen, and X. H. Jiang, "Thresholding using two-dimensional histogram and fuzzy entropy principle," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 732–735, Apr. 2000.

[6] M. S. Zhao, A. A. Fu, and H. Yan, "A technique of three-level thresholding based on probabilistic partition and fuzzy three-partition," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 3, pp. 469–479, Jun. 2001.

[7] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[8] J. O. Eklundh, H. Yamamoto, and A. Rosenfeld, "A relaxation method for multi-spectral pixel classification," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, no. 1, pp. 72–75, Jan. 1980.

[9] J. Y. Hsiao and A. Sawchuk, "Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 12, pp. 1279–1292, Dec. 1989.

[10] ——, "Unsupervised textured image segmentation using feature smoothing and probabilistic relaxation techniques," *Comput. Vision, Graph., Image Process.*, vol. 48, no. 1, pp. 1–21, 1989.

[11] Y. A. Tolias and S. M. Panas, "Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 28, no. 3, pp. 359–369, May 1998.

[12] A. W. C. Liew, S. H. Leung, and W. H. Lau, "Fuzzy image clustering incorporating spatial continuity," *Proc. Inst. Elect. Eng.—Vision, Image, Signal Process.*, vol. 147, no. 2, pp. 185–192, Apr. 2000.

[13] ——, "Segmentation of color lip images by spatial fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 4, pp. 542–549, Aug. 2003.

[14] P. Dierckx, "Curve and surface fitting with splines," in *Monographs on Numerical Analysis*. Oxford, U.K.: Oxford Univ., 1993.

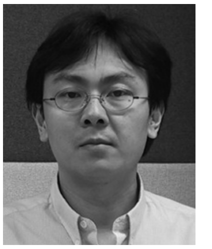
[15] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1992.

[16] D. E. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proc. IEEE Conf. Decision and Control*, vol. 2, K. S. Fu, Ed.. Piscataway, NJ, 1979, pp. 761–766.

[17] A. W. C. Liew and H. Yan, "An adaptive spatial fuzzy clustering algorithm for MR image segmentation," *IEEE Trans. Med. Imag.*, vol. 22, no. 9, pp. 1063–1075, Sep. 2003.

[18] R. W. G. Hunt, *Measuring Color*, 2nd ed. London, U.K.: Ellis Horwood, 1992.

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



Alan Wee-Chung Liew (M'03) received the B.Eng. degree (with first class honors) in electrical and electronic engineering from the University of Auckland, Auckland, New Zealand, in 1993, and the Ph.D. degree in electronic engineering from the University of Tasmania, Hobart, Australia, in 1997.

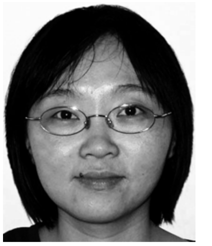
He is currently an Assistant Professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. His current research interests include computer vision, medical imaging, pattern recognition, and bioinformatics.

Dr. Liew has served as a Technical Reviewer for a number of international conferences and journals in IEEE Transactions, IEE Proceedings, bioinformatics, and computational biology. His biography is listed in the 2005 Marquis Who's Who in the World and Marquis Who's Who in Science and Engineering.

Hong Yan (S'88–M'89–SM'93) received the B.E. degree from Nanking Institute of Posts and the M.S.E. degree from the University of Michigan, Ann Arbor, in 1984, and the Ph.D. degree from Yale University, New Haven, CT, in 1989, all in electrical engineering.

In 1982 and 1983, he worked on signal detection and estimation as a graduate student and research assistant at Tsinghua University, Beijing, China. From 1986 to 1989, he was a Research Scientist at General Network Corporation, New Haven, CT, where he worked on design and optimization of computer and telecommunications networks. He joined the University of Sydney, Sydney, Australia, in 1989, and became Professor of Imaging Science in 1997. He is currently Professor of Computer Engineering at the City University of Hong Kong. His research interests include image processing, pattern recognition and bioinformatics. He is the author or coauthor of one book and over 200 refereed technical papers in these areas.

Dr. Yan is a Fellow of the International Association for Pattern Recognition (IAPR), a Fellow of the Institution of Engineers, Australia (IEAust), and a Member of the International Society for Computational Biology (ISCB).



Ngai-Fong Law (M'98) received the B.Eng. degree (with first-class honors) from the University of Auckland, Auckland, New Zealand, in 1993, and the Ph.D. degree from the University of Tasmania, Hobart, Australia, in 1997, both in electrical and electronic engineering.

She is currently an Assistant Professor in the Electronic and Information Engineering Department, Hong Kong Polytechnic University, Hong Kong. Her research interests include signal and image processing, wavelet transform, image enhancement, and compression. Recently, she has also been working on Web-based system design and video searching for internet applications.