Automatic Fuzzy Clustering Framework for Image Segmentation

Tao Lei, *Member, IEEE*, Peng Liu, Xiaohong Jia, Xuande Zhang, Hongying Meng, *Senior Member, IEEE*, and Asoke K. Nandi, *Fellow, IEEE*

***Abstract*—Clustering algorithms by minimizing an objective function share a clear drawback of having to set the number of clusters manually. Although density peak clustering is able to find the number of clusters, it suffers from memory overflow when it is used for image segmentation because a moderate-size image usually includes a large number of pixels leading to a huge similarity matrix. To address this issue, here we proposed an automatic fuzzy clustering framework (AFCF) for image segmentation. The proposed framework has threefold contributions. First, the idea of superpixel is used for the density peak (DP) algorithm, which efficiently reduces the size of the similarity matrix and thus improves the computational efficiency of the DP algorithm. Second, we employ a density balance algorithm to obtain a robust decision-graph that helps the DP algorithm achieve fully automatic clustering. Finally, a fuzzy c-means clustering based on prior entropy is used in the framework to improve image segmentation results. Because the spatial neighboring information of both the pixels and membership are considered, the final segmentation result is improved effectively. Experiments show that the proposed framework not only achieves automatic image segmentation, but also provides better segmentation results than state-of-the-art algorithms.**

***Index Terms*—Density peak (DP) algorithm, fuzzy clustering, image segmentation, superpixel.**

Manuscript received February 17, 2019; revised April 20, 2019; accepted July 12, 2019. Date of publication July 23, 2019; date of current version September 1, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61871259, Grant 61811530325 (IEC\NSFC\170396, Royal Society, U.K.), Grant 61461025, Grant 61871260, Grant 61672333, and Grant 61873155, and in part by China Postdoctoral Science Foundation under Grant 2016M602856. *(Corresponding author: Tao Lei.)*

T. Lei is with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an 710021, China, and also with the School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: leitao@sust.edu.cn).

P. Liu and X. Jia are with the School of Electrical and Control Engineering, Shaanxi University of Science and Technology, Xi'an 710021, China (e-mail: liupengctu@foxmail.com; jiaxhsust@163.com).

X. Zhang is with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an 710021, China (e-mail: zhangxuande@sust.edu.cn).

H. Meng is with the Department of Electronic and Computer Engineering, Brunel University London, UB8 3PH London, U.K. (e-mail: hongying.meng@brunel.ac.uk).

A. K. Nandi is with the College of Electronic and Information Engineering, Tongji University, Key Laboratory of Embedded Systems and Service Computing, Shanghai 200092, China, and also with the Department of Electronic and Computer Engineering, Brunel University London, UB8 3PH London, U.K. (e-mail: asoke.nandi@brunel.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2019.2930030

I. Introduction

CLUSTERING, grouping the objects of a dataset into meaningful subclasses, is one of the most popular research topics, since it is a useful tool for data mining [1], machine learning [2], and computer vision [3]. With the rapid development of intelligent technologies, automated knowledge discovery based on clustering becomes more and more important. Although many clustering algorithms have been successfully used in image segmentation and data classification [4], [5], it is still a challenging topic because it is difficult to achieve automatic clustering and to provide fine results for image segmentation. Image segmentation algorithms based on clustering have three advantages. First, they can achieve unsupervised image segmentation without labels. Second, they are more robust than other image segmentation algorithms such as active contour models [6], graph cuts [7], random walkers [8], and region merging [9], since they require fewer parameters. Finally, clustering has a clear advantage on multichannel image segmentation because it is easy to apply clustering algorithms to high-dimensional data classification. Inevitably, clustering has some disadvantages for image segmentation as well. It is sensitive to noise because the local spatial information of pixels is missed, and it takes much time for high-resolution images, as repeated calculations and an iterative optimization are required for the same pixels.

The first shortcoming is addressed by incorporating local spatial information into objective functions to improve the robustness of algorithms to noise, such as fuzzy c-means (FCM) clustering algorithm with spatial constraints (FCM_S) [10], FCM_S1/S2 [11], fuzzy local information c-means clustering algorithm (FLICM) [12], neighborhood weighted FCM clustering algorithm (NWFCM) [13], the FLICM based on kernel metric and weighted fuzzy factor (KWFLICM) [14], and deviation-sparse fuzzy c-means with neighbor information constraint (DSFCM_N) [15]. Nonetheless, they have two limitations. One is that they need much more time than conventional fuzzy c-means clustering algorithm due to the high computational complexity, and the running time is much worse when these algorithms are used for color image segmentation, as the spatial neighboring information is calculated in each iteration. The other one is that these algorithms employ a fixed neighboring window for each pixel in an image, which leads to a poor segmentation result. For this problem, an instinctive idea is to employ adaptive neighboring information to improve segmentation results. Liu *et al.* [16] improved FCM algorithm by integrating the distance between different regions obtained by mean-shift [17] and the distance of pixels into its objective function. However, as the algorithm employs adaptive neighboring information, its computational

complexity is still very high, which limits its practicability in image segmentation.

For the second shortcoming, because the number of gray levels is much smaller than the number of pixels in an image, researchers often perform clustering on gray levels instead of pixels to avoid the repeated distance computation, which can indeed reduce the execution time of algorithms such as enhanced FCM (EnFCM) [18], fast generalized FCM algorithm (FGFCM) [19], and fast and robust FCM (FRFCM) [20]. These improved FCM algorithms achieve a high computational efficiency by integrating histogram to its objective function. But it is difficult to extend these algorithms to color images because the histogram of color images is more complex than the one of grayscale images. A new problem is how to reduce the computational complexity of algorithms while efficiently improving the utilization of spatial neighboring information. To reduce the computational complexity while utilizing the adaptive spatial neighboring information, Gu *et al.* [21] employed a superpixel approach to obtain adaptive neighboring information and to reduce the number of clustering samples. They proposed a fuzzy double c-means clustering based on sparse self-representation (FDCM_SSR), which still has a higher computational complexity than most of popular algorithms. Inspired by superpixel technology [22] and EnFCM, Lei *et al.* [23] proposed a superpixel-based fast FCM algorithm (SFFCM) for color image segmentation. SFFCM has two advantages. One is that the proposed watershed transform based on multiscale morphological gradient reconstruction (MMGR-WT) is able to provide an excellent superpixel result that is useful for improving the final clustering result. The other one is that the color histogram is integrated into the objective function of FCM to speed up the implementation of the algorithm. Although SFFCM is excellent for color image segmentation, it requires the number of clusters to be set manually.

To achieve automatic clustering algorithms, researchers tried to estimate the number of clusters using different algorithms such as eigenvector analysis [24], genetic algorithm [25], the particle swarm optimization [26], and the robust learning-based schema [27]. Although these algorithms can find the number of clusters in any unlabeled data set, they are unsuitable for image segmentation since the spatial information is missed and the corresponding segmentation result is coarse. Density peaks (DP) algorithm proposed by Rodriguez and Laio [28] first finds the local density peaks of data, then computes the minimal distance between a center and other centers that have higher local density than the center, and finally obtains a decision-graph to achieve fast clustering. However, DP algorithm only provides decision-graph without giving the number of clusters. Wang and Song [29] proposed a more robust and effective automatic clustering algorithm to overcome the shortcomings of DP algorithm. Though this new algorithm can obtain automatically the number of clusters and provides better experimental results, it is still unsuitable since the spatial information of images is ignored.

In this paper, we propose an automatic fuzzy clustering framework (AFCF) for image segmentation. The proposed AFCF is inspired by image superpixel, the DP algorithm, and prior entropy-based fuzzy clustering. Although the similarity matrix of an image is often huge, which limits the application of the DP algorithm in image segmentation, we use a superpixel algorithm to simplify an image to obtain a small similarity matrix that depends on the number of superpixel. Based on the small similarity matrix, we compute the corresponding decision graph. To obtain automatic clustering algorithms, we need to improve the decision graph to obtain the number of clusters directly without human–computer interaction. Finally, prior entropy is integrated into FCM to improve segmentation results. The proposed AFCF is able to achieve automatic image segmentation with a high precision. Three advantages of the proposed AFCF are presented below.

1) AFCF is a fully automatic clustering framework for image segmentation, where the number of clusters is not a required parameter, unlike existing clustering algorithms.
2) AFCF provides accurate number of clusters and achieves better image segmentation than state-of-the-art algorithms because of the utilization of both the spatial information of images and prior entropy.
3) Because image superpixel addresses the problem of memory overflow, AFCF has a low memory demand compared to algorithms connected with the DP algorithm.

The rest of this paper is organized as follows. In Section II, we illustrate motivations of this work. In Section III, we propose our methodology and analyze its superiority. The experimental results on synthetic and real images are described in Section IV. Finally, Section V concludes this paper.

II. Motivations

Image segmentation results provided by clustering always depend on the number of clusters. Although a lot of existing adaptive clustering algorithms [30], [31] can automatically estimate the number of clusters, they are difficult and unwieldy for practical image segmentation. The DP algorithm can generate a decision-graph that is helpful for finding the number of clusters but it suffers from high computational complexity when it is used in image segmentation. To achieve automatic clustering for image segmentation, two problems need to be overcome. The first one is to remove redundant information of images to obtain a small similarity matrix used for the DP algorithm. The second one is to improve the DP algorithm to obtain accurate number of clusters and to achieve better image segmentation. We employ superpixel algorithms to simplify the computation of the DP algorithm, then utilize a density balance algorithm to find the number of clusters, and use prior entropy to improve image segmentation effect.

A. Parameter-Free Clustering

In popular clustering algorithms, such as *k*-means, FCM, and spectral clustering, the number of clusters is set manually. DP algorithm can automatically recognize potential cluster centers assuming that a cluster center often has higher density than its surrounding points and that a cluster center often has a relatively large distance from other cluster centers with high density. For each sample x_i, the local density ρ_i and the minimal distance δ_i can be computed. Both ρ_i and δ_i are obtained from the data. The local density ρ_i of the sample x_i is presented as follows:

$$\rho_i = \sum_{j=1, j\neq i}^{N} e^{-\frac{d_{ij}^2}{d_c}} \tag{1}$$

where N is the total number of samples in a data set, $1 \leqslant i, j \leqslant N$, and d_{ij} denotes the Euclidean distance between x_i and x_j. The d_c is the cutoff distance that is an essential global decay parameter of the weight. The value of d_c is usually around 2% of neighbors [28]. According to (1), ρ_i describes the density

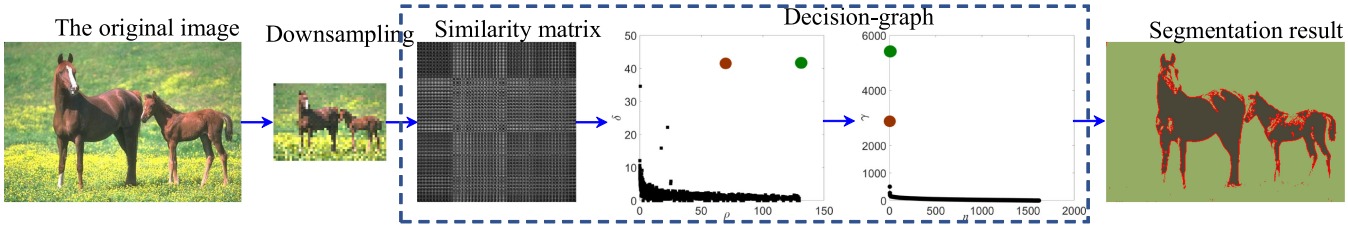


Fig. 1. Image segmentation framework using the DP algorithm.

intensity of x_i using Gaussian kernel. In general, a large ρ_i is considered as a cluster center while a small ρ_i is considered as noise or outliers in data sets.

The parameter δ_i indicates the minimal distance between the sample x_i and any other samples with higher density. The δ_i is defined as

$$\delta_i = \underbrace{\min}_{j:\rho_j > \rho_i} (d_{ij}). \tag{2}$$

Note that $\delta_i = \max_j(d_{ij})$ is used for the sample with the highest density. The anomalously large value of δ_i is helpful for recognizing the hidden cluster centers. By building a decision-graph with horizontal-axis ρ and vertical-axis δ, we can easily choose the samples of high ρ and relatively high δ as cluster centers. However, it is very difficult to select the appropriate cluster centers when there is a series of continuous sparse points in decision-graphs. To simplify the selection of cluster centers, the DP algorithm designs a new decision scheme by individually computing $\gamma_i = \rho_i \delta_i$ sorted in decreasing order. The new scheme can effectively avoid interference of false centers and easily define the potential centers. After finding the cluster centers, each remaining pixel is allocated to the same cluster as its nearest neighbor of higher density.

Yet it is difficult to extend the DP algorithm for image segmentation, since the computation of ρ and δ depends on a very large similarity matrix. For example, the similarity matrix corresponding to an image of size $P \times Q$ has a large size of $(P \times Q) \times (P \times Q)$, which always causes memory overflow if P or Q is large. To overcome this issue, the downsampling approach is often used to reduce the size of the matrix $(P \times Q) \times (P \times Q)$ [32], [33]. Fig. 1 shows the image segmentation framework using the DP algorithm. In Fig. 1, one can obtain cluster centers easily because there are two points with large value of ρ and δ in the decision-graph. However, the final segmentation result is coarse. Furthermore, Fig. 2 shows more results generated by the DP algorithm.

Figs. 1 and 2 show that the DP algorithm is able to achieve roughly automatic image segmentation. The segmentation result is good when the input image (the first image) is simple, but the results are poor when input images (the last three images) are complex as shown in Fig. 2. To improve these segmentation results, two issues need to be addressed.

1) The downsampling operation is a rough way to reduce the size of a similarity matrix. We need to develop a new algorithm that is not only able to reduce the size of a similarity matrix, but also can preserve the structuring information of images.
2) Decision-graphs need to be improved to obtain automatically the number of clusters.

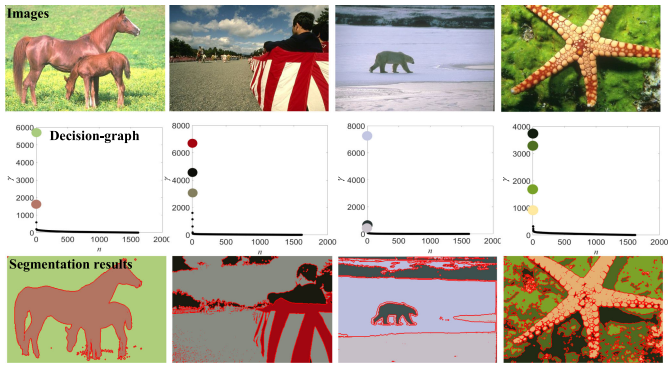


Fig. 2. Segmentation results using the DP algorithm. The DP algorithm provides a good segmentation result for the first image but poor segmentation results for the last three images.

In this paper, we employ superpixel algorithms to address the first issue, and use a density balance algorithm to overcome the second issue. The detailed description is presented in Section III.

B. Superpixel-Based Fast FCM

Automatic and fast image segmentation algorithms have important practical value for intelligent image understanding systems. First, image segmentation is a key step in image understanding, as the computational efficiency of algorithms contributes to the practical value of image understanding systems. Second, a fast image segmentation algorithm can reduce the requirement and consumption of hardware resources, which is especially important for mobile electronic devices. Therefore, improved computationally efficient image segmentation algorithms are of much practical significance. However, compared with some fast image segmentation algorithms [34], [35], clustering algorithms have a lower computational efficiency when they are used to segment high-resolution images. Fortunately, superpixel technology [36] plays a key role in improving the execution efficiency of image segmentation algorithms. Superpixel means that an image is divided into a large number of small and independent areas with different sizes and shapes [37]. Based on the superpixel result of an image, one can use a pixel to replace all pixels in a superpixel area to reduce efficiently the number of pixels in an image. Motivated by this, Lei *et al.* proposed SFFCM [23] for color image segmentation. The SFFCM addresses two difficulties in existing clustering algorithms for color image segmentation. One is that the SFFCM presents an excellent superpixel approach named MMGR-WT, and the superpixel image obtained by MMGR-WT is helpful for improving segmentation effect because the adaptive neighboring

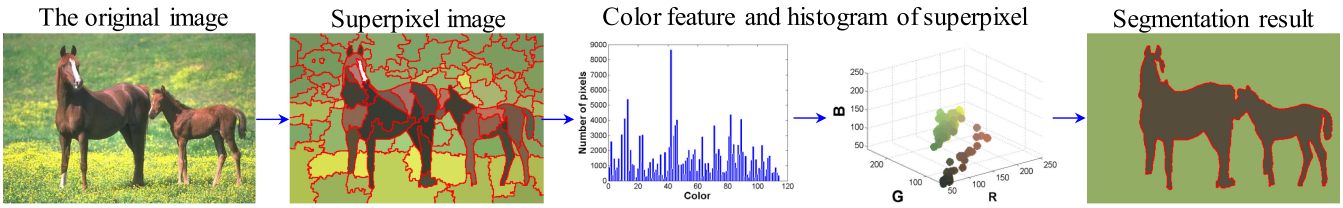


Fig. 3. Image segmentation framework using the SFFCM.

information of pixels is integrated into the objective function of clustering algorithms. The other one is that the color histogram is integrated into the objective function to achieve fast clustering due to the fact that the number of different pixels in a color image has been effectively reduced.

The objective function of SFFCM is

$$J = \sum_{l=1}^{N'} \sum_{k=1}^{c} S_l u_{kl}^m \left\| \left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p \right) - v_k \right\|^2 \tag{3}$$

where ∂_l represents the lth superpixel area, S_l is the total number of pixels in the superpixel area ∂_l, $1 \leqslant l \leqslant N'$. N' is the total number of superpixel area in an image f, and c is the number of clusters. The u_{kl} represents the fuzzy membership of the lth superpixel area with respect to the kth cluster, v_k denotes the kth clustering center, and x_p denotes a pixel in a color image f.

According to (3), the membership partition matrix u_{kl} and the cluster center v_k of the SFFCM are given as follows:

$$u_{kl} = \frac{\left\| \left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p \right) - v_k \right\|^{-2/(m-1)}}{\sum_{j=1}^{c} \left\| \left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p \right) - v_j \right\|^{-2/(m-1)}} \tag{4}$$

$$v_k = \frac{\sum_{l=1}^{N'} u_{kl}^m \left(\sum_{p \in \partial_l} x_p \right)}{\sum_{l=1}^{N'} S_l u_{kl}^m}. \tag{5}$$

It can be seen from (3)–(5) that the computational cost of the SFFCM is clearly lower than the FCM due to the fact that $N' \ll N$. Therefore, the SFFCM achieves fast and effective color image segmentation. The image segmentation framework based on the SFFCM is shown in Fig. 3.

In Fig. 3, the segmentation result is better than the result shown in Fig. 1. The main reason can be attributed to the superpixel image generated by the MMR-WT. The result further demonstrates the advantages of the SFFCM. Although the SFFCM achieves fast and robust image segmentation, the number of clusters is still an essential parameter, which limits the application of the SFFCM. Moreover, the use of Euclidean distance to measure the similarity between different superpixel areas in the SFFCM causes erroneous segmentation results as shown in Fig. 4.

For the problem shown in Fig. 4, hidden Markov random fields (HMRF) [38], [39] is a popular algorithm for overcoming the problem. HMRF uses cluster centers and the prior probability of membership to obtain the final membership called posterior probability [40]. Motivated by this, in this work, we employ fuzzy clustering based on prior entropy to achieve image segmentation. The detailed analysis is presented in Section III-B.

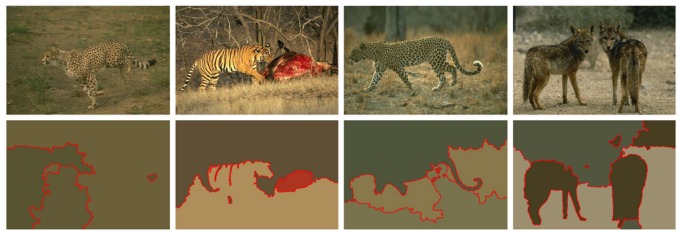


Fig. 4. Error segmentation results using the SFFCM ($c = 3$ for each image).

III. Methodology

In Section II, we presented the motivation of this paper. We employ parameter-free clustering to obtain automatically the number of clusters as well as employ the image superpixel and fuzzy clustering based on prior entropy to achieve image segmentation. Based on these two ideas, we present the stages of segmentation framework of images using the proposed AFCF as shown in Fig. 5.

First, the proposed AFCF employs a superpixel algorithm to obtain presegmentation result. Second, the DP algorithm is performed on the superpixel image to generate a decision-graph. Because the number of areas in the superpixel image is much smaller than the number of pixels in the original image, a small similarity matrix is obtained resulting in a small memory requirement and a low computational complexity for the DP algorithm. After that, the density balance algorithm is used to obtain a more robust decision-graph that directly outputs the number of clusters. By computing the maximal interval of adjacent points, the points in the decision-graph are divided into two groups; the first group of points is considered as cluster centers. Finally, a fuzzy clustering based on prior entropy is used for achieving image segmentation.

A. Decision-Graph on Superpixel Images

The DP algorithm achieves semiautomatic clustering because one can choose the number of clusters according to a decision-graph, but it generates a huge similarity matrix resulting in memory overflow and high computational cost as well as it ignores the spatial information of images.

Because a superpixel algorithm can smooth the texture details and preserve the structuring information of objects, we employ it to reduce the size of the similarity matrix. Then we use a density balance algorithm and the maximal interval to obtain automatically the number of clusters. In practical applications, because different superpixel algorithms can be selected, we only propose a framework of automatic fuzzy clustering but not a specific algorithm in this paper. Fig. 6 shows superpixel results

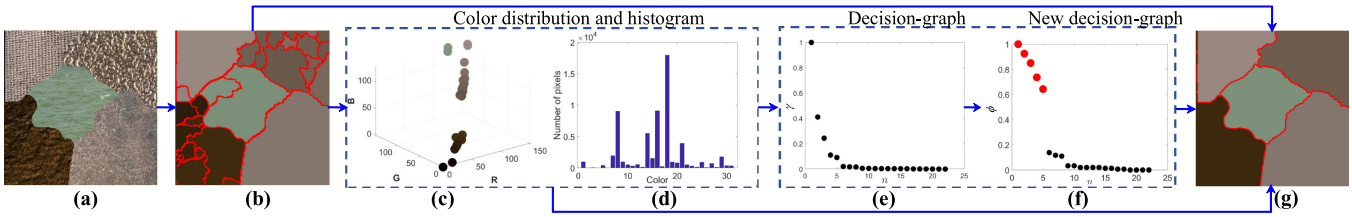


Fig. 5. The proposed stages of the image segmentation framework, which is fully automatic for image segmentation. (a) Image. (b) Superpixel image. (c) Color distribution of the superpixel image. (d) Histogram of the superpixel image. (e) Decision-graph obtained by the DP algorithm. (f) Improved decision-graph using the density balance algorithm. (g) Segmentation result using the FCM based on prior entropy.

Fig. 6. Superpixel results provided by different superpixel algorithms. (a) Image. (b) SLIC. (c) DBSCAN. (d) LSC. (e) GMMSP. (f) HS. (g) MMGR-WT. The number of superpixel area is 200 for the SLIC, DBSCAN, LSC, and HS. For the GMMSP, the size of areas is 20×20. For the MMGR-WT, $r_1 = 2$ and $\eta = 0.1$.

TABLE I
COMPARISON OF SIMILARITY MATRIX USING DIFFERENT SUPERPIXEL ALGORITHMS FOR AN IMAGE SIZE OF 321×481

Algorithms	SLIC	DBSCAN	LSC	GMMSP	HS	MMGR-WT
Number of areas	200	137	182	168	200	147

provided by different superpixel algorithms such as SLIC [41], DBSCAN [42], LSC [43], GMMSP [44], HS [45], and MMGR-WT [23]. Note that each of the SLIC, DBSCAN, LSC, and HS requires one parameter, i.e., the number of superpixel area; the GMMSP also requires one parameter that is the size of areas; but the MMGR-WT needs two parameters that include the initial structuring element denoted by r_1 and the minimal threshold error denoted by η. Table I shows the number of areas in different superpixel images for an image size of 321×481. In practical applications, η is usually a constant and $\eta = 10^{-4}$ in [23]. We set $\eta = 0.1$ here in order to obtain more superpixel areas for fair comparison.

Table I shows that these superpixel algorithms can efficiently reduce the total number of pixels in an image and thus obtain a small similarity matrix to improve the computational efficiency of the DP algorithm. Moreover, a superpixel area integrates both the color features and the spatial structuring features, which is helpful for improving image segmentation results. For instance, the size of the similarity matrix is reduced from $(321 \times 481)^2$ to 200×200 using the SLIC as shown in Fig. 6(b).

According to the aforementioned superpixel algorithms and the DP algorithm, the local density denoted by ρ_I and the minimal distance denoted by δ_I are presented as follows:

$$\rho_I = \sum_{J=1, J\neq I}^{N'} S_J e^{-\frac{D_{IJ}^2}{d_c}} \tag{6}$$

where $1 \leq I, J \leq N'$, D_{IJ} denotes the Euclidean distance between ∂_I and ∂_J. S_J is the total number of pixels in the Jth superpixel area, d_c is the cutoff distance, and δ_I indicates the

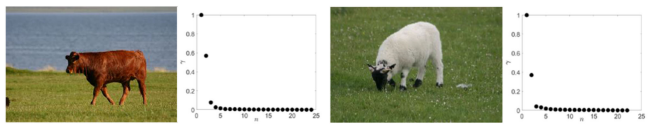


Fig. 7. Decision-graphs according to the DP algorithm and MMGR-WT. (a) Image 1. (b) Decision-graph of image 1. (c) Image 2. (d) Decision-graph of image 2.

minimal distance between the area ∂_I and any other area with higher density. The δ_I is defined as

$$\delta_I = \underbrace{\min}_{J:\rho_J > \rho_I} (D_{IJ}) \tag{7}$$

where $\delta_I = \max_J(D_{IJ})$ for the superpixel area with the highest density. To speed up the computation, D_{IJ} is defined as

$$D_{IJ} = \left\| \frac{1}{S_I} \sum_{p \in \partial_I} x_p - \frac{1}{S_J} \sum_{q \in \partial_J} x_q \right\|. \tag{8}$$

It can be seen that D_{IJ} is different from d_{ij}. According to the DP algorithm and $\gamma_i = \rho_i \delta_i$, we can obtain the initial decision-graph as shown in Fig. 7. Fig. 7 shows that although one can easily select the number of clusters depending on the decision-graph, it is difficult to obtain automatically the number of clusters by setting a threshold for the decision-graph. In Fig. 7(a), the threshold ranges from 0.15 to 0.56, while it ranges from 0.1 to 0.38 for Fig. 7(b).

B. Number of Clusters

To achieve fully automatic clustering, we propose a new decision-graph by using a density balance algorithm. We only compute the maximal interval in the new decision-graph, without having to select the number of clusters.

Algorithm 1: The Density Balance Algorithm.

Input: γ_j
Output: ϕ_j
Initialization: set $a = 1000$ and $\eta = 0.1, \chi = \chi_1, \chi_2, \ldots \chi_{a+1}$, where $\chi_1 = 0, \chi_2 = 0.001, \chi_3 = 0.002$, and $\chi_{a+1} = 1$.
for $j = 1$ to N', **do**
  for $e = 1$ to $a + 1$, **do**
   while $\chi_e \leq \gamma_j$ **do**
    for $b = 1$ to N', **do**
     if $\|\chi_e - \gamma_b\| \leq \kappa$ **then**
      $\varphi_b = 1$
     else
      $\varphi_b = 0$
     end if
     $\xi(\chi_e) = \xi(\chi_e) + \varphi_b$
    end for
    $\phi_j = \phi_j + \xi(\chi_e)$
   end while
  end for
end for

The density balance algorithm aims to map the original decision-graph γ_j into a new decision-graph ϕ_j that is superior to γ_j for finding the best number of clusters, where $1 \leq j \leq q$, where q denotes the number of superipixel area. Let a denote the number of intervals in the range [0,1] and κ denote the radius of the neighborhood, where $\chi = \{\chi_1, \chi_2, \ldots, \chi_{a+1}\}$ represents the set of data interval, $\chi_1 = 0, \chi_2 = 1/a, \chi_2 = 2/a$, and $\chi_{a+1} = 1$. Generally, a is chosen empirically and let $a = 1000$. We define that $\xi(\chi_\kappa)$ is the number of γ_j under the constraint condition $\|\chi_a - \gamma_j\| \leq \kappa$, where $1 \leq e \leq a + 1$ and γ_j is the normalized result $\gamma_j = (\gamma_j - \min(\gamma_j))/(\max(\gamma_j) - \min(\gamma_j))$, $e, a \in N^+$. $\xi(\chi_e)$ can be computed as follows:

$$\xi(\chi_e) = \sum_{j=1}^{N'} \varphi_j \tag{9}$$

$$\varphi_j = \begin{cases} 1, & \|\chi_e - \gamma_j\| \leq \kappa \\ 0, & \text{otherwise} \end{cases}. \tag{10}$$

We define that ϕ_j is the mapping result of γ_j, where ϕ_j can be computed as follows:

$$\phi_j = \sum_{\chi_e \leq \gamma_j} \xi(\chi_e). \tag{11}$$

We presented the pseudocode of the algorithm as recorded in Algorithm 1.

According to the Algorithm 1, we compute the new decision-graphs corresponding to Fig. 7(a) and (c). Fig. 8 shows the results.

Comparing Figs. 7 and 8, it is clear that the proposed density balance algorithm is useful for finding the best number of clusters as it is able to distinguish efficiently cluster centers and noncluster centers.

C. Prior Entropy-Based Fuzzy Clustering

According to Section III-B, we can obtain the number of clusters and the final clustering result using the superpixel-based DP algorithm. However, in [28], Euclidean distance is used to measure the similarity between different superpixel areas, which often leads to poor segmentation results for complex images as shown in Fig. 4. Since both the covariance analysis and the Markov random field (MRF) are useful for improving high-dimensional data classification [46], [47], we integrate them into the FCM algorithm, and propose prior entropy-based fuzzy clustering algorithm (PEFC).

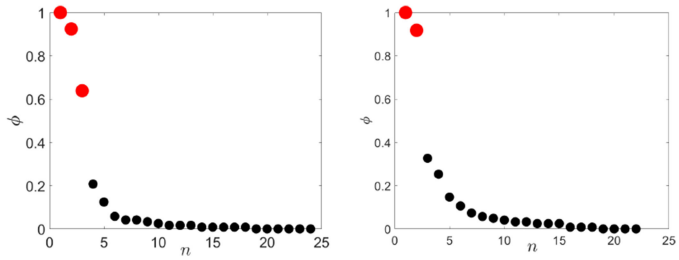


Fig. 8. New decision-graphs of Fig. 7(a) and (c).

Based on superpixel results of images and the number of clusters provided by the Algorithm 1, we propose the objective function of PEFC as follows:

$$J = \sum_{l=1}^{N'} \sum_{k=1}^{c} S_l u_{kl} \Phi\left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p | v_k, \Sigma_k\right) + \sum_{l=1}^{N'} \sum_{k=1}^{c} S_l u_{kl} \log\left(\frac{u_{kl}}{\pi_k}\right) \tag{12}$$

where $\frac{1}{S_l} \sum_{p \in \partial_l} x_p$ is the mean value of the superpixel area ∂_l. The Σ_k is the covariance matrix with respect to the correlation of different dimensions. The proportion π_k is the prior probability of the superpixel area $\frac{1}{S_l} \sum_{p \in \partial_l} x_p$ belonging to v_k, which satisfies $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{c} \pi_k = 1$. The π_k indicates that the MRF is integrated into the objective function of the PEFC. It is clear that the objective function of the PEFC has a more complex structure than the FCM, as the PEFC fuses multiple factors that influence image segmentation results.

Furthermore, c is the number of cluster prototypes, which is provided by Algorithm 1, v_k is the kth cluster center, u_{kl} represents the membership intensity of the superpixel $\frac{1}{S_l} \sum_{p \in \partial_l} x_p$ with respect to the kth clustering center v_k, which satisfies $0 \leq u_{kl} \leq 1$ and $\sum_{k=1}^{c} u_{kl} = 1$, and S_l denotes the number of pixels within the lth superpixel area ∂_l.

In (12), $\Phi(\frac{1}{S_l} \sum_{p \in \partial_l} x_p | v_k, \Sigma_k)$ denotes the multivariate Gaussian distribution, which is defined as follows:

$$\Phi\left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p | v_k, \Sigma_k\right) = -\log \rho\left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p | v_k, \Sigma_k\right). \tag{13}$$

In (13), ρ is the Gaussian density function, which is presented as follows:

$$\rho\left(\frac{1}{S_l} \sum_{p \in \partial_l} x_p | v_k, \Sigma_k\right) = \frac{\left[\exp(-\frac{1}{2}((\frac{1}{S_l} \sum_{p \in \partial_l} x_p) - v_k)^T \times \sum_k^{-1}((\frac{1}{S_l} \sum_{p \in \partial_l} x_p) - v_k))\right]}{(2\pi)^{D/2} |\sum_k|^{1/2}} \tag{14}$$

where D denotes the dimension of image data or the number of image channel, Σ_k is a diagonal matrix of size $D \times D$, and $|\Sigma_k|$ is the determinant of Σ_k. Substituting (14) into (13), we get

$$\Phi\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p|v_k,\Sigma_k\right) = \frac{1}{2}\left(\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p - v_k\right)^T \Sigma_k^{-1}\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p - v_k\right) + \log|\Sigma_k| + Dlog(2\pi)\right) \tag{15}$$

According to $\sum_{k=1}^{c} u_{kl} = 1$ and $\sum_{k=1}^{c}\pi_k = 1$, we use the Lagrange multiplier approach to compute the optimal value, and we have

$$J = \sum_{l=1}^{N'}\sum_{k=1}^{c} S_l u_{kl}\Phi\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p|v_k,\Sigma_k\right) + \sum_{l=1}^{N'}\sum_{k=1}^{c} S_l u_{kl}\log\left(\frac{u_{kl}}{\pi_k}\right) - \lambda_1\left(\sum_{k=1}^{c} u_{kl} - 1\right) - \lambda_2\left(\sum_{k=1}^{c}\pi_k - 1\right) \tag{16}$$

where λ_1 and λ_2 are Lagrange multipliers. Thus, we have

$$\frac{\partial J}{\partial u_{kl}} = S_l\Phi\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p|v_k,\Sigma_k\right) + S_l\left(\log\left(\frac{u_{kl}}{\pi_k}\right) + 1\right) - \lambda_1 = 0.$$

According to $\sum_{k=1}^{c} u_{kl} = 1$, the solution of $\frac{\partial J}{\partial u_{kl}} = 0$ yields

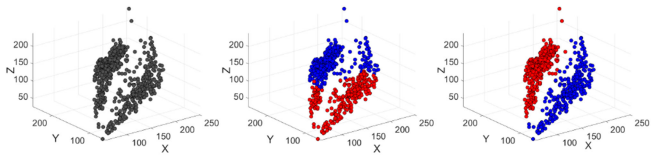
$$u_{kl} = \frac{\pi_k \exp(-\Phi(\frac{1}{S_l}\sum_{p\in\partial_l} x_p|v_k,\Sigma_k))}{\sum_{h=1}^{c}\pi_h\exp(-\Phi(\frac{1}{S_l}\sum_{p\in\partial_l} x_p|v_h,\Sigma_h))}. \tag{17}$$

By computing $\frac{\partial J}{\partial v_k} = 0$, we have

$$\frac{\partial J}{\partial v_k} = \sum_{l=1}^{N'} S_l u_{kl}\left(\frac{\partial\left[\begin{array}{c}((\frac{1}{S_l}\sum_{p\in\partial_l} x_p) - v_k)^T \\ \times\Sigma_k^{-1}((\frac{1}{S_l}\sum_{p\in\partial_l} x_p) - v_k)\end{array}\right]}{\partial v_k}\right) = \sum_{l=1}^{N'} S_l u_{kl}\left(\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p\right) - v_k\right) = 0$$

and

$$v_k = \frac{\sum_{l=1}^{N'} u_{lk}\sum_{p\in\partial_l} x_p}{\sum_{l=1}^{N'} u_{lk}S_l}. \tag{18}$$



(a) (b) (c)

Fig. 9. Comparison of data classification using FCM and PEFC, respectively. (a) The 3-D data. (b) FCM. (c) PEFC.

Similarly, we compute $\frac{\partial J}{\partial \Sigma_k} = 0$

$$\frac{\partial J}{\partial \Sigma_k} = \sum_{l=1}^{N'}\sum_{k=1}^{c}\frac{1}{2}u_{kl}S_l\left(\frac{\partial\left[\begin{array}{c}((\frac{1}{S_l}\Sigma_{p\in\partial_l} x_p) - v_k)^T \\ \times\Sigma_k^{-1}((\frac{1}{S_l}\Sigma_{p\in\partial_l} x_p) - v_k)\end{array}\right]}{\partial\Sigma_k} + \frac{\partial log|\Sigma_k|}{\partial\Sigma_k}\right)$$

$$= \sum_{l=1}^{N'} S_l u_{kl}\left(-\left(\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p\right) - v_k\right)^T \Sigma_k^{-2}\left(\left(\frac{1}{S_l}\sum_{p\in\partial_l} x_p\right) - v_k\right) + \Sigma_k^{-1}\right) = 0$$

The solution of $\frac{\partial J}{\partial \Sigma_k} = 0$ yields

$$\Sigma_k = \frac{\sum_{l=1}^{N'} S_l u_{kl}((\frac{1}{S_l}\sum_{p\in\partial_l} x_p) - v_k)^T((\frac{1}{S_l}\sum_{p\in\partial_l} x_p) - v_k)}{\sum_{l=1}^{N'} S_l u_{kl}}. \tag{19}$$

Finally, by computing $\frac{\partial J}{\partial \pi_k} = 0$,

$$\frac{\partial J}{\partial \pi_k} = -\sum_{l=1}^{N'} S_l\frac{u_{kl}}{\pi_k} - \lambda_2 = 0$$

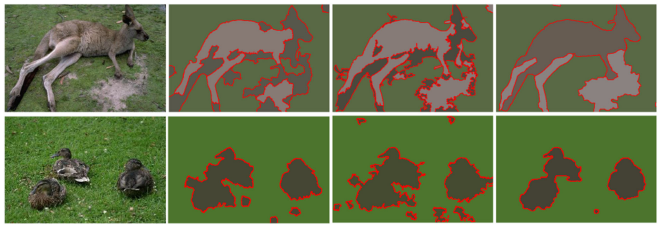
according to $\sum_{k=1}^{c}\pi_k = 1$, we have

$$\pi_k = \frac{\sum_{l=1}^{N'} u_{kl}S_l}{\sum_{l=1}^{N'} S_l}. \tag{20}$$

According to (17)–(20), we obtain a membership partition matrix $\mathbf{U} = [u_{kl}]^{c\times N'}$, the cluster center $\mathbf{V} = [v_k]^{c\times D}$, the covariance matrix $\mathbf{\Sigma} = [\Sigma_k]^{c\times(D\times D)}$, and the proportion $\boldsymbol{\pi} = [\pi_k]^{c\times 1}$.

We can see from (17)–(20) that the PEFC integrates the adaptive neighboring information of prior probability distribution, and it considers the distribution characteristic of data. Therefore, it is often used to divide high-dimensional data into different groups. Fig. 9 shows an example where the PEFC is used for the classification of 3-D data.

Based on the analysis of Section III-A–III-C so far, we propose the detailed steps of the AFCF in the following. The



(a) (b) (c) (d)

Fig. 10. Segmentation results using the proposed AFCF. (a) Images. (b) SLIC-AFCF. (c) LSC-AFCF. (d) MMGR-AFCF. The number of superpixel area is 200 for SLIC and LSC. For MMGR-WT, $r_1 = 2$ and $\eta = 10^{-4}$.

AFCF mainly includes three parts: superpixel pre-processing, the improved DP algorithm based on Algorithm 1, and the PEFC.

The detailed steps of the AFCF[1] are as follows.

Step 1: Compute the superpixel result of the original image, where each area of the superpixel image is denoted by the mean value, $\hat{x}_l = \frac{1}{S_l}\sum_{p\in\partial_l} x_p$. As a result, we get a data set to be classified $\hat{x} = \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{N'}\}$ and a corresponding color histogram $S = \{S_1, S_2, \ldots, S_{N'}\}$, where S_l denotes the total number of pixels in the superpixel area ∂_l;

Step 2: Implement the DP algorithm on $\hat{x}$ to obtain γ_j;

Step 3: Implement Algorithm 1 to obtain ϕ_j and the number of clusters;

Step 4: Initialize the variables $\mathbf{U}^{(0)}, \mathbf{V}^{(0)}, \mathbf{\Sigma}^{(0)}$, and $\boldsymbol{\pi}^{(0)}$ using FCM algorithm, where the weighting exponent, the convergence condition, and the maximal number of iterations are 2, 10^{-5}, and 50, respectively;

Step 5: Set the loop counter $t = 0$;

Step 6: Update the variables $\mathbf{U}^{(t)}, \mathbf{V}^{(t)}, \mathbf{\Sigma}^{(t)}$, and $\boldsymbol{\pi}^{(t)}$;

1) Update the membership matrix $\mathbf{U}^{(t)}$ using (17).
2) Update the cluster center $\mathbf{V}^{(t)}$ using (18).
3) Update the covariance $\mathbf{\Sigma}^{(t)}$ using (19).
4) Update the prior probability $\boldsymbol{\pi}^{(t)}$ using (20).

Step 7: If $\max\{\mathbf{U}^{(t)} - \mathbf{U}^{(t+1)}\} < 10^{-5}$, then stop, otherwise, set $t = t + 1$ and go to step 6.

In this paper, three excellent superpixel algorithms, namely SLIC, LSC, and MMGR-WT, are used for the AFCF to obtain three automatic image segmentation algorithms, i.e., SLIC-AFCF, LSC-AFCF, and MMGR-AFCF. Fig. 10 shows the segmentation results generated by these three algorithms. It can be seen that they obtain the same number of clusters but offer different segmentation results. The MMGR-AFCF obtains a better result than the SLIC-AFCF and LSC-AFCF, for the MMGR-WT generates a better superpixel result than the SLIC and LSC.

IV. Experiments

We conducted experiments on two types of images: synthetic color images with complex texture information (image size is 256×256), as well as real images from the Berkeley segmentation dataset and benchmark (BSDS500) [48] (image size is 481×321 or 321×481). Two synthetic images include three and four different colors and textures, respectively. The BSDS500 includes 300 training images and 200 testing images. There are 4–9 ground truth segmentations for each image in BSDS500, and each ground truth is delineated by one human subject. All algorithms and experimental evaluations are performed on a workstation with an Intel(R) Core (TM) CPU, i7-6700, 3.4 GHz, and 16 GB RAM.

To evaluate the effectiveness and efficiency of the proposed AFCF, ten popular clustering-based image segmentation algorithms are considered for comparisons. These include FCM [49], FGFCM [19], HMRF-FCM [40], FLICM [12], NWFCM [13], Liu's algorithm [16], NDFCM [50], FRFCM [20], DSFCM_N [15], and SFFCM [23]. In addition, because the proposed AFCF is an image segmentation framework, three different algorithms, SLIC-AFCF, LSC-AFCF, and MMGR-AFCF, have been considered in our experiments.

A. Parameter Setting

Comparative algorithms and the proposed AFCF require three indispensable parameters, namely, the weighting exponent, the minimal error threshold, and the maximal number of iterations. In our experiments, the values of these parameters are 2, 10^{-5}, and 50, respectively. The parameter setting of comparative algorithms follows the original paper. As all comparative algorithms require a neighboring window except the FCM, HMRF-FCM, FLICM, Liu's algorithm, and SFFCM, a window of size 3×3 is used for those algorithms that require a neighboring window for fair comparison. The spatial scale factor and the gray-level scale factor in the FGFCM are $\lambda_s = 3$ and $\lambda_g = 5$, respectively. The NWFCM only refers to the gray-level scale factor, $\lambda_g = 5$. The three parameters, the spatial bandwidth $h_s = 10$, the range bandwidth $h_r = 10$, and the minimum size of final output regions $h_k = 100$, are used for Liu's algorithm. Except three indispensable parameters mentioned above and the number of the cluster prototypes, the FCM, HMRF-FCM, FLICM, and DSFCM_N do not require any other parameters. In the FRFCM, both the structuring element and the filtering window are a square of size 3×3 for fair comparison. For the SFFCM and the proposed MMGR-AFCF, they share two same parameters used for the MMGR-WT, $r_1 = 2$, and $\eta = 10^{-4}$. For the proposed SLIC-AFCF and LSC-AFCF, the number of superpixel areas is set to 400 here.

B. Results on Synthetic Images

We demonstrate that the proposed framework is able to provide accurate number of clusters and achieve better image segmentation than comparative algorithms. Two synthetic images are considered as testing images, where we use the texture information of the Colored Brodatz Texture database[2] to generate complex texture images. Then these two images are corrupted by two kinds of noise–Gaussian noise and Salt & Pepper noise. The final segmentation results are shown in Figs. 11 and 12.

Note that all comparative algorithms require the number of clusters except the proposed AFCF since it is fully automatic for image segmentation. Figs. 11(c) and 12(c) show the final decision-graph and it is clear that the number of clusters c is 3 for Fig. 11 and c is 4 for Fig. 12, which demonstrates that the proposed AFCF is effective for finding the best value of c. For fair comparison, $c = 3$ and $c = 4$ are used for all comparative algorithms in Fig. 11(a) and (b), respectively.

[1]Source code is available at https://github.com/SUST-reynole/AFCF

[2][Onlne]. Available: http://multibandtexture.recherche.usherbrooke.ca/colored%20_brodatz.html

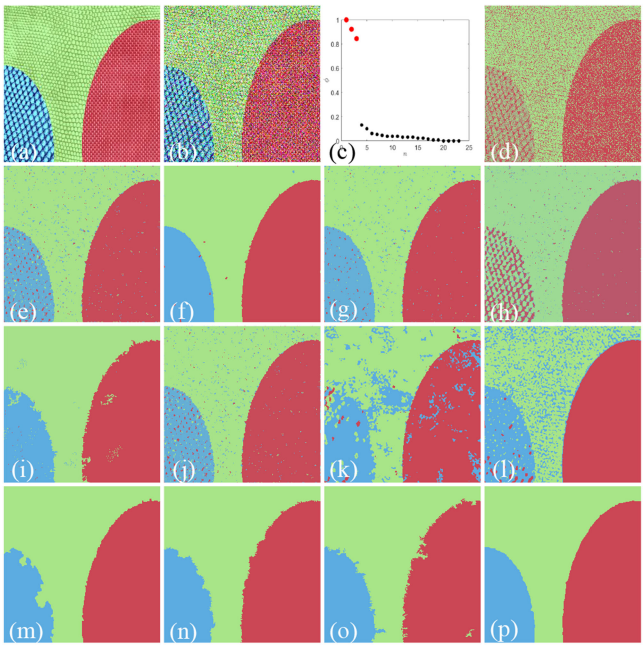


Fig. 11. Comparison of segmentation results on the first synthetic image. (a) The original synthetic image including three areas with different texture and color. (b) The noisy image corrupted by the mixture noise (the mean value is zero and the variance is 0.05 for the Gaussian noise; the density of Salt & Pepper is 0.1). (c) The decision-graph. (d) FCM. (e) FGFCM. (f) HMRF-FCM. (g) FLICM. (h) NWFCM. (i) Liu's algorithm. (j) NDFCM. (k) FRFCM. (l) DSFCM_N. (m) SFFCM. (n) SLIC-AFCF. (o) LSC-AFCF. (p) MMGR-AFCF.

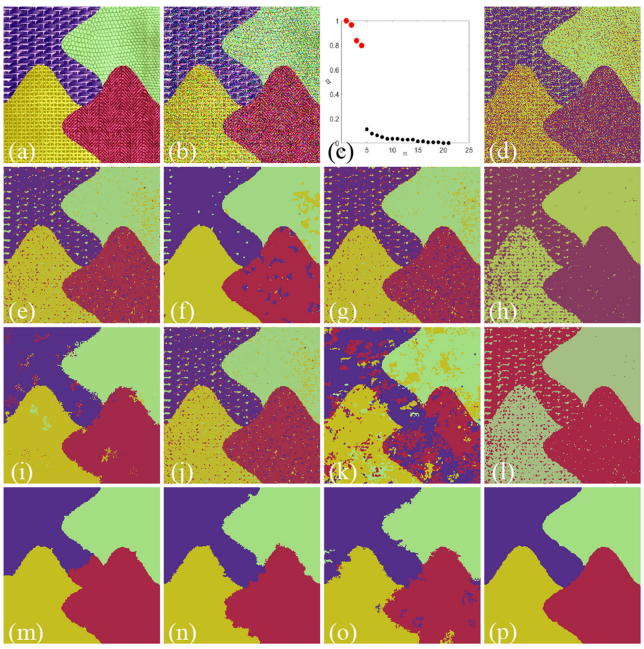


Fig. 12. Comparison of segmentation results on the second synthetic image. (a) The original synthetic image including four areas with different texture and color. (b) The noisy image corrupted by the mixture noise (the mean value is zero and the variance is 0.05 for the Gaussian noise; the density of Salt & Pepper is 0.1). (c) The decision-graph. (d) FCM. (e) FGFCM. (f) HMRF-FCM. (g) FLICM. (h) NWFCM. (i) Liu's algorithm. (j) NDFCM. (k) FRFCM. (l) DSFCM_N. (m) SFFCM. (n) SLIC-AFCF. (o) LSC-AFCF. (p) MMGR-AFCF.

Fig. 11(d) shows that the FCM provides a poor segmentation result since the spatial information of images is missing. Although the FRFCM and DSFCM_N integrate spatial information of images into their objective function, lots of pixels are classified falsely as shown in Fig. 11(k) and (l). The FRFCM obtains a poor result, since the multivariate morphological reconstruction is unsuitable for images with complex texture. The DSFCM_N employs the sparse representation to improve image segmentation results, but it is not effective for images corrupted by the mixture noise. Fig. 11(e), (g), (h), and (j) provides good segmentation results because the neighboring information employed by the FGFCM, FLICM, NWFCM, and NDFCM is effective for improving segmentation results. Furthermore, Liu's algorithm and the HRMF-FCM provide better results, which show that they are robust for images corrupted by noise. The AFCF has a strong capability of noise suppression, and the MMGR-AFCF obtains the best result that looks similar to the result provided by the HMRF-FCM.

Fig. 12(a) has a more complex shape and texture than Fig. 11(a). In Fig. 12(d), (e), and (l), the FCM, FGFCM, and DSFCM_N generate poor segmentation results because they obtain wrong cluster centers, which indicates that the three algorithms have a weak capability of pixel classification for complex images corrupted by noise. Fig. 12(f)–(j) offers similar results, which means that HMRF-FCM, FLICM, NWFCM, NDFCM, and Liu's algorithm have a limited capability for improving segmentation effect on complex images. Both the SFFCM and AFCF obtain better segmentation results as shown in Fig. 12(m)–(p), which shows that the superpixel technology and the prior entropy are effective for improving segmentation results on complex images. Similar to Fig. 11(p), the MMGR-AFCF obtains the best segmentation result as shown in Fig. 12(p).

To evaluate the performance of different algorithms on noisy images, two performance indices are employed here. The first is the quantitative index score (S) that is the degree of equality between pixel sets A_k and the ground truth C_k [20]. The second is the optimal segmentation accuracy (SA) that is the sum of the number of the correctly classified pixels divided by the number of ground truth pixels. They are defined as

$$S = \sum_{k=1}^{c} \frac{A_k \bigcap C_k}{A_k \bigcup C_k} \tag{21}$$

$$SA = \sum_{k=1}^{c} \frac{A_k \bigcap C_k}{\sum_{j=1}^{c} C_j} \tag{22}$$

where A_k is the set of pixels belonging to the kth class found by an algorithm while C_k is the set of pixels belonging to the class in the ground truth. All algorithms are performed on the two synthetic images corrupted by different kinds and levels of noise. The experimental results are shown in Tables II and III. As the algorithms used in experiments show different performance for images corrupted by different kinds of noise, we further presented the mean value and the root-mean-square error (RMSE) of S and SA in Tables II and III.

In Tables II and III, the FCM obtains low mean value and high RMSE of S and SA because it is sensitive to both Salt & Pepper noise and Gaussian noise. The NWFCM, FRFCM, and DSFCM_N show better performance than the FCM, but they are poor compared to other comparative algorithms, especially

TABLE II
SCORES (S%) ON THE FIRST SYNTHETIC IMAGE CORRUPTED BY NOISE ($c = 3$), WHERE SP REPRESENTS SALT & PEPPER NOISE AND G REPRESENTS GAUSSIAN NOISE

Algorithms	SP 10%	SP 20%	SP 30%	G 5%	G 10%	G 15%	SP 10% +G 5%	SP 20% +G 10%	SP 30% +G 15%	Mean Value	RMSE
FCM	84.24	71.43	66.91	73.50	67.76	64.52	69.57	61.84	57.03	68.53	7.72
FGFCM	98.88	96.52	89.40	97.92	93.98	89.18	96.08	83.78	74.71	91.16	7.89
HMRF-FCM	93.69	90.58	88.13	99.84	95.30	89.87	99.80	96.46	83.10	92.97	5.57
FLICM	99.07	94.77	84.05	99.14	96.90	92.85	97.71	85.99	78.68	92.13	7.44
NWFCM	94.77	90.14	82.11	86.56	82.35	81.03	84.48	79.06	74.63	83.90	6.00
Liu's algorithm	99.68	98.43	97.75	99.35	97.65	94.68	97.76	80.17	68.40	92.65	10.91
NDFCM	98.90	96.56	89.32	98.06	94.31	89.53	96.23	84.01	74.94	91.32	7.84
FRFCM	95.75	86.13	85.09	84.40	78.55	76.53	84.13	70.45	64.08	80.57	9.38
DSFCM_N	99.88	86.47	85.84	89.80	81.89	80.24	87.08	78.68	73.99	84.87	7.46
SFFCM	99.29	96.49	93.25	97.80	97.70	95.92	97.78	90.83	90.36	95.49	3.24
SLIC-AFCF	99.48	98.94	98.92	99.21	98.81	98.46	99.16	98.39	87.36	97.64	3.87
LSC-AFCF	99.67	99.41	98.42	98.75	98.19	91.39	97.51	93.83	75.91	94.79	7.60
AMR-AFCF	99.67	99.58	99.39	99.66	99.38	99.24	99.58	99.38	99.23	**99.46**	**0.17**

The best values are provided in bold.

TABLE III
SEGMENTATION ACCURACIES (SA%) OF DIFFERENT ALGORITHMS ON THE SECOND SYNTHETIC IMAGE CORRUPTED BY NOISE ($c = 4$), WHERE SP REPRESENTS SALT & PEPPER NOISE AND G REPRESENTS GAUSSIAN NOISE

Algorithms	SP 10%	SP 20%	SP 30%	G 5%	G 10%	G 15%	SP 10% +G 5%	SP 20% +G 10%	SP 30% +G 15%	Mean Value	RMSE
FCM	64.99	46.69	37.37	42.10	32.54	29.14	35.68	33.81	24.25	38.51	11.93
FGFCM	91.77	82.53	69.84	86.05	74.13	65.71	78.95	59.96	48.89	73.09	13.53
HMRF-FCM	99.61	98.89	64.96	97.03	82.01	66.47	90.07	52.51	39.35	76.78	21.97
FLICM	85.00	67.99	48.77	88.02	73.07	53.96	74.92	40.57	25.49	61.98	21.10
NWFCM	95.61	44.34	38.69	50.50	33.25	32.30	39.82	31.73	30.87	44.12	20.39
Liu's algorithm	98.97	94.30	86.82	97.59	83.53	68.43	91.34	48.43	29.93	77.70	24.11
NDFCM	91.73	82.27	69.62	86.50	74.62	66.30	79.17	60.33	48.86	73.27	13.50
FRFCM	94.49	85.11	53.89	59.31	43.68	36.99	57.80	40.07	37.32	56.52	20.80
DSFCM_N	97.57	96.31	39.09	68.01	48.83	38.37	32.93	30.95	29.48	53.50	27.28
SFFCM	98.79	97.94	90.58	98.14	89.74	87.60	98.59	84.57	71.33	90.80	9.07
SLIC-AFCF	98.39	96.18	96.78	98.52	96.68	95.90	95.03	94.35	84.4	95.14	4.25
LSC-AFCF	98.44	97.83	95.8	95.80	93.54	82.60	95.23	86.21	70.88	90.70	9.15
AMR-AFCF	99.16	98.81	98.37	99.06	98.90	98.50	98.63	97.57	96.96	**98.44**	**0.73**

The best values are provided in bold.

when the noise level is high. The DSFCM_N is robust against Salt & Pepper noise, but it is sensitive to Gaussian noise, and thus it is difficult to obtain a good segmentation result using the DSFCM_N for images corrupted by the mixture noise.

The FGFCM, HMRF-FCM, FLICM, Liu's algorithm, and NDFCM obtain similar average performance for the first synthetic image as shown in Table II, but the FLICM shows worse performance than other four algorithms for the second synthetic image as shown in Table III. The SFFCM shows excellent performance for two synthetic images. We can see that the SFFCM is insensitive to both Salt & Pepper noise and Gaussian noise when the noise level is low, but it is sensitive to Gaussian noise when the noise level is high, as shown in Table III. The proposed AFCF shows better performance than comparative algorithms except the SFFCM. Especially, the MMGR-AFCF obtains the largest mean value as well as the smallest RMSE of S and SA on two synthetic images, which demonstrates that the MMGR-AFCF is robust against different kinds of noise corruption.

C. Results on Real Images

Here we demonstrate the superiority of the AFCF on real image segmentation with images from BSDS500. The parameters values in all algorithms are the same as those in Section IV-B except the FRFCM and SFFCM. The related parameters of the FRFCM and SFFCM follow the original papers [20] and [23]. In addition, the CIE-Lab color space is used for all algorithms for a fair comparison.

To evaluate the performance of different algorithms for real image segmentation, we consider the probabilistic rand index (PRI), the covering (CV), the variation of information (VI), the global consistency error (GCE), and the boundary displacement error (BDE) [51], as the performance metrics. Both PRI and CV are usually used for the evaluation of the pixelwise classification task, where PRI is the similarity of labels and CV is the overlap of regions between two clustering results. VI is also used for the purpose of clustering comparison, and it is the distance of average conditional entropy between two clustering results.

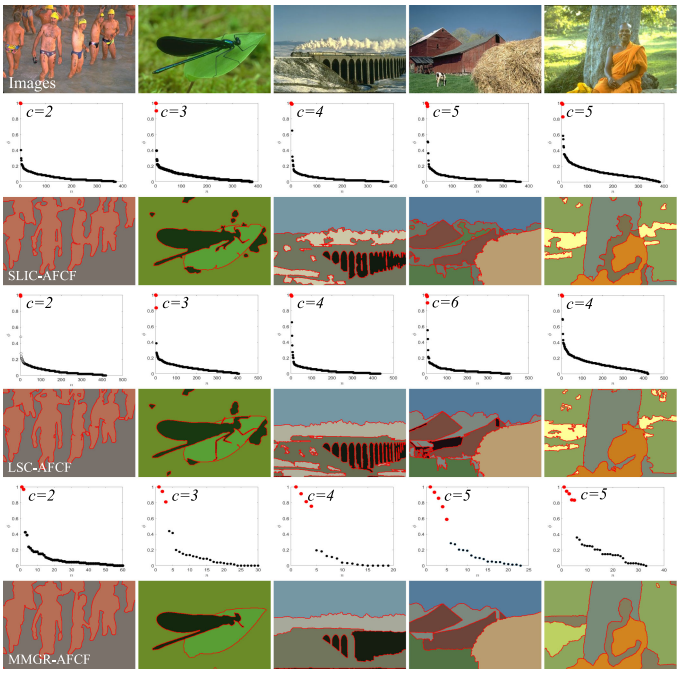


Fig. 13. Segmentation results using the proposed AFCF. The first row denotes original images. The second, fourth, and sixth rows denote decision-graphs obtained by SLIC-AFCF, LSC-AFCF, and MMGR-AFCF, respectively. The third, fifth, and seventh rows are segmentation results generated by the three proposed algorithms.

Additionally, GCE and BDE are often used for the evaluation of image segmentation, where GCE computes the global error to which two segmentations are mutually consistent and BDE measures the average displacement error of boundary pixels between two segmentations. In general, a good segmentation corresponds to high value of PRI and CV, but corresponds to low values of VI, GCE, and BDE.

First, we demonstrate that the proposed AFCF is able to obtain an accurate number of clusters for real images. As it is difficult to give an accurate number of clusters for each image in BSDS500, we choose five simple images as shown in Fig. 13. According to the proposed AFCF that includes the SLIC-AFCF, LSC-AFCF, and MMGR-AFCF, the corresponding decision-graphs and segmentation results are obtained as shown in Fig. 13. We can see that the SLIC-AFCF, LSC-AFCF, and MMGR-AFCF obtain the approximate number of clusters, indicating that the AFCF provides similar number of clusters for an image independent of the selected superpixel algorithm.

In Fig. 13, the SLIC-AFCF and MMGR-AFCF obtain the same number of cluster for each image, but the LSC-AFCF obtains different results for images "97010" and "376020." In practical applications, because three superpixel algorithms lead to different presegmentation results, it is impossible to obtain the same number of clusters for each image in the BSDS500 using three algorithms. However, Fig. 13 shows that three suprepixel algorithms can obtain an approximate number of clusters. Because the MMGR-WT has been able to provide better superpixel results, we consider the number of clusters provided by the MMGR-AFCF for comparative algorithms in the following experiments.

TABLE IV
AVERAGE PERFORMANCE ON FIVE IMAGES SHOWN IN FIG. 13

Algorithms	c	PRI↑	CV↑	VI↓	GCE↓	BDE↓
SLIC-AFCF	c=2	0.66	0.58	2.06	**0.08**	16.76
	c=3	0.81	0.64	1.82	0.13	9.98
	c=4	0.81	0.60	1.85	0.17	9.83
	c=5	0.81	0.54	1.98	0.25	9.02
	c=6	0.80	0.50	2.07	0.28	9.69
	Adaptive c	**0.87**	**0.69**	**1.61**	0.16	**5.91**
LSC-AFCF	c=2	0.66	0.59	0.96	**0.05**	18.99
	c=3	0.81	0.64	1.79	0.13	9.85
	c=4	0.83	0.62	1.78	0.17	10.41
	c=5	0.83	0.58	1.89	0.22	9.57
	c=6	0.82	0.51	2.07	0.29	9.41
	Adaptive c	**0.85**	**0.65**	**1.66**	0.15	**7.14**
MMGR-AFCF	c=2	0.70	0.59	2.01	**0.08**	12.89
	c=3	0.83	0.64	1.75	0.13	8.70
	c=4	0.82	0.60	1.74	0.17	9.57
	c=5	0.83	0.61	1.66	0.17	8.33
	c=6	0.82	0.56	1.83	0.22	9.00
	Adaptive c	**0.90**	**0.74**	**1.32**	0.10	**6.04**

The best values are provided in bold.

TABLE V
PERFORMANCE COMPARISON OF DIFFERENT ALGORITHMS ON THE BSDS500 DATASET

Algorithms	PRI↑	CV↑	VI↓	GCE↓	BDE↓
FCM	0.70	0.41	2.87	0.37	14.01
FGFCM	0.69	0.40	2.92	0.38	14.29
HMRF-FCM	0.72	0.44	2.59	0.33	14.22
FLICM	0.71	0.43	2.73	0.35	13.47
NWFCM	0.71	0.42	2.79	0.36	13.70
Liu's algorithm	0.74	0.47	2.47	0.30	**12.95**
NDFCM	0.69	0.40	2.93	0.38	14.38
FRFCM	0.71	0.44	2.67	0.34	13.52
DSFCM_N	0.70	0.40	2.92	0.38	14.48
SFFCM	0.73	0.50	2.18	0.25	14.13
SLIC-AFCF	0.71	0.47	2.43	0.28	14.55
LSC-AFCF	0.73	0.47	2.44	0.29	13.98
MMGR-AFCF	**0.76**	**0.54**	**2.05**	**0.22**	**12.95**

The best values are provided in bold.

Table V shows the performance of the proposed AFCF on real image segmentation. Because our purpose is to demonstrate that the proposed AFCF can obtain the best number of clusters, we present the average values of PRI, CV, VI, GCE, and BDE on five images shown in Fig. 13. The proposed AFCF can automatically obtain the value of c, which means that the AFCF obtains a value of c adaptively. However, the conventional clustering algorithms use a fixed value of c for each image. In Table IV, the fixed value of c and the adaptive value of c are used for AFCF, respectively. It is clear that the adaptive value of c is superior to the fixed value of c, which demonstrates that the proposed AFCF is effective for obtaining an accurate number of clusters.

Also, we applied all algorithms on each image in the BSDS500, using the same value of c obtained by the proposed MMGR-AFCF for all algorithms. Experimental results on some images are shown in Figs. 14 and 15. Moreover, Table V

Fig. 14. Comparison of segmentation results on six images from the BSDS500 using different algorithms.

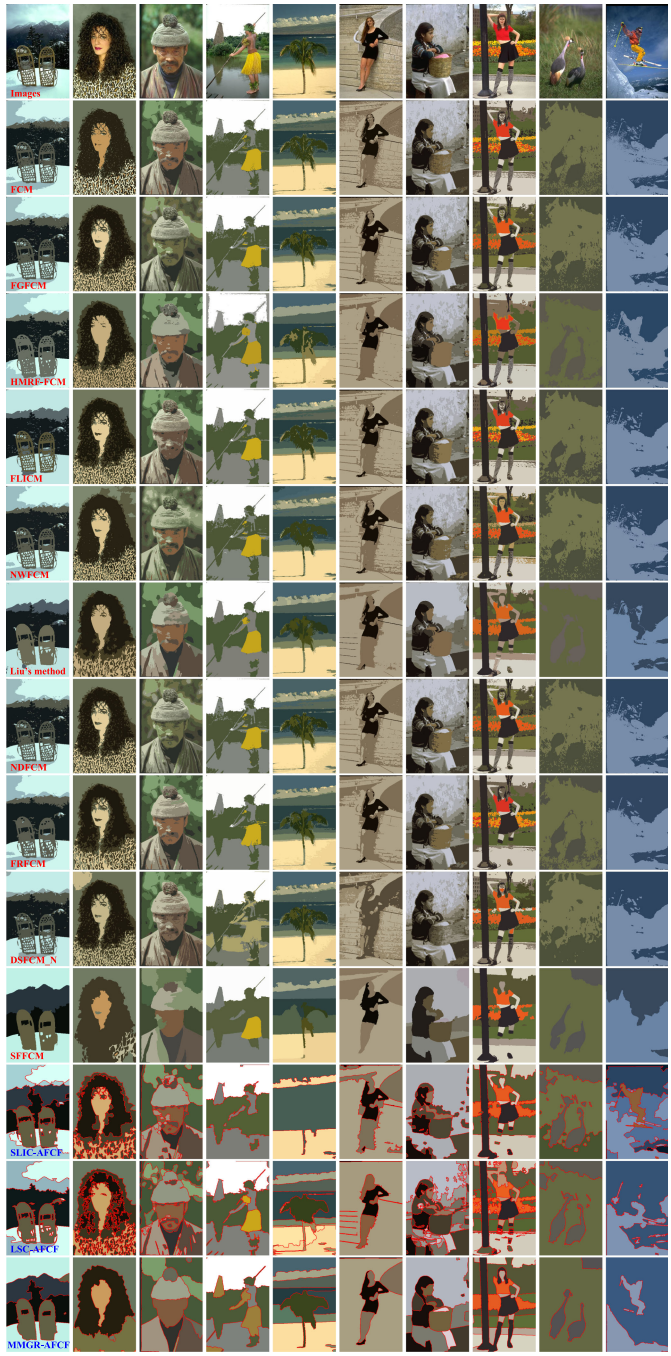


Fig. 15. Comparison of segmentation results on ten images from the BSDS500 using different algorithms.

shows the performance comparison of different algorithms on BSDS500.

In Figs. 14 and 15, we can see that the FCM, FGFCM, HMRF-FCM, FLICM, NWFCM, KWFLICM, NDFCM, FR-FCM, and DSFCM_N generate segmentation results including a great number of isolated regions. Therefore, it is difficult to obtain fine contours for segmentation results. The main reason is that these algorithms employ fixed-size windows to obtain spatial neighboring information. The Liu's algorithm, SFFCM, and the proposed AFCF obtain better segmentation results due to the employment of superpixel algorithms, which means that the adaptive spatial information is useful for improving segmentations. However, because the MMGR-WT always generates better superpixel areas than the SLIC and LSC, the MMGR-AFCF provides better segmentation results than the SLIC-AFCF and LSC-AFCF. Compared to the SFFCM that employs the Sobel operator, the MMGR-AFCF employs the structured forests (SE) [52] to generate a gradient image to be used for the MMGR. Therefore, the MMGR-AFCF provides better superpixel results than the SFFCM. Furthermore, as the former employs prior entropy to obtain the final result, it has clear advantages than the later for image segmentation.

In Table V, the FCM, FGFCM, FLICM, NWFCM, NDFCM, and DSFCM_N obtain similar values of PRI, CV, VI, GCE, and BDE, which shows that these algorithms have similar performance on real image segmentation. Similarly, the HMRF-FCM has a similar performance with the FRFCM. Liu's algorithm clearly outperforms other algorithms due to the employment of superpixel images generated by mean-shift algorithm. The SFFCM obtains better CV, VI, and GCE but worse PRI and

TABLE VI
COMPUTATIONAL COMPLEXITY OF DIFFERENT ALGORITHMS

Algorithms	Computational complexity
FCM	$O(N \times c \times t)$
FRFCM	$O(N \times w^2 + N \times c \times t)$
DSFCM_N	$O(N \times w^2 \times c \times t)$
SFFCM	$O(N \times T' + N' \times c \times t)$
SLIC-AFCF	$O(N \times K \times t' + N' \times c \times t \times 2)$
LSC-AFCF	$O(N \times K \times t' + N' \times c \times t \times 2)$
MMGR-AFCF	$O(N \times T' + N' \times c \times t \times 2)$

TABLE VII
COMPARISON OF EXECUTION TIME (IN SECONDS) OF DIFFERENT ALGORITHMS

Algorithms	FCM	FRFCM	DSFCM_N	SFFCM	SLIC -AFCF	LSC -AFCF	MMGR -AFCF
Time	1.15	1.87	19.15	**0.74**	2.44	2.88	1.05

The best values are highlighted.

BDE than Liu's algorithm. The proposed SLIC-AFCF and LSC-AFCF show similar performance with Liu's algorithm. However, they are superior to Liu's algorithm since the proposed AFCF is fully automatic and it has lower computational complexity than Liu's algorithm. The proposed MMGR-AFCF provides the best results in each of the five performance metrics, which demonstrates that the MMGR-AFCF is able to obtain excellent segmentation results for real images.

D. Computational Cost

The proposed image segmentation framework can be roughly divided into two stages. The first is image superpixel and the second is automatic fuzzy clustering. It is well known that the computational complexity of the SLIC and LSC is $O(N \times K \times t')$, and the MMGR has lower computational complexity $O(N \times T')$ than the SLIC and LSC, where both t' and T' are iterations and they are often smaller than t. For the AFCF, as both the construction of decision-graph and the decision of the number of clusters do not require iterations, they are fast as compared to subsequent fuzzy clustering algorithms. Consequently, the computational complexity of the proposed AFCF is $O(N' \times c \times t)$, where N is the total number of pixels and N' is the total number of superpixel areas. It is clear that AFCF has a lower computational complexity due to $N' \ll N$. Table VI shows the computational complexity of different algorithms.

In Table VI, w denotes the size of neighboring window and K denotes the number of neighboring centers. The DSFCM_N has the highest computational complexity since the neighboring information is computed on each pixel. The SFFCM has the lowest computational complexity since the MMGR is fast and $N' \ll N$.

Furthermore, to estimate the practicability of different algorithms, we computed the execution time of different algorithms on all images in the BSDS500 and Table VII shows the comparison of average execution time on 500 images. It is known that the FLICM, HMRF-FCM, NWFCM, and Liu's algorithm are quite time-consuming [16]. Moreover, the FRFCM has a lower computational complexity than the FGFCM and NDFCM [50]. Therefore, we only present the execution time of the FCM, FRFCM, DSFCM_N, SFFCM, and the proposed SLIC-AFCF, LSC-AFCM, MMGR-AFCF in Table VII. The execution time further demonstrates that the proposed AFCF is fast and it is practical.

In Table VII, we can see that the SFFCM is the fastest due to the utilization of superpixel and color histogram. The DSFCM_N needs a long execution time because the neighboring information is repeatedly computed in each iteration. The proposed SLIC-FCM and LSC-FCM require a similar execution time that is longer than the MMGR-AFCF, for both the SLIC and LSC has a higher computational complexity than the MMGR-WT. Clearly, the proposed MMGR-AFCF is not only faster than all comparative algorithms (except the SFFCM), but also it is fully automatic for image segmentation.

V. CONCLUSION

In this paper, we have studied automatic image segmentation algorithms using fuzzy clustering. We proposed an AFCF for image segmentation by integrating superpixel algorithms, density peak clustering, and prior entropy. The proposed AFCF addresses two difficulties in existing popular algorithms. It is fully automatic for image segmentation since the number of clusters is obtained automatically and it provides better image segmentation results than popular clustering algorithms due to the employment of superpixel algorithms and the prior entropy. The proposed AFCF was used to segment synthetic and real images, demonstrating that it is able to obtain accurate number of clusters. Moreover, the AFCF is superior to state-of-the-art clustering algorithms because it provides the best segmentation results.

However, we only use the average color in a superpixel area as the features of the superpixel area, which is a drawback for image segmentation. In the future, we will consider convolutional neural networks to extract image features and will explore feature learning algorithm to achieve better automatic image segmentation.

REFERENCES

[1] A. Fahad, N. Alshatri, and Z. Tari, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 267–279, Sep. 2014.
[2] M. T. Law, R. Urtasun, and R. S. Zemel, "Deep spectral clustering learning," in *Proc. Int. Conf. Mach. Learn.*, Sydney, Australia, 2017, pp. 1985–1994.
[3] G. Nebehay and R. Pflugfelder, "Clustering of static-adaptive correspondences for deformable object tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Boston, MA, USA, 2015, pp. 2784–2791.
[4] G. Dong and M. Xie, "Color clustering and learning for image segmentation based on neural networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 4, pp. 925–936, Jul. 2005.
[5] N. Kumar, P. Uppala, and K. Duddu, "Hyperspectral tissue image segmentation using semi-supervised NMF and hierarchical clustering," *IEEE Trans. Med. Imag.*, vol. 38, no. 5, pp. 1304–1313, May 2019, doi: 10.1109/TMI.2018.2883301.
[6] K. Zhang, L. Zhang, K. M. Lam, and D. Zhang, "A level set approach to image segmentation with intensity inhomogeneity," *IEEE Trans. Cybern*, vol. 46, no. 2, pp. 546–557, Feb. 2016.
[7] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[8] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1452–1458, Nov. 2004.
[9] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1452–1458, Sep. 2004.

[10] M. N. Ahmed, S. M. Yamany, N. A. Mohamed, A. A. Farag, and T. Moriarty, "A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data," *IEEE Trans. Med. Imag.*, vol. 21, no. 3, pp. 193–199, Mar. 2002.
[11] S. Chen and D. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 34, no. 4, pp. 1907–1916, Aug. 2004.
[12] S. Krinidis and V. Chatzis, "A robust fuzzy local information c-means clustering algorithm," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1328–1337, May 2010.
[13] Z. Zhao, L. Cheng, and G. Cheng, "Neighbourhood weighted fuzzy c-means clustering algorithm for image segmentation," *IET Image Process.*, vol. 8, no. 3, pp. 150–161, Mar. 2014.
[14] M. Gong, Y. Liang, J. Shi, W. Ma, and J. Ma, "Fuzzy c-means clustering with local information and kernel metric for image segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 573–584, Feb. 2013.
[15] Y. Zhang, X. Bai, R. Fan, and Z. Wang, "Deviation-sparse fuzzy c-means with neighbor information constraint," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 1, pp. 185–199, Jan. 2019.
[16] G. Liu, Y. Zhang, and A. Wang, "Incorporating adaptive local information into fuzzy clustering for image segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3990–4000, Nov. 2015.
[17] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.
[18] L. Szilágyi, Z. Benyó, S. Szilágyi, and H. Adam, "MR brain image segmentation using an enhanced fuzzy c-means algorithm," in *Proc. 25th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Cancun, Mexico, 2003, pp. 17–21.
[19] W. Cai, S. Chen, and D. Zhang, "Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation," *Pattern Recognit.*, vol. 40, no. 3, pp. 825–838, Mar. 2007.
[20] T. Lei, X. Jia, Y. Zhang, L. He, H. Meng, and A. K. Nandi, "Significantly fast and robust fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 5, pp. 3027–3041, Oct. 2018.
[21] J. Gu, L. Jiao, S. Yang, and F. Liu, "Fuzzy double c-means clustering based on sparse self-representation," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 612–626, Apr. 2018.
[22] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: Fast superpixels using geometric flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2290–2297, Dec. 2009.
[23] T. Lei, X. Jia, Y. Zhang, S. Liu, H. Meng, and A. K. Nandi, "Superpixel-based fast fuzzy c-means clustering for color image segmentation," *IEEE Trans. Fuzzy Syst.*, 2018, doi: 10.1109/TFUZZ.2018.2889018
[24] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. Adv. Neural Inf. Proc. Syst.*, Vancouver, CA, USA, 2005, pp. 1601–1608.
[25] L. Y. Tseng and S. B. Yang, "A genetic approach to the automatic clustering problem," *Pattern Recognit.*, vol. 34, no. 2, pp. 415–424, Feb. 2001.
[26] S. Das, A. Abraham, and A. Konar, "Automatic clustering using an improved differential evolution algorithm," *IEEE Trans. Syst. Man Cybern. A, Syst. Humans*, vol. 38, no. 1, pp. 218–237, Jan. 2008.
[27] M. S. Yang and Y. Nataliani, "Robust-learning fuzzy c-means clustering algorithm with unknown number of clusters," *Pattern Recognit.*, vol. 71, pp. 45–59, Nov. 2017.
[28] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496. Jun. 2014.
[29] G. Wang and Q. Song, "Automatic clustering via outward statistical testing on density metrics," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 8, pp. 1971–1985, Aug. 2016.
[30] Y. Zhu, K. M. Ting, and M. J. Carman, "Density-ratio based clustering for discovering clusters with varying densities," *Pattern Recognit.*, vol. 60, pp. 983–997, Dec. 2016.
[31] R. J. Kuo and F. E. Zulvia, "Automatic clustering using an improved artificial bee colony optimization for customer segmentation," *Knowl. Inf. Syst.*, vol. 57, no. 2, pp. 331–357, Nov. 2018.
[32] J. Hou, H. Gao, and X. Li, "DSets-DBSCAN: A parameter-free clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3182–3193, Jul. 2016.
[33] T. Lei, X. Jia, T. Liu, S. Liu, H. Meng, and A. K. Nandi, "Adaptive morphological reconstruction for seeded image segmentation," *IEEE Trans. Image Process.*, 2019, doi: 10.1109/TIP.2019.2920514.
[34] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vision.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.
[35] H. Lombaert, Y. Sun, L. Grady, and C. Xu, "A multilevel banded graph cuts method for fast image segmentation," in *Proc. IEEE Int. Conf. Comput. Vision*, Beijing, China, 2005, pp. 259–265.
[36] A. P. Moore, S. J. D. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2008, pp. 1–8.
[37] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1451–1462, Apr. 2014.
[38] F. Forbes and N. Peyrard, "Hidden Markov random field model selection criteria based on mean field-like approximations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1089–1101, Sep. 2003.
[39] A. Quattoni, S. Wang, L. P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1848–1852, Oct. 2007.
[40] S. P. Chatzis and T. A. Varvarigou, "A fuzzy clustering approach toward hidden Markov random field models for enhanced spatially constrained image segmentation," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 5, pp. 1351–1361, Oct. 2008.
[41] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
[42] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5933–5942, Dec. 2016.
[43] Z. Li and J. Chen, "Superpixel segmentation using linear spectral clustering," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Boston, MA, USA, 2015, pp. 1356–1363.
[44] Z. Ban, J. Liu, and L. Cao, "Superpixel segmentation using gaussian mixture model," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4105–4117, Aug. 2018.
[45] X. Wei, Q. Yang, Y. Gong, and N. Ahuja, "Superpixel hierarchy," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4838–4849, Oct. 2018.
[46] M. Gong, L. Su, M. Jia, and W. Chen, "Fuzzy clustering with a modified MRF energy function for change detection in synthetic aperture radar images," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 1, pp. 98–109, Feb. 2014.
[47] J. Hu and Y. Wen, "Adaptive fuzzy clustering algorithm with local information and Markov random field for image segmentation," in *Proc. Int. Conf. Neural Inf. Process.* Springer, 2018, pp. 170–180.
[48] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
[49] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2–3, pp. 191–203, May 1984.
[50] F. Guo, X. Wang, and J. Shen, "Adaptive fuzzy c-means algorithm based on local noise detecting for image segmentation," *IET Image Process.*, vol. 10, no. 4, pp. 272–279, Apr. 2016.
[51] X. Wang, Y. Tang, S. Masnou, and L. Chen, "A global/local affinity graph for image segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1399–1411, Apr. 2015.
[52] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558–1570, Aug. 2015.

Tao Lei (M'17) received the Ph.D. degree in information and communication engineering from the Northwestern Polytechnical University, Xi'an, China, in 2011.

From 2012 to 2014, he was a Postdoctoral Research Fellow with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China. From 2015 to 2016, he was a Visiting Scholar with the Quantum Computation and Intelligent Systems Group, University of Technology Sydney, Sydney, Australia. He has authored and co-authored 80+ research papers including IEEE TIP, TFS, TGRS, TGRSL, ICASSP, ICIP, and FG. He is currently a Professor with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China. His current research interests include image processing, pattern recognition, and machine learning.

Peng Liu received the B.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, and the M.S. degree from the Chengdu University of Information and Technology, Chengdu, China. He is currently working toward the Ph.D. degree with the School of Electrical and Control Engineering, Shaanxi University of Science and Technology, Xi'an, China.

His current research interests include computer vision and pattern recognition.

Xiaohong Jia received the M.S. degree in signal and information processing from the Lanzhou Jiaotong University, Lanzhou, China, in 2017. He is currently working toward the Ph.D. degree with the School of Electrical and Control Engineering, Shaanxi University of Science and Technology, Xi'an, China. His current research interests include image processing and pattern recognition.

Xuande Zhang received the B.S. degree from the Ningxia University, Yinchuan, China, in 2000, and the M.S. and Ph.D. degrees from the Xidian University, Xi'an, China, in 2006 and 2013, respectively.

From 2009 to 2010, he was a Visiting Scholar with the Department of Computing, The Hong Kong Polytechnic University. He is currently a professor with the School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Hung Hom, Hong Kong. His research interests include image processing and pattern recognition.

Hongying Meng (M'10–SM'17) received the Ph.D. degree in communication and electronic systems from Xi'an Jiaotong University, Xi'an, China, in 1998.

He is currently a Senior Lecturer with the Department of Electronic and Computer Engineering, Brunel University London, London, U.K. His research interests include digital signal processing, affective computing, machine learning, human–computer interaction, computer vision, and embedded systems with over 90 publications in these areas.

Dr. Meng is a member of the Institute of Environment, Health and Societies, Human Centred Design Institute (HCDI), and Wireless Networks and Communications Research Center at Brunel. He is a Fellow of the Higher Education Academy (HEA) and a member of Engineering Professors Council in U.K. His audio-based and video-based emotion recognition systems have won the International Audio/Visual Emotion Challenges AVEC2011 and AVEC2013 Prizes, respectively.

Asoke K. Nandi (F'11) received the Ph.D. degree in physics from the University of Cambridge (Trinity College), Cambridge, U.K.

He held academic positions in several universities, including Oxford University, Oxford, U.K., Imperial College London, London, U.K., University of Strathclyde, Glasgow, U.K., and University of Liverpool, Liverpool, U.K., as well as Finland Distinguished Professorship in Jyvaskyla, Jyvaskyla, Finland. He was also an IEEE EMBS Distinguished Lecturer (2018–2019). In 2013, he moved to Brunel University London, London, U.K., to become the Chair and Head of Electronic and Computer Engineering. He is a Distinguished Visiting Professor with the Tongji University, Shanghai, China, and an Adjunct Professor with the University of Calgary, Calgary, AB, Canada. He has a great deal of expertise in "Big Data," dealing with heterogeneous data, and extracting information from multiple datasets obtained in different laboratories and different times. He has authored over 590 technical publications, including 240 journal papers and five books. The H-index of his publications is 70 (Google Scholar) and ERDOS number is 2. His current research interests include the areas of signal processing and machine learning, with applications to communications, gene expression data, functional magnetic resonance data, and biomedical data. He has made many fundamental theoretical and algorithmic contributions to many aspects of signal processing and machine learning.

Dr. Nandi is a Fellow of the Royal Academy of Engineering (U.K.) as well as a Fellow of seven other institutions, including the IET. Among the many awards he received are the Institute of Electrical and Electronics Engineers (USA) Heinrich Hertz Award in 2012, the Glory of Bengal Award for his outstanding achievements in scientific research in 2010, the Water Arbitration Prize of the Institution of Mechanical Engineers (U.K.) in 1999, and the Mountbatten Premium, Division Award of the Electronics and Communications Division, of the Institution of Electrical Engineers (U.K.) in 1998. In 1983, he co-discovered the three fundamental particles known as W^+, W^-, and Z^0 (by the UA1 team at CERN), providing the evidence for the unification of the electromagnetic and weak forces, for which the Nobel Committee for Physics in 1984 awarded the prize to his two team leaders for their decisive contributions.