

## Research Article

# Robust FCM Algorithm with Local and Gray Information for Image Segmentation

**Hanane Barrah, Abdeljabbar Cherkaoui, and Driss Sarsri**

*Laboratory of Innovative Technologies, National School of Applied Sciences, Tangier, Morocco*

Correspondence should be addressed to Hanane Barrah; hananbarah@gmail.com

Received 25 July 2016; Revised 8 September 2016; Accepted 21 September 2016

Academic Editor: Gözde Ulutagay

Copyright © 2016 Hanane Barrah et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The FCM (fuzzy c-mean) algorithm has been extended and modified in many ways in order to solve the image segmentation problem. However, almost all the extensions require the adjustment of at least one parameter that depends on the image itself. To overcome this problem and provide a robust fuzzy clustering algorithm that is fully free of the empirical parameters and noise type-independent, we propose a new factor that includes the local spatial and the gray level information. Actually, this work provides three extensions of the FCM algorithm that proved their efficiency on synthetic and real images.

## 1. Introduction

Clustering unlabeled data into the most homogeneous groups is a problem that has received extensive attention in many application domains [1–3]. Thus, several clustering methods have been developed. The hard (or crisp), probabilistic, and possibilistic c-means [4] are the well-known partitioning methods that have been extended to many different versions based on the data type and the application purpose. The probabilistic or fuzzy c-means (FCM) is always used to generate fuzzy partitions and, thus, it is widely useful to segment images [2, 5] where the fuzzy data is redundant. In fact, Abdel-Maksoud et al. [6] used the fuzzy c-means algorithm combined with its hard version k-means to extract brain tumors from MRI (Magnetic Resonance Imaging) images. In order to detect targets from radar images, Gupta [3] extended the fuzzy c-means to the fuzzy Gustafson-Kessel algorithm that uses the Mahalanobis distance instead of the Euclidean one. In addition to target detection, the proposed fuzzy Gustafson-Kessel algorithm proved its ability to clutter rejection.

Even though the standard FCM algorithm has demonstrated its accuracy in segmenting different kinds of images, it is still inefficient in the presence of noise, where its performance gradually decreases as the image noise increases. This problem is due to the lack of spatial information. To

enhance the robustness and the efficiency of the standard FCM algorithm and make it strong enough in the presence of noise, lots of researchers have modified it in different ways; some have modified the objective function, while the others have used different distance metrics. In fact, Pham [7] proposed a Robust Fuzzy C-Means (RFCM) algorithm based on a generalized objective function that includes a spatial penalty on the membership function. Despite its strength in handling noisy pixels, the RFCM algorithm still suffers from many problems. First of all, the penalty term has to be computed in each iteration, which increases the computational burden. Second, the algorithm depends on a crucial parameter  $\beta$  that requires being selected properly in order to achieve the optimal result. Third, the spatial constraint causes a smoothing effect which can remove some fine details.

To deal with the intensity inhomogeneity in MRI images, Ahmed et al. [8] also modified the objective function of the standard FCM by including a neighborhood term that biases the labeling of a pixel by the labels of its immediate neighboring pixels. The proposed algorithm (always referred to by FCM.S) outperformed the FCM and demonstrated its usefulness in coping with “Salt and Pepper” noise. However, the FCM.S suffers from the same problems as the RFCM algorithm. In fact, the clustering accuracy depends on the selection of the parameter  $\alpha$  that controls the tradeoff

between noise elimination and detail preservation; the spatial information causes the blurring of some fine details and computing the neighborhood term in each iteration requires the algorithm to be highly consumer in the running times point of view. To overcome this latter drawback, the FCM.S has been extended to three algorithms: The EnFCM (Enhanced FCM), FCM.S1, and FCM.S2. The first extension EnFCM was proposed by Szilágyi et al. [9] to reduce the required calculations by introducing a new factor  $\gamma \in [0.5 \ 1.2]$ . This algorithm consists first of computing a linearly weighted sum image and then clustering it based on the gray level histogram rather than the image pixels. The segmentation quality of this algorithm is comparable to FCM.S, although the EnFCM performs quicker than its ancestors. With the same aim of making the FCM.S fast enough, Chen and Zhang [10] proposed the FCM.S1 and FCM.S2 that calculate the neighborhood term based on the mean filtered and median filtered images, respectively. As the filtered image has to be computed once and before the clustering process, the computations needed to compute the neighborhood term are drastically reduced. In fact, the authors demonstrated the effectiveness of their algorithms in artificial and real-world datasets. In [11], the authors have improved the speed of the FCM.S1 and FCM.S2 by introducing a new parameter that balances between the fastness of the hard clustering and the good quality of the fuzzy clustering. Even though the proposed algorithms have proved their fastness over the FCM.S1 and FCM.S2, they are more parameter-dependent.

By combining the main ideas of FCM.S1, FCM.S2, and EnFCM and incorporating the local spatial and the gray information together, Cai et al. [12] came up with a set of Fast Generalized Fuzzy C-Means (FGFCM) clustering algorithms. The authors proved the superiority of the FGFCM over all the aforementioned algorithms, where it overcomes the majority of their drawbacks such as controlling the tradeoff between noise-immunity and detail preserving and removing the empirically adjusted parameter  $\alpha$ , although it requires the adjustment of a new parameter  $\lambda_g$  to achieve better result.

In the same context of improving the standard FCM by including the spatial information, Chuang et al. [13] proposed a fuzzy c-means algorithm that integrates the spatial information in a different way. Indeed, the authors introduced a new spatial function that is used to force the membership value of each pixel to be influenced by the membership values of its immediate neighborhood. Despite its robustness to noise and its ability to reduce the spurious blobs, this algorithm (noted by  $sFCM_{p,q}$ ) still suffers from a major drawback where achieving the optimal segmentation requires the adjustment of two parameters  $p$  and  $q$ .

To improve the robustness to noise of the FCM.S and  $sFCM_{p,q}$ , Zheng et al. [14] combined their main ideas. Thus, the authors used a modified version of the spatial function proposed for  $sFCM_{p,q}$  to minimize an objective function that is slightly different from the FCM.S's. The resulting algorithm surpasses all the aforementioned algorithms, but it is more parameter-dependent.

In order to deal with noise in MRI images, Ji et al. [15] proposed a Robust Spatially Constrained Fuzzy C-Means (RSCFCM) algorithm that is based on a spatial factor that

works as a linear filter for smoothing and restoring noisy images. The RSCFCM algorithm minimizes a fuzzy objective function that integrates the bias field estimation, which makes it effective for intensity inhomogeneity. By testing this algorithm on synthetic and clinical images, the authors realized its better segmentation accuracy over several state-of-the-art algorithms. Nevertheless, the RSCFCM algorithm requires the adjustment of a parameter  $\beta$ .

So far, all the aforementioned extensions of the standard FCM have succeeded to different extents in dealing with noise. However, they all share the major drawback of adjusting empirical parameters ( $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\lambda_g$ ,  $p$ ,  $q$ , and  $h$ ). In case of the FCM.S and its two variants FCM.S1 and FCM.S2, the parameter  $\alpha$  controls the tradeoff between noise elimination and detail preservation. In fact,  $\alpha$  has to be chosen large enough to remove noise and small enough to preserve fine details. Thus, the selection of this parameter is strongly dependent on the type and the level of noise. As the type and the level of noise are always a priori unknown, choosing the proper value of  $\alpha$  remains a very difficult task, where it is always determined using trial-and-error experiments. To overcome this latter problem, this work proposes replacing the parameter  $\alpha$  with a new factor  $S$  that includes the local spatial and the gray level information. Actually, we propose three Robust FCM algorithms: RFCMLGI (Robust FCM with Local and Gray Information), RFCMLGI.1, and RFCMLGI.2, which are direct extensions of the FCM.S, FCM.S1, and FCM.S2, respectively [10]. The proposed algorithms use the local spatial and the gray level information together to calculate the weight of the neighborhood term; the main idea here is to amplify this weight for noisy pixels and minimize it for nonnoisy ones.

In addition to the inherited advantages from FCM.S, FCM.S1, and FCM.S2, the proposed algorithms come up with valuable ones. At first, they are all fully free of the empirical parameters. Second, they control the tradeoff between noise elimination and detail preservation automatically. Third, the RFCMLGI algorithm is noise type-independent. Finally, all the algorithms are easy to be implemented, because the new factor  $S$  is proposed in a way to be easily and rapidly computed.

## 2. Material and Methods

The fuzzy clustering is always defined as the process of grouping, with uncertainty, unlabeled data into the most homogeneous groups or clusters as much as possible [16–18], such that the data within the same cluster are the most similar, and data from different clusters are the most dissimilar. It is an unsupervised classification, because it does not have any previous knowledge about the data structure.

**2.1. Standard Fuzzy C-Means Algorithm: FCM.** The fuzzy c-means or the FCM is the well-known and the best used fuzzy clustering algorithm that is based on the fuzzy sets theory [19] to create homogeneous clusters. This algorithm considers the clustering as an optimization problem where an objective function must be minimized. It receives through its input the

dataset  $I = \{x_j \in \mathbb{R}^d\}_{j=1,\dots,N}$  (part of a  $d$ -dimensional space) and the number of clusters  $C$  in order to minimize iteratively the following objective function:

$$J(I, U, V) = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|x_j - v_i\|^2. \quad (1)$$

$\|\cdot\|$  is the Euclidean distance,  $N$  is the number of elements in  $I$ , and  $m$  is the fuzziness exponent.  $V = [v_i]$  is the set of the cluster centers.  $U = [u_{ij}]$  is the fuzzy partition matrix that satisfies the following condition:

$$E = \left\{ u_{ij} \in [0, 1] \mid \sum_{i=1}^C u_{ij} = 1, \forall j, 0 < \sum_{j=1}^N u_{ij} < N, \forall i \right\}. \quad (2)$$

The minimization of the objective function presented in (1) is carried out by updating iteratively the fuzzy partition matrix and the cluster centers as follows:

$$u_{ij} = \frac{1}{\sum_{k=1}^C (\|x_j - v_i\| / \|x_j - v_k\|)^{2/(m-1)}}, \quad (3)$$

$$v_i = \frac{\sum_{j=1}^N u_{ij}^m \cdot x_j}{\sum_{j=1}^N u_{ij}^m}. \quad (4)$$

#### Algorithm Steps

*Step 0.* Fix the clustering parameters (the converging error  $\varepsilon$ , the fuzziness exponent  $m$ , and the number of clusters  $C$ ), input the dataset  $I$ , and initialize randomly the cluster centers.

**REPEAT**

*Step 1.* Update the partition matrix using (3).

*Step 2.* Update the clusters centers using (4).

**UNTIL.**  $\|V_{\text{new}} - V_{\text{old}}\| < \varepsilon$ .

$V_{\text{new}}$  is the set of the cluster centers found in the current iteration, and  $V_{\text{old}}$  represents the previous one.

**2.2. FCM with Spatial Information and Its Variants: FCM.S, FCM.S1, and FCM.S2.** In order to improve the standard FCM and deal with the intensity inhomogeneities in MRI images, Ahmed et al. [8] modified the objective function (1) by introducing a neighborhood term that biases the labeling of a pixel by the labels of its immediate neighboring pixels. Thus, the authors proposed the FCM.S algorithm that minimizes the following objective function (5) using the updating functions (6) and (7) and with respect to condition E:

$$J(I, U, V) = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \|x_j - v_i\|^2 + \frac{\alpha}{N_R} \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \sum_{r \in N_j} \|x_r - v_i\|^2, \quad (5)$$

$u_{ij}$

$$= \frac{(\|x_j - v_i\|^2 + (\alpha/N_R) \sum_{r \in N_j} \|x_r - v_i\|^2)^{-1/(m-1)}}{\sum_{k=1}^C (\|x_j - v_k\|^2 + (\alpha/N_R) \sum_{r \in N_j} \|x_r - v_k\|^2)^{-1/(m-1)}}, \quad (6)$$

$$v_i = \frac{\sum_{j=1}^N u_{ij}^m (x_j + (\alpha/N_R) \sum_{r \in N_j} x_r)}{(1 + \alpha) \sum_{j=1}^N u_{ij}^m}. \quad (7)$$

$N_j$  stands for the set of neighbors that exist in a window around  $x_j$  and  $N_R$  is its cardinality. The parameter  $\alpha$  controls the effect of the neighboring term.

It is noteworthy that the neighborhood information appears in both updating functions (6) and (7), which means that the neighboring term has to be computed in each iteration; thus the FCM.S algorithm becomes very time-consuming. To get over this drawback, Chen and Zhang [10] simplified the objective function (5) to the following one:

$$J(D, U, V) = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|x_j - v_i\|^2 + \alpha \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|\bar{x}_j - v_i\|^2. \quad (8)$$

$\bar{x}_j$  could be the mean or the median value of the neighbors within a specified window around  $x_j$ . Actually, the authors came up with two fuzzy clustering algorithms: the FCM.S1 and FCM.S2 that use the mean filtered and median filtered images, respectively.

Like the standard FCM and FCM.S algorithms, the FCM.S1 and FCM.S2 algorithms minimize iteratively the objective function (8) by updating the fuzzy partition matrix and the cluster centers as follows:

$$u_{ij} = \frac{(\|x_j - v_i\|^2 + \alpha \|\bar{x}_j - v_i\|^2)^{-1/(m-1)}}{\sum_{k=1}^C (\|x_j - v_k\|^2 + \alpha \|\bar{x}_j - v_k\|^2)^{-1/(m-1)}}, \quad (9)$$

$$v_i = \frac{\sum_{j=1}^N u_{ij}^m (x_j + \alpha \bar{x}_j)}{(1 + \alpha) \sum_{j=1}^N u_{ij}^m}. \quad (10)$$

#### Algorithm Steps

*Step 0.* Fix the clustering parameters (the converging error  $\varepsilon$ , the fuzziness exponent  $m$ , the number of clusters  $C$ , and the new parameter  $\alpha$ ), input the dataset  $I$ , and initialize the clusters centers randomly.

*Step 1.* Compute the mean (median, resp.) filtered image in case of the FCM.S1 (FCM.S2, resp.).

**REPEAT**

*Step 2.* Update the partition matrix using (9).

*Step 3.* Update the clusters centers using (10).

**UNTIL.**  $\|V_{\text{new}} - V_{\text{old}}\| < \varepsilon$ .

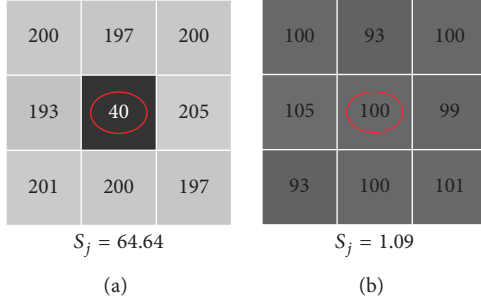


FIGURE 1: 2D square window. (a) The central pixel is noisy. (b) The central pixel is not noisy.

**2.3. Robust FCM with Local and Gray Information: RFCMLGI.** Even though the FCM\_S, FCM\_S1, and FCM\_S2 have shown their strength in handling noise, adjusting the parameter  $\alpha$  is still their major limitation. It is highly important to note that this parameter  $\alpha$  has to be chosen large enough to eliminate noisy pixels and small enough to preserve more fine details. In other words, if the pixel under consideration is noisy, the weight of the neighboring term has to be large enough to bias the pixel's belongingness by its immediate neighborhood; if it is not, this weight has to be small enough in order not to alter significantly the pixel's belongingness and preserve it as fine detail. To respect this important note and control automatically the effect of the neighboring term, this work proposes a Robust FCM with Local and Gray Information (RFCMLGI) that is a direct extension of the FCM\_S and replaces  $\alpha$  with the new factor  $S$  defined as follows:

$$S_j = \frac{1}{N_R} \sum_{r \in N_j} \frac{\|x_j - x_r\|}{d_{rj} + 1}. \quad (11)$$

$N_R$  and  $N_j$  are defined as in the FCM\_S, and  $d_{rj}$  represents the spatial Euclidean distance between the pixels  $x_j$  and  $x_r$ .

The new factor  $S$  is calculated using the local spatial information (the spatial Euclidean distances  $d_{rj}$ ) and the gray level information (the gray levels of the neighboring pixels  $x_r$ ). It is defined in a way to be amplified for noisy pixels and minimized for nonnoisy ones. In fact, it is obviously deducible that  $S_j$  tends towards a maximum if  $x_j$  is noisy and its neighborhood is homogeneous, which increases the effect of the neighborhood term (see example in Figure 1(a)). Similarly, in a homogeneous window, the parameter  $S_j$  tends to a minimum, because the central pixel is not noisy; thus, the neighborhood effect decreases (see Figure 1(b)). Moreover, the contribution degree of each neighboring pixel (for calculating  $S_j$ ) is inversely proportional to its spatial distance from the central pixel, which means that the nearest neighbors to the central pixel contribute more strongly than those more distant.

The RFCMLGI algorithm clusters data by minimizing iteratively the following objective function and under the previous condition  $E$ :

$$J(I, U, V) = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|x_j - v_i\|^2 + \frac{1}{N_R} \sum_{i=1}^C \sum_{j=1}^N S_j \cdot u_{ij}^m \cdot \sum_{r \in N_j} \|x_r - v_i\|^2. \quad (12)$$

This optimization problem will be solved using Lagrange multiplier:

$$J = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|x_j - v_i\|^2 + \frac{1}{N_R} \sum_{i=1}^C \sum_{j=1}^N S_j \cdot u_{ij}^m \cdot \sum_{r \in N_j} \|x_r - v_i\|^2 + \lambda \cdot \left(1 - \sum_{i=1}^C u_{ij}\right). \quad (13)$$

By taking the first derivative of  $J$  with respect to  $u_{ij}$  and setting the result to zero we find

$$\left[ m \cdot u_{ij}^{m-1} \cdot \|x_j - v_i\|^2 + \frac{m \cdot S_j}{N_R} \cdot u_{ij}^{m-1} \cdot \sum_{r \in N_j} \|x_r - v_i\|^2 - \lambda \right]_{u_{ij}=u_{ij}^*} = 0. \quad (14)$$

Solving (14) for  $u_{ij}$ ,

$$u_{ij}^* = \left[ \frac{\lambda}{m \cdot (\|x_j - v_i\|^2 + (S_j/N_R) \cdot \sum_{r \in N_j} \|x_r - v_i\|^2)} \right]^{1/(m-1)}. \quad (15)$$

As  $\sum_{i=1}^C u_{ij} = 1 \quad \forall j \in \{1, \dots, N\}$ , then

$$\sum_{i=1}^C \left[ \frac{\lambda}{m \cdot (\|x_j - v_i\|^2 + (S_j/N_R) \cdot \sum_{r \in N_j} \|x_r - v_i\|^2)} \right]^{1/(m-1)} = 1. \quad (16)$$

Thus,

$$\lambda = \frac{m}{\left[ \sum_{i=1}^C \left( 1/\|x_j - v_i\|^2 + (S_j/N_R) \cdot \sum_{r \in N_j} \|x_r - v_i\|^2 \right)^{1/(m-1)} \right]^{(m-1)}}. \quad (17)$$

Substituting  $\lambda$  into (15), we find

$$u_{ij}^* = \frac{\left( \|x_j - v_i\|^2 + (S_j/N_R) \sum_{r \in N_j} \|x_r - v_i\|^2 \right)^{-1/(m-1)}}{\sum_{k=1}^C \left( \|x_j - v_k\|^2 + (S_j/N_R) \sum_{r \in N_j} \|x_r - v_k\|^2 \right)^{-1/(m-1)}}. \quad (18)$$

This time, we take the first derivative of  $J$  with respect to  $v_i$  and setting the result to zero:

$$\left[ \sum_{j=1}^N u_{ij}^m \cdot (x_j - v_i) + \frac{1}{N_R} \cdot \sum_{j=1}^N S_j \cdot u_{ij}^m \cdot \sum_{r \in N_j} (x_r - v_i) \right]_{v_i=v_i^*} = 0. \quad (19)$$

Solving for  $v_i$ , we find

$$v_i = \frac{\sum_{j=1}^N u_{ij}^m \cdot (x_j + (S_j/N_R) \sum_{r \in N_j} x_r)}{\sum_{j=1}^N (1 + S_j) \cdot u_{ij}^m}. \quad (20)$$

It is noticeable that the factor  $S$  is independent of the cluster centers and the membership values. Thus, its calculation takes place once at the beginning of the clustering process, which does not require much processing time. Besides, the computation of the neighborhood term is required in each iteration, which makes the algorithm much slower than the standard FCM. To overcome this last shortcoming (and by analogy to FCM\_S and its two variants FCM\_S1 and FCM\_S2) the RFCMLGI algorithm has been extended to two simplified versions. Indeed, by simplifying the objective function presented in (12) to the following one (21) we come up with two algorithms RFCMLGI\_1 and RFCMLGI\_2 that update the partition matrix and the cluster centers using (22) and (23), respectively:

$$J(D, U, V)$$

$$= \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \cdot \|x_j - v_i\|^2 + \sum_{i=1}^C \sum_{j=1}^N S_j \cdot u_{ij}^m \cdot \|\bar{x}_j - v_i\|^2, \quad (21)$$

$$u_{ij} = \frac{\left( \|x_j - v_i\|^2 + S_j \|\bar{x}_j - v_i\|^2 \right)^{-1/(m-1)}}{\sum_{k=1}^C \left( \|x_j - v_k\|^2 + S_j \|\bar{x}_j - v_k\|^2 \right)^{-1/(m-1)}}, \quad (22)$$

$$v_i = \frac{\sum_{j=1}^N u_{ij}^m \cdot (x_j + S_j \cdot \bar{x}_j)}{\sum_{j=1}^N (1 + S_j) \cdot u_{ij}^m}. \quad (23)$$

As in FCM\_S1 and FCM\_S2,  $\bar{x}_j$  could be the mean or the median value of the neighbors within a specified window around  $x_j$ .

#### Algorithm Steps

*Step 0.* Fix the clustering parameters (the converging error  $\varepsilon$ , the fuzziness exponent  $m$ , and the number of clusters  $C$ ), input the dataset  $I$ , and initialize randomly the cluster centers.

*Step 1.* Compute the mean (median, resp.) filtered image in case of the RFCMLGI\_1 (RFCMLGI\_2, resp.) algorithm.

*Step 2.* Compute the new factor  $S$  using (11).

*REPEAT*

*Step 3.* In case of RFCMLGI (RFCMLGI\_1 or RFCMLGI\_2, resp.), update the partition matrix using (18) ((22), resp.).

*Step 4.* In case of RFCMLGI (RFCMLGI\_1 or RFCMLGI\_2), update the cluster centers using (20) ((23), resp.).

*UNTIL.*  $\|V_{\text{new}} - V_{\text{old}}\| < \varepsilon$ .

The major advantages of the proposed algorithms are summarized as follows:

- (i) They are fully free of the empirical parameters.
- (ii) Controlling the tradeoff between noise elimination and detail preservation is automatically made.
- (iii) They are easy to be implemented.
- (iv) The first version of RFCMLGI is noise type-independent.

### 3. Results

In this section, we present some experimental results to show the efficiency of the proposed algorithms RFCMLGI, RFCMLGI\_1, and RFCMLGI\_2 compared to four other fuzzy clustering algorithms: FCM, FCM\_S, FCM\_S1, and FCM\_S2. Thus, several experiments were performed on synthetic and real images and under different types and levels of noise. The clustering parameters were fixed as follows:  $m = 2$ , based on a study presented in [20], and  $N_R = 8$  ( $3 \times 3$  window centered around each pixel except the central pixel).

To evaluate quantitatively the segmentation results, we use the segmentation accuracy (SA) defined as follows:

$$SA = \frac{\text{Number of correctly classified pixels}}{\text{Total number of pixels}}. \quad (24)$$

*3.1. Synthetic Image.* First, we apply all the algorithms to a synthetic image corrupted by different levels of Gaussian and "Salt and Pepper" noise, respectively. This image is composed of  $250 \times 250$  pixels spanning into three classes with three gray level values taken as 0, 100, and 200; thus,  $C$  is fixed at 3. In these experiments, we fixed  $\alpha$  to 4. The segmentation accuracies and results of all the algorithms are depicted in Figures 2 and 3 and Table 1, respectively.



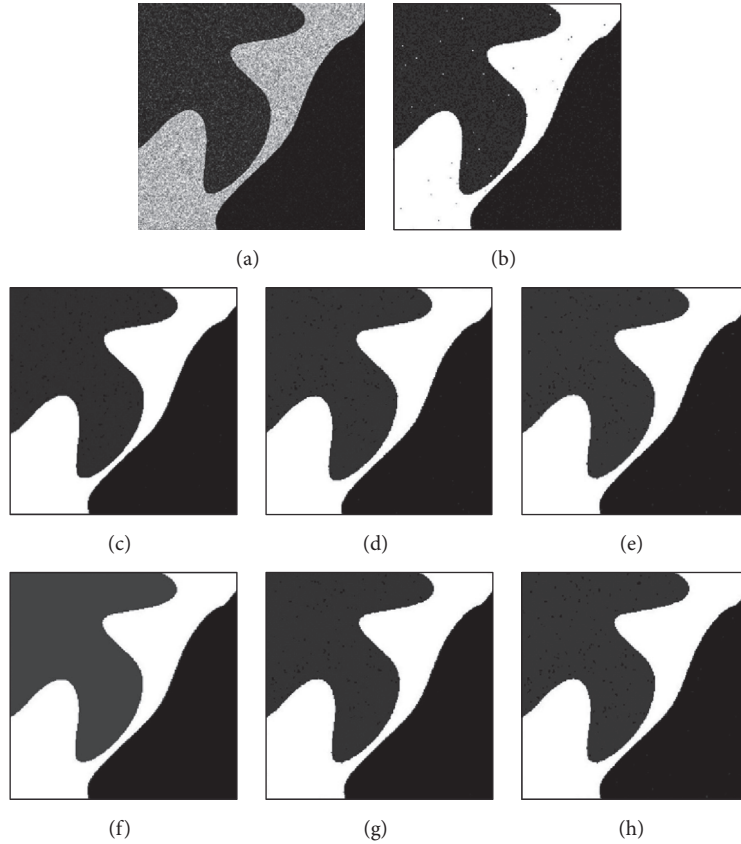


FIGURE 2: Segmentation results on synthetic image. (a) Original image corrupted by 15% of Gaussian noise. (b) FCM result. (c) FCM\_S result. (d) FCM\_S1 result. (e) FCM\_S2 result. (f) RFCMLGI result. (g) RFCMLGI.1 result. (h) RFCMLGI.2 result.

From the visual results presented in Figures 2 and 3, it is clearly noticeable that all the algorithms (except the standard FCM) succeeded to different extent in handling noisy pixels. Moreover, the RFCMLGI performed better in both cases as well as the RFCMLGI.2 under Salt and Pepper noise, which is quantitatively demonstrated in Table 1.

From the numerical results depicted in Table 1, we could point out the following important notes:

- (i) The segmentation accuracy decreases as the level of noise increases for all the algorithms except for the RFCMLGI under both types of noise and RFCMLGI.2 under Salt and Pepper noise.
- (ii) For each type and level of noise, the proposed algorithms RFCMLGI, RFCMLGI.1, and RFCMLGI.2 outperformed the FCM\_S, FCM\_S1, and FCM\_S2, respectively. And, more specifically, the segmentation accuracies produced by the RFCMLGI are more or less similar, which means that this algorithm is less dependent on the noise type.
- (iii) Under Salt and Pepper noise, the segmentation accuracies performed by the RFCMLGI.2 are equal and tend towards the maximum, which proves the convenience of this algorithm to segment images corrupted by Salt and Pepper noise.

- (iv) Under Gaussian noise, RFCMLGI has the best performance.

Based on the previous remarks, we conclude that the proposed algorithms surpassed the FCM\_S and its two variants. In addition, if the type of noise is unknown the RFCMLGI is the best choice.

**3.2. Real Images.** To validate our methods, we test them on two real images and compare their results with the best results of the FCM\_S, FCM\_S1, and FCM\_S2 that are obtained by seeking the value of  $\alpha$  after which there are no apparent changes in the segmentation accuracy.

**3.2.1. Selection of  $\alpha$ .** We use the trial-and-error method to select the best values of  $\alpha$  to segment the previous synthetic image corrupted by 30% of Gaussian and Salt and Pepper noise, respectively. Actually, under Gaussian (Salt and Pepper, resp.) noise and for  $\alpha \geq 10.2$  ( $\alpha \geq 6.8$ , resp.) there are no apparent changes in the segmentation accuracy of the FCM\_S and FCM\_S1 (FCM\_S2, resp.).

**3.2.2. Eight Image.** This image was corrupted by 30% of Gaussian and Salt and Pepper noise, respectively. Even though this image contains two objects, we fixed  $C$  to 3; the third cluster

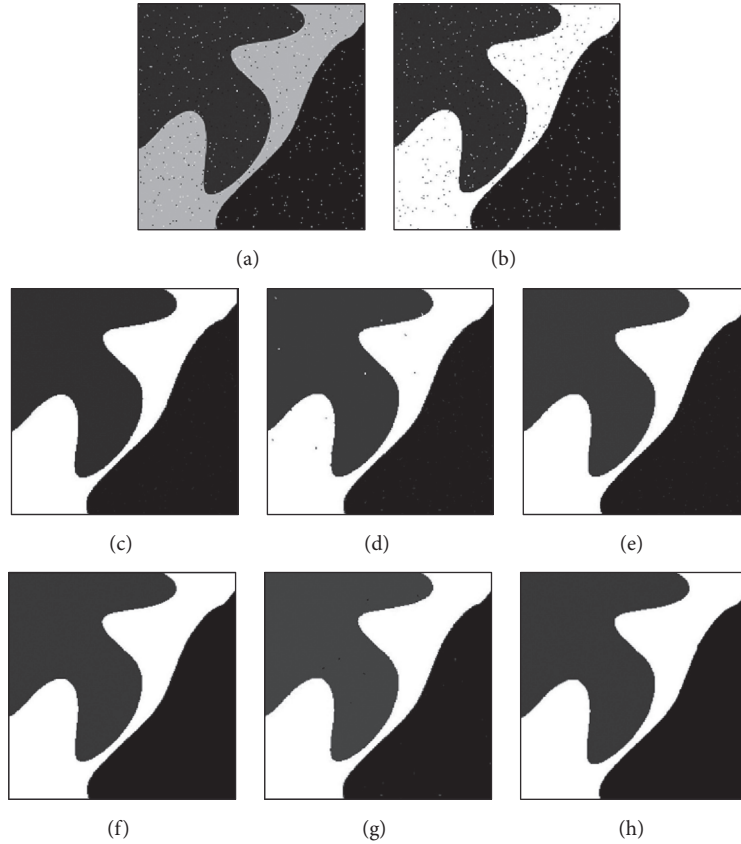


FIGURE 3: Segmentation results on synthetic image. (a) Original image corrupted by 15% of Salt and Pepper noise. (b) FCM result. (c) FCM\_S result. (d) FCM\_S1 result. (e) FCM\_S2 result. (f) RFCMLGI result. (g) RFCMLGI\_1 result. (h) RFCMLGI\_2 result.

TABLE 1: Segmentation accuracies (SA, in %) of seven algorithms on synthetic image.

Algorithm	Gaussian			Salt and Pepper		
	9%	12%	15%	9%	12%	15%
FCM	91.24	88.73	86.55	97.89	97.76	97.50
FCM_S	98.23	97.94	97.58	98.21	98.18	98.10
RFCMLGI	<b>98.89</b>	<b>98.89</b>	<b>98.89</b>	<b>98.90</b>	<b>98.89</b>	<b>98.89</b>
FCM_S1	98.79	98.61	98.29	98.73	98.58	98.55
RFCMLGI_1	<b>98.76</b>	<b>98.66</b>	<b>98.44</b>	<b>98.91</b>	<b>98.86</b>	<b>98.86</b>
FCM_S2	98.93	98.72	98.33	98.89	98.79	98.77
RFCMLGI_2	<b>98.94</b>	<b>98.76</b>	<b>98.40</b>	<b>99.01</b>	<b>99.01</b>	<b>99.01</b>

is for the details presented on the coins. The segmentation results of all the algorithms are presented in Figures 4 and 5.

In Figures 4 and 5, we can see that our algorithms RFCMLGI, RFCMLGI\_1, and RFCMLGI\_2 performed better (to different extents) than the FCM\_S, FCM\_S1, and FCM\_S2, respectively. Indeed, in case of Gaussian noise (Figure 4), we notice that the RFCMLGI has the best performance followed by the RFCMLGI\_1. Thus, these two latter algorithms are the most favorable in the presence of Gaussian noise. The failure of the RFCMLGI\_2 to deal correctly with noise is due to the

median filter used inside the algorithm which is known by its weakness in removing Gaussian noise [21].

Under Salt and Pepper noise (Figure 5), we can order the algorithms according to their performances from the best to the worst as follows: RFCMLGI, RFCMLGI\_2, FCM\_S2, RFCMLGI\_1, FCM\_S1, FCM\_S, and FCM. In addition to their best results, the RFCMLGI and RFCMLGI\_2 performed very similarly, which means that they are both very convenient to handle Salt and Pepper noise. The RFCMLGI\_1 could not achieve the best performance because of using the mean filter that is not recommended for Salt and Pepper noise.

In terms of detail preserving, we notice clearly (from Figures 4 and 5) that the RFCMLGI algorithm surpassed all the algorithms, where it found a good balance between noise elimination and detail preserving.

As has been concluded in the previous section, the RFCMLGI algorithm is the most convenient one when noise is a priori unknown.

**3.2.3. Moon Image.** To show the effect of our algorithms on images with mixed noise, we use the “moon” image corrupted at the same time by 20% of Gaussian and Salt and Pepper noise. In this experiment,  $C$  is fixed to 2. The segmentation results are shown in Figure 6.

From Figure 6, we note that the standard FCM has the worst performance. In contrast, the FCM\_S, FCM\_S1,

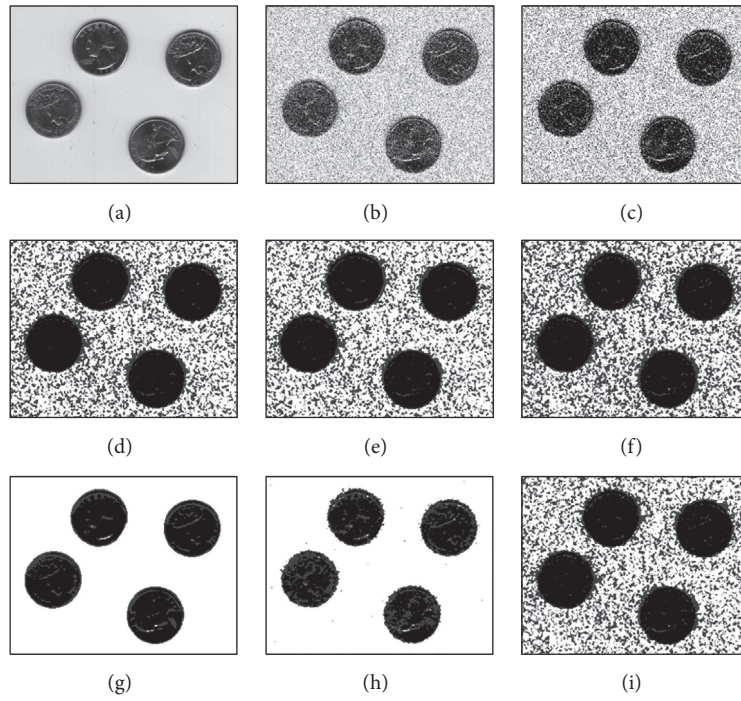


FIGURE 4: Segmentation result on *eight* image. (a) Originaire image. (b) Image corrupted by 30% of Gaussian noise. (c) FCM result. (d) FCM\_S result. (e) FCM\_S1 result. (f) FCM\_S2 result. (g) RFCMLGI result. (h) RFCMLGI\_1 result. (i) RFCMLGI\_2 result.

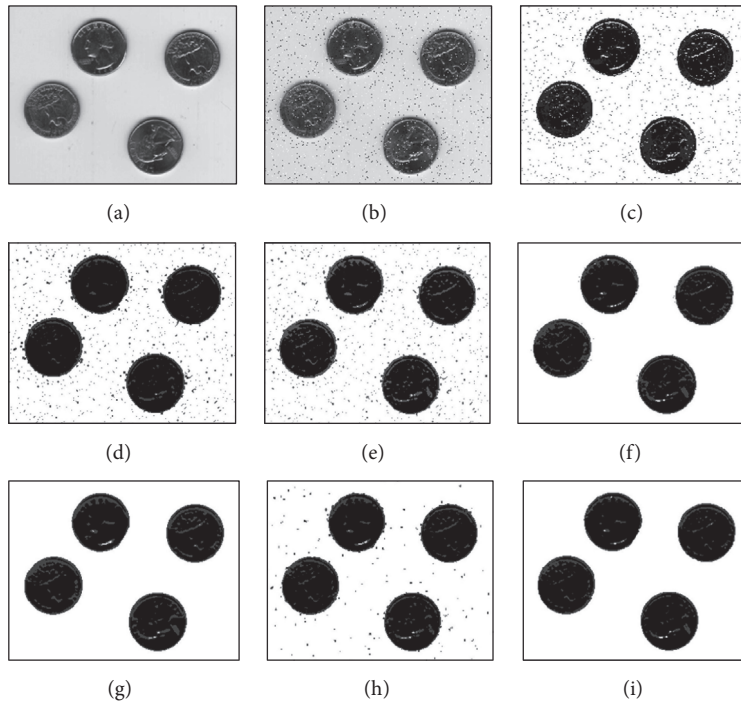


FIGURE 5: Segmentation result on *eight* image. (a) Original image. (b) Image corrupted by 30% of Salt and Pepper noise. (c) FCM result. (d) FCM\_S result. (e) FCM\_S1 result. (f) FCM\_S2 result. (g) RFCMLGI result. (h) RFCMLGI\_1 result. (i) RFCMLGI\_2 result.



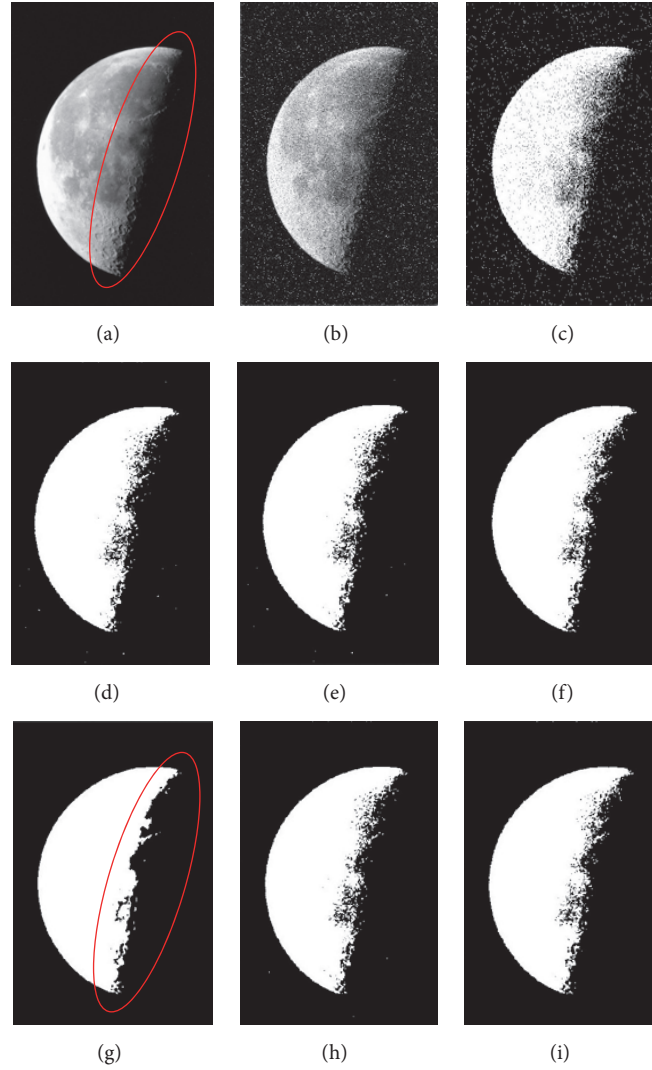


FIGURE 6: Segmentation result on *moon* image. (a) Original image. (b) Image corrupted by Gaussian and “Salt and Pepper” noise. (c) FCM result. (d) FCM.S result. (e) FCM.S1 result. (f) FCM.S2 result. (g) RFCMLGI result. (h) RFCMLGI.1 result. (i) RFCMLGI.2 result.

FCM.S2, RFCMLGI.1, and RFCMLGI.2 succeeded in handling noisy pixels and their performances are close to each other. However, the RFCMLGI algorithm made an exception where it outperformed all the algorithms in handling fuzzy pixels of the intersection region between the moon and the background (see regions circled in red), which proves its ability to retain fine details.

Globally, in the experimental results presented in this section we found that the proposed algorithms RFCMLGI, RFCMLGI.1, and RFCMLGI.2 performed better than the FCM.S, FCM.S1, and FCM.S2, respectively, and the RFCMLGI had the best performance. Even though in some cases RFCMLGI.1 and RFCMLGI.2 performed closely to the FCM.S1 and FCM.S2, they remain better because they are free of any parameter selection and they control the effect of the neighboring term automatically.

The standard FCM and its extensions FCM.S, FCM.S1, and FCM.S2 have the same time complexity which is

$O(HWC)$  [22], where  $H$  and  $W$  are the image dimensions and  $C$  is the number of clusters. The proposed algorithms RFCMLGI, RFCMLGI.1, and RFCMLGI.2 are similar to their antecedents FCM.S, FCM.S1, and FCM.S2, respectively, where the difference lies in the parameter that controls the tradeoff between noise elimination and detail preserving. Thus, the proposed algorithms have also the same time complexity  $O(HWC)$  with small variations caused by computing the new factor  $S$ .

#### 4. Conclusion

In order to furnish a fuzzy clustering algorithm that is fully free of empirical parameters and noise type-independent, this work extended the FCM.S and its two variants to three algorithms based on a new factor that uses the local spatial and the gray level information to calculate the weight of the neighboring term. Generally, all the proposed algorithms

RFCMLGI, RFCMLGI<sub>1</sub>, and RFCMLGI<sub>2</sub> proved their efficiency on synthetic and real images. More specifically, the RFCMLGI algorithm surpassed considerably the others where it showed its noise type-independence and its ability to retain fine details.

In spite of their fruitful results, the proposed algorithms need to be improved in the running times point of view, where computing the factor  $S$  makes them slower. This drawback will be the main issue of a future work.

## Competing Interests

The authors declared that there is no conflict of interests regarding the publication of this paper.

## References

- [1] S. Fayeche, N. Essoussi, and M. Limam, "Partitioning clustering algorithms for protein sequence data sets," *BioData Mining*, vol. 2, no. 1, article 3, 2009.
- [2] D. Aneja and T. K. Rawat, "Fuzzy clustering algorithms for effective medical image segmentation," *International Journal of Intelligent Systems and Applications*, vol. 5, no. 11, pp. 55–61, 2013.
- [3] M. Gupta, "Target detection by fuzzy gustafson-kessel algorithm," *International Journal of Image Processing (IJIP)*, vol. 7, no. 2, p. 203, 2013.
- [4] J. C. Bezdek, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Springer, New York, NY, USA, 2005.
- [5] M. S. Choudhry and R. Kapoor, "Performance analysis of fuzzy C-means clustering methods for MRI image segmentation," *Procedia Computer Science*, vol. 89, pp. 749–758, 2016.
- [6] E. Abdel-Maksoud, M. Elmogy, and R. Al-Awadi, "Brain tumor segmentation based on a hybrid clustering technique," *Egyptian Informatics Journal*, vol. 16, no. 1, pp. 71–81, 2015.
- [7] D. L. Pham, "Spatial models for fuzzy clustering," *Computer Vision and Image Understanding*, vol. 84, no. 2, pp. 285–297, 2002.
- [8] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag, and T. Moriarty, "A modified fuzzy C-means algorithm for bias field estimation and segmentation of MRI data," *IEEE Transactions on Medical Imaging*, vol. 21, no. 3, pp. 193–199, 2002.
- [9] L. Szilágyi, Z. Benyó, S. M. Szilágyi, and H. S. Adam, "MR brain image segmentation using an enhanced fuzzy c-means algorithm," in *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 1, pp. 724–726, September 2003.
- [10] S. Chen and D. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 4, pp. 1907–1916, 2004.
- [11] A. Cherkaoui and H. Barrah, "Fast robust fuzzy clustering algorithm for grayscale image segmentation," in *Proceedings of the Xème Conférence Internationale: Conception et Production Intégrées (CPI '15)*, Tangier, Morocco, December 2015.
- [12] W. Cai, S. Chen, and D. Zhang, "Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation," *Pattern Recognition*, vol. 40, no. 3, pp. 825–838, 2007.
- [13] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T.-J. Chen, "Fuzzy c-means clustering with spatial information for image segmentation," *Computerized Medical Imaging and Graphics*, vol. 30, no. 1, pp. 9–15, 2006.
- [14] F. Zheng, C. Zhang, X. Zhang, and Y. Liu, "A fast anti-noise fuzzy C-means algorithm for image segmentation," in *Proceedings of the 20th IEEE International Conference on Image Processing (ICIP '13)*, pp. 2728–2732, Melbourne, Australia, September 2013.
- [15] Z. Ji, J. Liu, G. Cao, Q. Sun, and Q. Chen, "Robust spatially constrained fuzzy c-means algorithm for brain MR image segmentation," *Pattern Recognition*, vol. 47, no. 7, pp. 2454–2466, 2014.
- [16] S. Ghosh and S. K. Dubey, "Comparative analysis of K-means and fuzzy C-means algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 4, pp. 35–38, 2013.
- [17] D. Lam and D. C. Wunsch, "Chapter 20-Clustering," in *Academic Press Library in Signal Processing*, P. S. R. Diniz, J. A. K. Suykens, R. Chellappa, and S. Theodoridis, Eds., vol. 1, pp. 1115–1149, 2014.
- [18] L. Rokach and O. Maimon, "Clustering methods," in *Data Mining and Knowledge Discovery Handbook*, pp. 321–352, Springer, Berlin, Germany, 2005.
- [19] D. Dubois, W. Ostasiewicz, and H. Prade, "Fuzzy sets: history and basic notions," in *Fundamental of Fuzzy Sets*, Kluwer Academic, Boston, Mass, USA, 2000.
- [20] I. Ozkan and I. B. Turksen, "Upper and lower values for the level of fuzziness in FCM," *Information Sciences*, vol. 177, no. 23, pp. 5143–5152, 2007.
- [21] A. B. Hamza and H. Krim, "Image denoising: a nonlinear robust statistical approach," *IEEE Transactions on Signal Processing*, vol. 49, no. 12, pp. 3045–3054, 2001.
- [22] S. Krinidis and V. Chatzis, "A robust fuzzy local information C-means clustering algorithm," *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp. 1328–1337, 2010.



Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

