Applied Soft Computing Journal 87 (2020) 105928

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Fuzzy C-Means clustering through SSIM and patch for image segmentation

Yiming Tang [a,b], Fuji Ren [a,*], Witold Pedrycz [b,c,d]

[a] Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machine, School of Computer and Information, Hefei University of Technology, Hefei, Anhui, 230601, China
[b] Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, T6R 2V4, Canada
[c] Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[d] Systems Research Institute, Polish Academy of Sciences, Warsaw, 01-447, Poland

ARTICLE INFO

Article history:
Received 7 April 2018
Received in revised form 7 September 2019
Accepted 3 November 2019
Available online 18 November 2019

Keywords:
Image segmentation
Fuzzy C-Means (FCM) clustering
Structural similarity (SSIM)
Image patch

ABSTRACT

In this study, we propose a new robust Fuzzy C-Means (FCM) algorithm for image segmentation called the patch-based fuzzy local similarity c-means (PFLSCM). First of all, the weighted sum distance of image patch is employed to determine the distance of the image pixel and the cluster center, where the comprehensive image features are considered instead of a simple level of brightness (gray value). Second, the structural similarity (SSIM) index takes into account similar degrees of luminance, contrast, and structure of image. The DSSIM (distance for structural similarity) metric is developed on a basis of SSIM in order to characterize the distance between two pixels in the whole image. Next a new similarity measure is proposed. Furthermore, a new fuzzy coefficient is proposed via the new similarity measure together with the weighted sum distance of image patch, and then the PFLSCM algorithm is put forward based on the idea of image patch and this coefficient. Through a collection of experimental studies using synthetic and publicly available images, we demonstrate that the proposed PFLSCM algorithm achieves improved segmentation performance in comparison with the results produced by some related FCM-based algorithms.

1. Introduction

Image segmentation is embodied as a key task in many fields such as computer vision, pattern recognition, affective computing, and multimedia [1–5]. For example, Hernández et al. [6] extended the idea of residue properties, which helped to generate the image quantization table with regard to an arithmetic approach. Kushwaha and Welekar [7] investigated feature selection for content-based image retrieval, in which optimal features were obtained from the feature selection process realized by means of the genetic algorithm. Rezaie and Habiboghli [8] proposed a strategy for the detection of malignant and benign tumors on the CT scan images, where fractal segmentation was used. Image segmentation aims to divide image pixels into several non-overlapping regions, where the pixels in a given region exhibit similar characteristics while pixels positioned in different regions are different. Fuzzy sets [9–11], especially Fuzzy C-Means (FCM) clustering algorithms [12,13], have been extensively employed to carry out image segmentation leading to the improved performance of the segmentation process. The "standard" FCM algorithm works well for most noise-free images, however it is sensitive to noise, outliers and other imaging artifacts. The main reasons behind these drawbacks lie in neglecting spatial context information.

Since the introduction of the FCM algorithm, it has attracted growing interest in the area of image segmentation. Tolias and Panas [14] presented a hierarchical fuzzy clustering-based image segmentation algorithm that was able to cope with nonstationarity and high correlations between pixels. Its performance was better than the possibilistic c-means (PCM) algorithm. Pham and Prince [15] introduced a multiplier field to propose a fuzzy segmentation algorithm for images that were subject to multiplicative intensity inhomogeneities. Wang et al. [16] incorporated the information-theoretic framework and adaptive spatial weighting factors into the FCM-type algorithms to enhance its robustness for image segmentation. Zhou et al. [17] presented a modified mode of the FCM algorithm for image segmentation, in which one used a simple way to update the cluster centers and partitioned the pixels by adding a new bias term into the FCM method. Ji et al. [18] proposed a novel fuzzy clustering approach for brain MR image segmentation. It employed the negative log-posterior regarded as the dissimilarity function, and introduced a new

* Corresponding author.
E-mail addresses: tym608@163.com (Y. Tang), ren@is.tokushima-u.ac.jp (F. Ren), wpedrycz@ualberta.ca (W. Pedrycz).

https://doi.org/10.1016/j.asoc.2019.105928

factor with the spatial direction, and finally incorporated the bias field estimation model into the optimized objective function. Adhikari et al. [19] presented a conditional spatial FCM algorithm for MRI image segmentation. It was constructed by the introduction of conditioning effects imposed by an auxiliary variable related to each pixel as well as spatial information into the membership functions. Chatzis and Varvarigou [20] combined the benefits of the hidden Markov random field (HMRF) with FCM, and established the HMRF-FCM algorithm for image segmentation, which utilized the spatial coherency expressing abilities of HMRF to enhance the FCM segmentation effect. Following the HMRF-FCM algorithm, Liu et al. [21] emphasized the treatment of local information, and introduced region-level information to adjust the range and strength of interactive image pixels. This work was mainly aimed at segmentation of natural color images, and synthetic aperture radar images. In some studies, regularization terms were considered to control the effect of the membership functions. Li et al. [22] employed regularization with the entropy for the membership function. In [23], Miyamoto and Umayahara regularized the FCM function with a quadratic term. But, similar to the classical FCM algorithm, they are only related to the image intensity. Hou et al. [24] regarded a moving-average filter as the regularizer, so it adjusted the main function with the window average of neighborhoods. In [25], the FCM algorithm was improved by a regularizing functional via total variation (TV) related to gradient sparsity, and a regularization parameter was utilized to balance clustering and smoothing. This algorithm was found to be effective and robust in testing images affected by noise and missing data.

1.1. Related work

As an important improvement of FCM, the spatial and gray-level information were introduced into the generic FCM algorithm. Pham [26] employed a spatial penalty based on cross-validation for the FCM objective functions. The corresponding iterative algorithm was only slightly different from the FCM algorithm and allowed the estimation of spatially smooth membership functions. Ahmed et al. [27] proposed FCM_S, in which the basic formula of the FCM was adjusted to compensate for the intensity inhomogeneity and to determine the pixel labeling according to its immediate neighborhood. Furthermore, Chen and Zhang [28] put forward the FCM_S1 and FCM_S2 algorithms as simplified versions of FCM_S, which led to acceptable segmentation results. To speed up the image segmentation process, Szilagyi et al. [29] presented the enhanced FCM (EnFCM) algorithm. In this algorithm, a new image was generated from linearly weighted sum of the original image, and then the gray level histogram of the new image was used for further fuzzy clustering. Similar to EnFCM, Cai et al. [30] proposed a fast generalized FCM (FGFCM) clustering algorithm. This algorithm employed a local similarity measure according to local spatial closeness as well as intensity information, which formed a non-linearly weighted sum image and thus also was characterized by high computational speed. Following it, by virtue of simple local similarity measures, the FGFCM_S1 and FGFCM_S2 algorithms were also presented. Zhao et al. [31] presented a FCM algorithm with non-local spatial information obtained from a large image domain to form the spatial constraint term, in which the non-local spatial information of a pixel was achieved by employing the pixels with a similar configuration of the given pixel. Ma et al. [32] proposed an improved FGFCM algorithm with non-local spatial information, where local and non-local similarity measures were employed with an adaptive weight to balance their impact.

In the previous FCM algorithms, hyper parameters were usually required to control the balance of eliminating noise and retain image details. The values of these hyper parameters were selected experimentally through a trial-and-error method. To solve such problem, Krinidis and Chatzis [33] proposed the fuzzy local information c-means (FLICM) algorithm. FLICM is a special method with a sound and convincing idea, which incorporates the local spatial information and gray level information in a coherent manner. For the first time, FLICM puts forward a fuzzy coefficient G_{ki} which is regarded as a fuzzy local (both spatial and gray level) similarity measure in order to guarantee noise robustness as well as retention of details. Moreover, FLICM is free from empirically adjustable parameters whose tuning usually creates a certain challenge. Based on these observations, FLICM is effective and efficient in the sense that it exhibits robustness in case of noisy images. However, it exhibits some disadvantages:

- In the fuzzy clustering algorithm, one important issue is how to characterize the relationship between the image pixel x_i and the cluster center v_k. It is embodied as the key problem to deliver ideal segmentation result. Actually, FLICM only utilizes the distance $d(x_i, v_k)$ to capture this dependency, which is the same as in the FCM algorithm. Strictly speaking, $d(x_i, v_k)$ cannot adequately characterize such relationship, which only involves two values x_i, v_k without considering the overall characteristics of a more comprehensive character.
- Another important factor is how to characterize the relationship between two pixels x_i and x_j. In the FLICM method, one uses only the spatial Euclidean distance d_{ij} between two pixels i and j to reflect this relationship. By looking more carefully at the essence of the problem, we should establish a general similarity measure between i and j vis-à-vis the entire image. Here d_{ij} is not sufficient to grasp the generalized characteristics of the segmented image.

Following the FLICM algorithm, some further improvements were proposed. Li et al. [34] presented the FCM algorithm with edge and local information (FELICM), which reduced the edge degradation by incorporating the weights of pixels within local neighbor windows. Gong et al. [35] put forward an improvement of FLICM algorithm (RFLICM), which employed the local coefficient of variation to replace the spatial distance as a local similarity measure. Then Gong et al. [36] proposed FCM clustering with local information and kernel metric (KWFLICM) algorithm by setting up a tradeoff weight fuzzy coefficient and a kernel metric, in which the fuzzy coefficient was simultaneously determined in the space distance of all neighboring pixels and their gray-level difference. Verma et al. [37] presented an improved intuitionistic FCM (IIFCM), which was concerned with the local spatial information under the intuitionistic fuzzy environment. Ji et al. [38] proposed FCM clustering with weighted image patch (WIPFCM), which considered image patches to replace pixels with a weighting scheme. Table 1 shows the main idea and highlights its merits along with shortcoming of the related fuzzy clustering algorithms such as FCM, FCM_S, FCM_S1, FCM_S2, EnFCM, FGFCM, FGFCM_S1, FGFCM_S2, FLICM, KWFLICM, IIFCM and WIPFCM.

Although there are some algorithms [34–38] enhancing the performance of the FLICM algorithm to some extent. However, they all employ the computing mechanism similar to FLICM, and it should be pointed out that these two disadvantages still exist. To alleviate them, we elaborate on further enhancements for this issue. In this paper, we will investigate the overall characteristics to characterize the relationship between x_i and v_k, and discover the general similarity measure between i and j, and develop a new image segmentation algorithm.

Table 1
Related fuzzy clustering algorithms.

Algorithm	Main idea	Merit	Shortcoming
FCM	Generic FCM	Simple	Use only the gray value of central pixel
FCM_S	Use of neighbor factor to improve FCM	Neighbor factor considered	Only the gray values considered
FCM_S1	Simplified FCM_S; Refers to mean-filtered image	Neighbor factor considered; Better for Gaussian noise	Only the gray values considered
FCM_S2	Simplified FCM_S; Refers to median-filtered image	Neighbor factor considered; Better for salt-and-pepper noise	Only the gray values considered
EnFCM	Use of a linearly-weighted sum image	Neighbor factor considered; High computational speed	Only the gray values considered; Information loss; Ordinary linearly-weighted sum image
FGFCM	Use of a linearly-weighted sum image from a local similarity measure	High computational speed; Combination of spatial and gray level information	Information loss
FGFCM_S1	Simplified FGFCM; Use of a linearly-weighted sum image derived from neighbor average gray value	Combination of spatial and gray level information; Better for Gaussian noise	Information loss; Ordinary local similarity measure
FGFCM_S2	Simplified FGFCM; Use of a linearly-weighted sum image derived from neighbor median gray value	Combination of spatial and gray level information; Better for salt-and-pepper noise	Information loss; Ordinary local similarity measure
FLICM	Use of a fuzzy coefficient as a fuzzy local similarity measure	No parameter; Use of fuzzy coefficient to combine spatial and gray level information	Ordinary fuzzy coefficient for characterizing the relationship of image elements
KWFLICM	Use of a fuzzy coefficient and kernel metric	No parameter; Use of fuzzy coefficient to combine spatial and gray level information	Huge computing cost; Distance depending on the gray values of two points
IIFCM	Under intuitionistic fuzzy environment; Use of a fuzzy coefficient	Use of intuitionistic fuzzy expressions; Use of fuzzy coefficient to combine spatial and gray level information	Large computing cost; Ordinary intuitionistic fuzzy value only related to image gray
WIPFCM	Use of image patch	Use of image patch; Local spatial information	Ordinary mechanism similar to FCM

1.2. Main contributions

In order to solve such key problem, in this study, we put forward a new FCM algorithm, referred to as the patch-based fuzzy local similarity c-means (PFLSCM) algorithm. First, since the image patches incorporate more general information than image pixels, we use image patch to analyze the relationship between the image pixel and the cluster center, and then employ the weighted sum distance of image patch to measure the distance of the image pixel and the cluster center. Second, we propose a new local distance measure derived from the structural similarity (SSIM) index to compute the distance between two image pixels in the overall image, and then put forward a novel similarity measure. The new one conveys not only the spatial relationship of two image pixels but also the relationship related to luminance and contrast as well as structure of two patches revolved around them. Third, the PFLSCM algorithm is designed based on the idea of image patch, the novel similarity measure as well as the corresponding fuzzy coefficient. Lastly, we carry on experiments using synthetic, real-world and medical images with several types of noises, and it is found that the PFLSCM algorithm has better performance than other seven algorithms in terms of evaluation indicators and visualization effects.

2. Proposed method

First of all, we explore how to adequately utilize the characteristics of image pixels. From the general viewpoint, using the image patch can reveal more structure of image than individual pixels. As a result, the basic distance $d(x_i, v_k)$ can be restructured into a weighted sum of image patch:

$$\sum_{r=1}^{p} \omega_r d(x_{ir}, v_{kr}). \tag{1}$$

Here x_{ir} is the value of the point in an image patch (e.g., a window) located around x_i, and p stands for the number of points in the image patch, while v_{kr} is the new cluster centers ($i = 1, \ldots, N, k = 1, \ldots, c, r = 1, \ldots, p$). Here $\sum_{r=1}^{p} \omega_r = 1$, in which ω_r is the weight associated with the distance $d(x_{ir}, v_{kr})$ (in which $\omega_r \geq 0$). Note that the idea of image patch works not only for the image pixel but also the cluster center.

It is intuitive to assume that ω_r can be determined by looking at the coordinate distance d_r between x_{ir} and x_i, Hence the center pixel x_i should have the highest weight. Therefore, we introduce the weights in the form

$$\omega_r' = \frac{1}{(1 + d_r)^{C_0}}, \tag{2}$$

where C_0 is a certain control parameter. Finally we obtain:

$$\omega_r = \frac{\omega_r'}{\sum_{r=1}^{p} \omega_r'}. \tag{3}$$

Furthermore, we propose a novel method to represent the relationships between pixels. As the similarity measure is a sound way to express it, then we put emphasis on this issue.

The SSIM [39–41] was proposed to measure structural similarity of images. It considers similarity degrees of luminance, contrast, and structure of two images (or image patches). The SSIM index between two image patches (or images) X_1, X_2 is defined as

$$SSIM(X_1, X_2) = \frac{(2\mu_{X_1}\mu_{X_2} + a_1)(2\sigma_{X_1 X_2} + a_2)}{(\mu_{X_1}^2 + \mu_{X_2}^2 + a_1)(\sigma_{X_1}^2 + \sigma_{X_2}^2 + a_2)}, \tag{4}$$

where μ, σ and $\sigma_{X_1X_2}$ act as the mean, standard deviation, and cross correlation between X_1, X_2, respectively. Furthermore $a_1,\ a_2$ are positive constants.

Recall that a distance is a mapping $D: X \times X \rightarrow R$, which satisfies three obvious conditions $(x, y, z \in X)$:

(C1) $D(x, y) \geq 0$, and $D(x, y) = 0 \Leftrightarrow x = y$;
(C2) $D(x, y) = D(y, x)$;
(C3) $D(x, z) \leq D(x, y) + D(y, z)$,

We define

$$DSSIM(X_1, X_2) = C_1(1 - SSIM(X_1, X_2)), \tag{5}$$

where C_1 is a positive constant. It is easy to note that the *DSSIM* (Distance from SSIM) is a distance measure.

For the FLICM algorithm, $1/(d_{ij}+1)$ is actually used to express a spatial similarity degree between two image pixels x_i and x_j, which aims to characterize the relationship between x_i and x_j. In fact, the spatial Euclidean distance d_{ij} only reflects the point-to-point position relationship of x_i and x_j. In contrast, the *DSSIM* distance expresses the relationship between the two patches located around x_i and x_j.

In the sequel, we come up with a new similarity measure:

$$R_{ij} = \frac{1}{1 + d_{ij} + DSSIM(x_i, x_j)}. \tag{6}$$

Here the similarity measure R_{ij} reflects not only the spatial relationship of two pixels x_i and x_j but also the relationship related to luminance and contrast together with structure of two image patches localized around x_i and x_j. Therefore, R_{ij} adequately captures the generalized characteristics of the segmented image, and thus emerges as a more suitable similarity measure.

After that, we propose a novel FCM algorithm based upon the idea of image patch as well as the new similarity measure mentioned above.

The new objective function J_m comes in the form (involving a fuzzy coefficient H_{ki}):

$$J_m = \sum_{i=1}^{N}\sum_{k=1}^{c}\left[u_{ki}^m \sum_{r=1}^{p} \omega_r d(x_{ir}, v_{kr}) + H_{ki}\right]. \tag{7}$$

Here N denotes the number of image pixels. x_i is the gray value of the ith pixel ($i = 1, 2, \ldots, N$). c is the number of clusters. u_{ki} represents the membership grade of x_i with regard to the kth cluster. v_k is the prototype of the kth cluster. Here ω_r, x_{ir}, v_{kr} have been already described in part A ($i = 1, \ldots, N$, $k = 1, \ldots, c$, $r = 1, \ldots, p$). The parameter m is embodied as a weighting exponent (fuzzification coefficient) of the partition matrix $U = [u_{ki}]$. Commonly, we assume that $m = 2$. Recall that the partition matrix satisfies the following obvious requirements:

$$U \in \{u_{ki} | \sum_{k=1}^{c} u_{ki} = 1, \forall i; 0 < \sum_{i=1}^{N} u_{ki} < N, \forall k\}. \tag{8}$$

Based on the above discussion, we define the fuzzy coefficient H_{ki} in the following form:

$$H_{ki} = \sum_{j \in N_i} R_{ij}(1 - u_{kj})^m \sum_{r=1}^{p} \omega_r d(x_{jr}, v_{kr}). \tag{9}$$

Here N_i denotes the set of neighbors (pixels) located in a window around x_i, while x_j stands for the neighboring pixel falling into the window around x_i. The fuzzy coefficient H_{ki} is obtained as the improvement of the fuzzy coefficient G_{ki} used in the FLICM algorithm (expressed as $G_{ki} = \sum_{j \in N_i} \frac{1}{d_{ij}+1}(1 - u_{kj})^m d(x_j, v_k)$). In detail, $\frac{1}{d_{ij}+1}$ is extended to more reasonable similarity measure R_{ij} expressed as (6). Moreover, $d(x_j, v_k)$ is transformed into the corresponding expression from the viewpoint of image patch, i.e., $\sum_{r=1}^{p} \omega_r d(x_{jr}, v_{kr})$. It is similar to $\sum_{r=1}^{p} \omega_r d(x_{ir}, v_{kr})$, where only the center x_i has been changed to x_j. By using the similarity measure R_{ij} and the weighted sum of image patch, it is easy to find that the fuzzy coefficient H_{ki} provides more detailed characterization than G_{ki}.

By virtue of (8), we use the Lagrange multipliers so that we arrive at the unconstrained minimization of J':

$$J' = J_m + \lambda(1 - \sum_{k=1}^{c} u_{ki}). \tag{10}$$

The necessary conditions that lead to the minimum of (10) are expressed as follows:

$$\frac{\partial J'}{\partial u_{ki}} = 0, \quad \frac{\partial J'}{\partial v_{kr}} = 0, \quad i = 1, \ldots, N, \ k = 1, \ldots, c, \quad r = 1, \ldots, p. \tag{11}$$

By setting the gradient of J' to zero with respect to u_{ki} and v_{kr}, we obtain from (11) that

$$u_{ki}^{m-1} = \frac{-\lambda}{m[\sum_{r=1}^{p} \omega_r d(x_{ir}, v_{kr}) + H_{ki}]}, \tag{12}$$

$$\sum_{i=1}^{N}\left[u_{ki}^m (x_{ir} - v_{kr})\right] = 0, \tag{13}$$

From (12) and (8) we can get (14), and from (13) we obtain (15); that is, the iterative updates of the partition matrix and the prototypes come in the form:

$$u_{ki} = \frac{\left(\sum_{r=1}^{p} \omega_r d(x_{ir}, v_{kr}) + H_{ki}\right)^{-1/(m-1)}}{\sum_{j=1}^{c}\left(\sum_{r=1}^{p} \omega_r d(x_{ir}, v_{jr}) + H_{ji}\right)^{-1/(m-1)}}, \tag{14}$$

$$v_{kr} = \frac{\sum_{i=1}^{N} u_{ki}^m x_{ir}}{\sum_{i=1}^{N} u_{ki}^m}. \tag{15}$$

The resulting FCM algorithm is referred to as the Patch-based Fuzzy Local Similarity C-Means (PFLSCM). The details are described in the form of Algorithm 1.

Algorithm 1. The PFLSCM Algorithm.

Step 1. Set values for c, m, ε, $iter$. Determine the size of image patch, and p is obtained.

Step 2. Initialize the fuzzy partition matrix $U^{(0)}$.

Step 3. Compute ω_r by (3), where $r = 1, \ldots, p$.

Step 4. Set the loop counter $b = 0$.

Step 5. Calculate the cluster centers $v_{kr}^{(b)}$ by (15).

Step 6. Calculate $U^{(b+1)}$ by (14).

Step 7. If $\{U^{(b)} - U^{(b+1)}\} < \varepsilon$ or $b > iter$ then stop; otherwise set $b = b$+1 and go to step 5.

Step 8. Assign the pixel x_i to the class C_k by virtue of the biggest membership, i.e., $C_k = \arg\{\max\{u_{ki}\}\}$, which is employed to transform the fuzzy image into the crisp segmented image.

The main characteristics of the PFLSCM algorithm are outlined as follows:

- We use the idea of image patch, which is realized as the weighted sum of image patch and is used to quantify the underlying distance between two pixels.
- A new distance measure DSSIM is derived from SSIM, and then this similarity measure for image pixels is established, which combines spatial distance with the DSSIM distance (as gray distance).
- The fuzzy coefficient standing in (9) is incorporated into the PFLSCM algorithm.

A graphical illustration is given to visualize the main characteristics of the PFLSCM algorithm, see Fig. 1.

Finally, as for the proposed PFLSCM algorithm, let us analyze its effect for three kinds of situations involving noise or outliers.

- Case 1: The central pixel is corrupted by noise while other pixels are not affected by noise, see Fig. 2. Here a 3×3 window (as illustrated in Fig. 2(a)) has been selected in a noisy image with two classes, which lies in the left part of this image. Employing the PFLSCM algorithm, after five

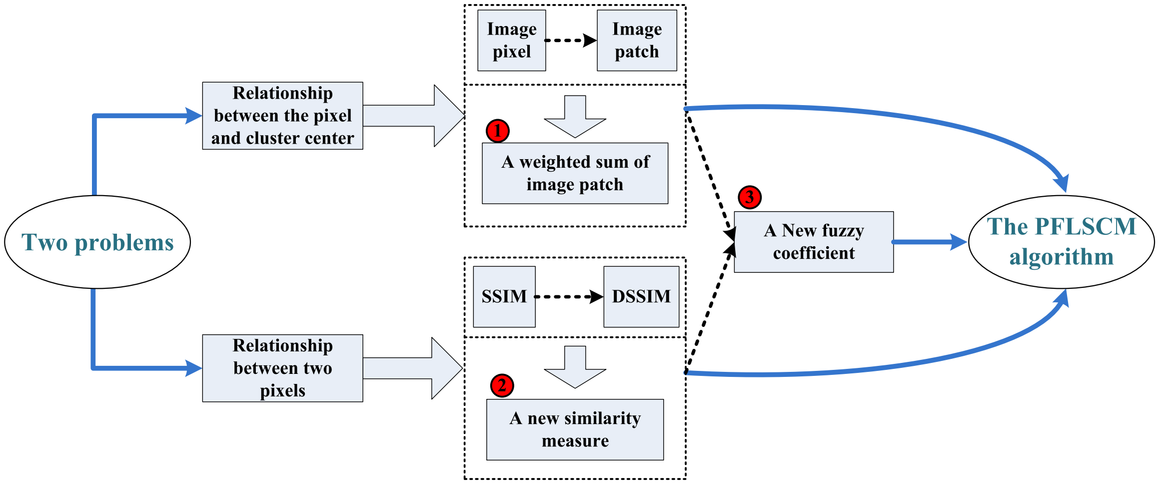

Fig. 1. The main characteristics of the proposed PFLSCM algorithm.

iterations the membership value of the noisy pixel has been changed into the similar value as the neighboring pixels located in this window. The correct classification result is obtained since all of these values are smaller than 0.5. Thus the influence of noise has been eliminated. More precisely, the central noisy pixel has different level of brightness from other pixels in this window, and thus the PFLSCM algorithm makes their membership values gradually converge to exhibit higher resemblance.

- Case 2: The central position is noise-free while some other pixels within this local window have been impacted by noise. Fig. 3 illustrates this situation. Here a 3×3 window with two noisy pixels (Fig. 3(a)) is selected positioned on the right part of the image. After five iterations of the PFLSCM algorithm, all of these pixels in this window come with similar membership grades. The correct classification result is achieved as all these values are higher than 0.5. Thus the impact of noise has been suppressed.
- Case 3: The central position is affected by noise and some other pixels located within its local window are also noisy. Fig. 4 is an illustrative example. Following five iterations of the PFLSCM algorithm, all of these pixels in this window converge to the similar membership value, while the correct classification result is gained.

These three cases and the corresponding examples illustrate some intuition behind robustness of the proposed algorithm. By integrating the idea of image patch, the novel similarity measure expressed as (6) together with the fuzzy coefficient denoted as (9), the robustness of the algorithm is reinforced. As a result, this is a preliminary validation of which the PFLSCM algorithm helps tolerate noise and becomes robust to outliers. Moreover, Section 3 will show more evidence to verify this point.

3. Experimental studies

All experiments include a comprehensive comparative analysis where we engage a number of clustering algorithms studied in the literature. Here we concentrate on the strategy, which incorporates the spatial and gray-level information together in the FCM algorithms. Therefore we compare the PFLSCM algorithm with several representative algorithms including EnFCM, FGFCM, FGFCM_S1, FGFCM_S2, and FLICM. Moreover, in several improvements of the FLICM, the KWFLICM algorithm becomes a successful alternative (see [36] for the details), while the IIFCM

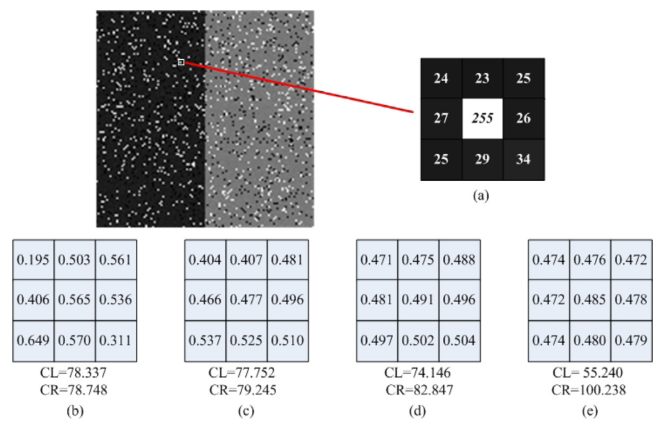

Fig. 2. 3×3 window with central noise (signed with a rectangle), their corresponding membership values and cluster centers (as CL and CR). (a) The chosen window, (b) the initial membership values, (c) after 1 iteration, (d) after 3 iterations, and (e) after 5 iterations.

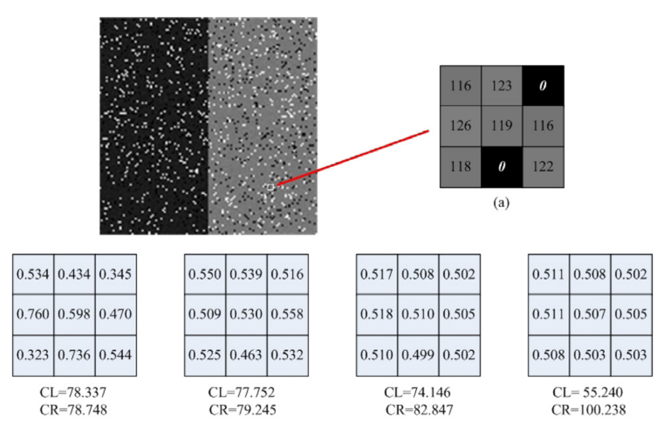

Fig. 3. 3×3 window with two noises around the center, their corresponding membership values and cluster centers. (a) The chosen window, (b) initial membership values, (c) after 1 iteration, (d) after 3 iterations, and (e) after 5 iterations.

algorithm [37] is the latest one. Thus we also compare the proposed algorithm with KWFLICM and IIFCM. Besides, WIPFCM is

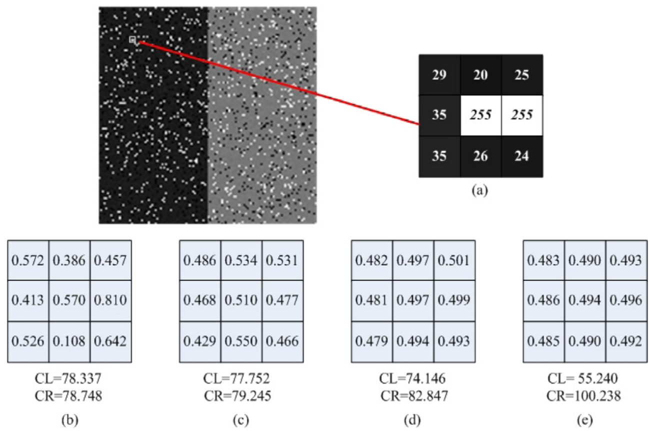

Fig. 4. 3×3 window with noises, their corresponding membership values together the cluster centers. (a) The chosen window, (b) the initial membership values, (c) after 1 iteration, (d) after 3 iterations, and (e) after 5 iterations.

also considered as a reference method. Default values of hyperparameters were used. We investigate the quality of the proposed algorithm by testing it using various synthetic and real images, with different types of noise and characteristics.

3.1. Performance indexes

Here we introduce some performance indexes used to assess the quality of the method. For the testing images with reference results of segmentation, five indexes (i.e., the *SA*, *S*, *PR*, *SP* and *SE*) are considered. For the testing images without reference result, two indexes (*E* and *SNR*) is employed.

There are four well-used criteria for segmentation methods, which are accuracy (*SA*), precision (*PR*), sensitivity (*SE*) and specificity (*SP*). The segmentation accuracy (SA) [27] is the ratio of the number of correctly classified pixels to the total number of pixels. True positives (TP) is the number of positive examples correctly divided. False positives (FP) is the number of positive examples incorrectly classified. False negatives (FN) is the number of instances incorrectly classified as negative. True negatives (TN) is the number of instances correctly classified as negative. Then *PR*, *SE* and *SP* are defined as follows [42]

$$PR = \frac{TP}{TP + FP}, \tag{16}$$

$$SE = \frac{TP}{TP + FN}, \tag{17}$$

$$SP = \frac{TN}{TN + FP}. \tag{18}$$

The higher the value of *SA* (or *PR*, *SE*, *SP*) , the better the segmentation results.

We also employ the following index [30,43]:

$$S = \sum_{i=1}^{c} \frac{|A_i \cap C_i|}{|A_i \cup C_i|}. \tag{19}$$

In (19), c denotes the number of clusters, and A_i stands for the set of pixels belonging to the ith class obtained by the segmentation algorithm, while C_i represented the set of pixels belonging to the ith class in the reference segmented image. The higher the values of S are, the better segmentation performance is.

In [44], an objective evaluation index E with entropy-based information was established to assess the performance of image segmentation. Note that $E = H_l(I) + H_r(I)$, which includes the expected region entropy $H_r(I)$ and the layout entropy $H_l(I)$. The essence of E is that the segmentation should maximize the uniformity of image pixels in every segmented region, while minimize the uniformity in different regions. So the better segmentation performance is characterized by smaller values of E.

Signal-to-noise ratio (*SNR*) [45] is a parameter used to compare the quality of the evaluated image with that of the original image. The higher the *SNR* value is, the better the image quality becomes.

3.2. Tests

In the PFLSCM algorithm, we used the 3 × 3 image patch, and $C_0 = 2.0$, $C_1 = 0.2$, while other algorithms assumed the default values of the parameters. Below we show the process of running the experiments. This is divided by these kinds of image types, which include synthetic images, real-world images and medical images.

To begin with, we evaluate these algorithms with a synthetic two-cluster test image shown in Fig. 5(a). It has the size of 128 × 128 pixels, and contain two clusters with two intensity values of 20 and 120. Different levels of noise, including Gaussian noise, salt-and-pepper noise and impulse noise, are added to the images. Gaussian noise is a kind of noise whose probability density function obeys Gaussian distribution. Salt-and-pepper noise is embodied as random white or black dots. Impulse noise is similar to the salt-and-pepper noise, but its dot values ranges from 0 to 255. They are typical noise types in the area of image segmentation.

Fig. 5 shows the segmentation results of different algorithms with regard to the 15% salt-and-pepper noise impacted synthetic image. As shown in Fig. 5(c)–(g), the EnFCM, FGFCM, FGFCM_S1, FGFCM_S2, FLICM algorithms are affected by noise to different extent. Among them, the EnFCM and FGFCM_S1 algorithms exhibit the worst performance due to the impact of the salt-and-pepper noise, where some misclassification becomes present for the two parts of image. Visually, Fig. 5(f), (g) supports an observation that the FGFCM_S2 and FLICM achieve better performance than the EnFCM, FGFCM and FGFCM_S1. It follows from Fig. 5(h), (i), (j) that the KWFLICM, IIFCM and WIPFCM algorithms yield better performance than FLICM, in which the KWFLICM is better than the IIFCM while IIFCM is greater than WIPFCM. Moreover, it is found from Fig. 5(k) that the proposed PFLSCM algorithm can eliminate almost all the noise.

Table 2 provides the *SA* and *S* values obtained for different algorithms, and Table 3 provides the *PR*, *SP* and *SE* values. Here, 5%, 10%, 15% and 20% of Gaussian, salt-and-pepper, and impulse noises are used. The EnFCM algorithm exhibits the worst performance for all these noises, and the FGFCM_S1 algorithm is detrimentally impacted by the salt-and-pepper noise and impulse noise, while the FGFCM_S2 algorithm is inferior in case of images impacted by Gaussian noise. The FLICM algorithm produces better performance than the EnFCM, FGFCM, FGFCM_S1, FGFCM_S2. The values of WIPFCM is greater than FLICM, while IIFCM, KWFLICM perform better than WIPFCM in these cases. Finally, as shown in Tables 2 and 3 the PFLSCM algorithm comes with the highest values achieve for different noise intensities.

Fig. 6 visualizes the effect of the parameters C_0 and C_1 used in the PFLSCM algorithm. Fig. 6(a) shows the SA value obtained by PFLSCM for 10% Gaussian-noisy synthetic image with regard to different C_0 (here $C_1 = 0.2$). The SA value increases and then decreases when C_0 varies from 0.5 to 6.0. Among them, $C_0 = 1.0$ is the best case where the SA value is biggest. Fig. 6(b) visualizes the SA values produced by PFLSCM for 10% Gaussian-noisy synthetic image versus different C_1 (for fixed $C_0 = 2.0$). The SA value increases and then decreases when C_1 changes from 0.05 to 6.0. Here $C_1 = 0.1$ is the best situation in which the SA value is

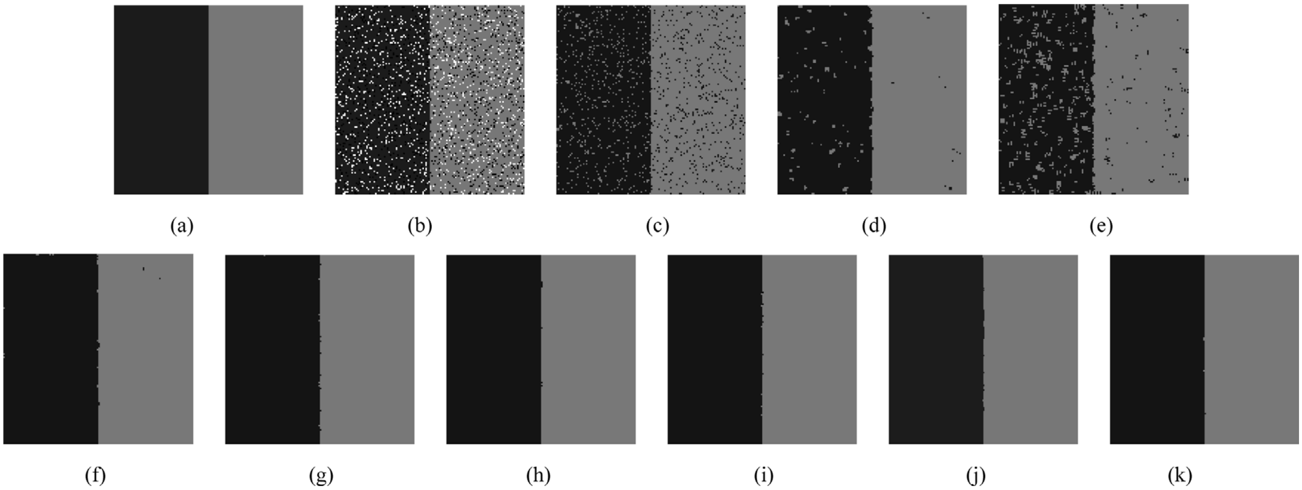

Fig. 5. 15% salt-and-pepper-noised synthetic image. (a) Original image, (b) Noisy image, (c) EnFCM result, (d) FGFCM result, (e) FGFCM_S1 result, (f) FGFCM_S2 result, (g) FLICM result, (h) KWFLICM result, (i) IIFCM result, (j) WIPFCM result, (k) PFLSCM result.

Table 2
The SA and S values of different methods applied to noisy two-cluster images.

	EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
SA for 5% Gaussian noise	0.9377	0.9863	0.9802	0.9742	0.9952	0.9968	0.9965	0.9960	**0.9981**
SA for 10% Gaussian noise	0.8389	0.9347	0.9255	0.9095	0.9631	0.9854	0.9667	0.9646	**0.9879**
SA for 15% Gaussian noise	0.7823	0.8989	0.8782	0.8609	0.9143	0.9501	0.9205	0.9171	**0.9566**
SA for 20% Gaussian noise	0.7451	0.8538	0.8466	0.8286	0.8855	0.8957	0.8889	0.8870	**0.9202**
SA for 5% salt-and-pepper noise	0.9740	0.9980	0.9881	0.9938	0.9883	0.9993	0.9982	0.9971	**0.9995**
SA for 10% salt-and-pepper noise	0.9517	0.9920	0.9649	0.9987	0.9882	0.9996	0.9978	0.9989	**0.9998**
SA for 15% salt-and-pepper noise	0.9282	0.9833	0.9379	0.9844	0.9875	0.9995	0.9972	0.9966	**0.9997**
SA for 20% salt-and-pepper noise	0.9027	0.9699	0.9284	0.9669	0.9843	0.9992	0.9964	0.9932	**0.9994**
SA for 5% impulse noise	0.9867	0.9952	0.9967	0.9947	0.9967	0.9985	0.9966	0.9937	**0.9999**
SA for 10% impulse noise	0.9806	0.9937	0.9903	0.9855	0.9929	0.9983	0.9962	0.9934	**0.9985**
SA for 15% impulse noise	0.9776	0.9936	0.9836	0.9846	0.9905	0.9963	0.9957	0.9929	**0.9966**
SA for 20% impulse noise	0.9675	0.9887	0.9697	0.9833	0.9894	0.9960	0.9949	0.9916	**0.9963**
S for 5% Gaussian noise	1.7654	1.9462	1.9226	1.8995	1.9810	1.9875	1.9861	1.9840	**1.9924**
S for 10% Gaussian noise	1.4438	1.7548	1.7224	1.6681	1.8578	1.9427	1.8711	1.8589	**1.9524**
S for 15% Gaussian noise	1.2812	1.6324	1.5654	1.5115	1.6828	1.8096	1.7054	1.6941	**1.8336**
S for 20% Gaussian noise	1.1840	1.4887	1.4673	1.4140	1.5855	1.6192	1.5996	1.5986	**1.7033**
S for 5% salt-and-pepper noise	1.8988	1.9922	1.9531	1.9982	1.9990	1.9997	1.9992	1.9997	**1.9999**
S for 10% salt-and-pepper noise	1.8157	1.9685	1.8642	1.9951	1.9968	1.9985	1.9978	1.9979	**1.9992**
S for 15% salt-and-pepper noise	1.7323	1.9346	1.7660	1.9934	1.9960	1.9982	1.9970	1.9933	**1.9990**
S for 20% salt-and-pepper noise	1.6455	1.8833	1.7323	1.9878	1.9926	1.9968	1.9948	1.9865	**1.9978**
S for 5% impulse noise	1.9755	1.9815	1.9804	1.9810	1.9869	1.9917	1.9913	1.9801	**1.9998**
S for 10% impulse noise	1.9627	1.9813	1.9802	1.9794	1.9857	1.9904	1.9890	1.9785	**1.9969**
S for 15% impulse noise	1.9123	1.9802	1.9673	1.9766	1.9841	1.9897	1.9765	1.9733	**1.9933**
S for 20% impulse noise	1.8741	1.9795	1.9393	1.9758	1.9834	1.9862	1.9738	1.9706	**1.9855**

best. Fig. 6(c) provides the SA value produced by PFLSCM for 15% salt-and-pepper noisy impacted synthetic image with regard to different C_0 (where $C_1 = 0.2$). The SA first increases and then decreases in which C_0 changes from 0.5 to 6.0. Among them, $C_0 = 1.5$ is the best case where the SA value is biggest. Fig. 6(d) shows the SA value obtained by PFLSCM for 15% salt-and-pepper noisy impacted synthetic image related to different C_1 (where $C_0 = 2.0$). The SA first increases and then decreases where C_1 transforms from 0.05 to 6.0. Here $C_0 = 0.5$ is the ideal case where the SA value is the biggest.

As shown in Fig. 7(a), we used the synthetic three-cluster image of size 256×256 to quantify the performance of the segmentation algorithms. For this image, three intensity values were set as 10, 110 and 228, while the arc boundaries elevated the difficulties of segmentation.

Fig. 7 shows the segmentation results produced by different algorithms when using 10% Gaussian-noise impacted synthetic image. Table 4 shows the values of SA and S obtained for different algorithms; Table 5 provides the values of PR, SP and SE achieved by these algorithms; the noise levels are set as 5%, 10%, 15% and 20% both the Gaussian, salt-and-pepper and impulse noises are used. The EnFCM algorithm exhibits the worst performance: there are many misclassified pixels for the three parts of the segmented image and the resulting values of SA and S are the lowest. The FLICM, WIPFCM and IIFCM algorithms yield acceptable result and the KWFLICM algorithm produces the best results. For the PFLSCM algorithm, only a small amount of noisy pixels in the left part is visible whereas the noise in the middle and right parts is eliminated.

Table 6 shows the values of SA, S, PR, SP and SE of different methods applied to three-cluster images with mixed noises. Here three kinds of mixed noises are employed, i.e., 5% Gaussian noise & 10% salt-and-pepper noise (denoting the mixed noises combined by 5% Gaussian noise and 10% salt-and-pepper noise), 10% Gaussian noise & 15% impulse noise, 5% salt-and-pepper noise & 20% impulse noise. Fig. 8 shows the segmentation results produced by different algorithms when utilizing the mixed noises of 5% Gaussian noise & 10% salt-and-pepper noise. From Table 6 and Fig. 8, we find that EnFCM is the worst one for both the segmented image and the five indexes. Then FGFCM, FGFCM _S1 and FGFCM _S2 also do not perform as well. Following that, FLICM and WIPFCM algorithms are slightly better, in which only

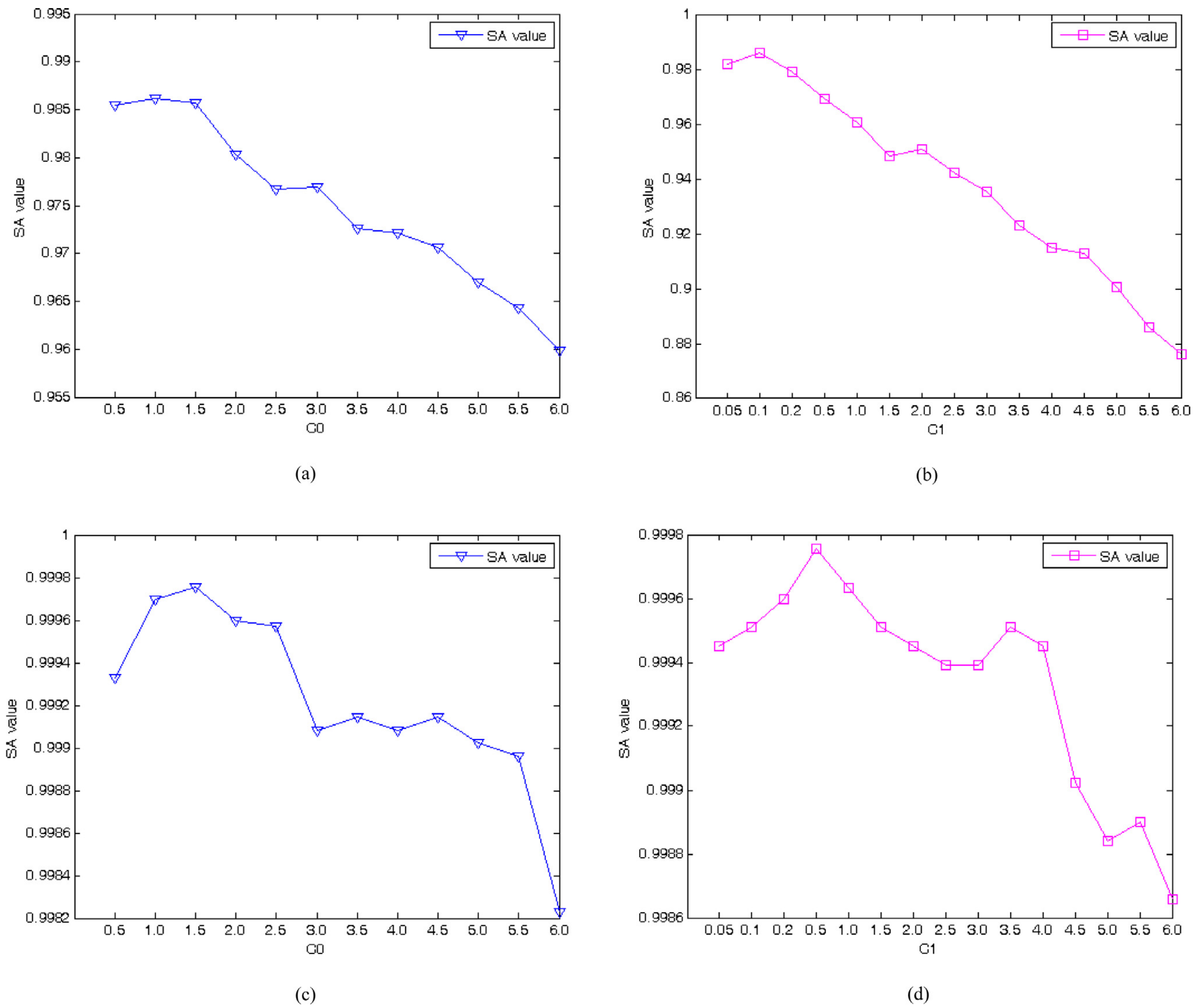

Fig. 6. The SA values obtained by PFLSCM. (a) Results for 10% Gaussian-noisy synthetic image w.r.t. different C_0 (where $C_1 = 0.2$), (b) Results for 10% Gaussian-noised synthetic image w.r.t. different C_1 (where $C_0 = 2.0$). (c) Results for 15% salt-and-pepper-noised synthetic image w.r.t. different C_0 (where $C_1 = 0.2$). (d) Results for 15% salt-and-pepper-noised synthetic image w.r.t. different C_1 (where $C_0 = 2.0$).

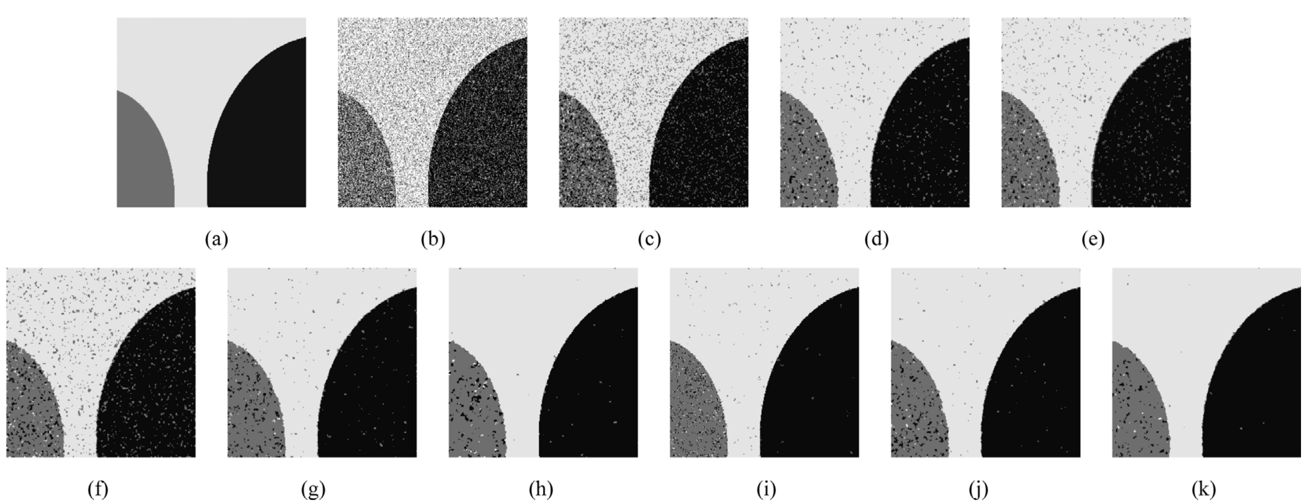

Fig. 7. 10% Gaussian-noisy three-cluster synthetic image. (a) Original image, (b) Noisy image, (c) EnFCM result, (d) FGFCM result, (e) FGFCM_S1 result, (f) FGFCM_S2 result, (g) FLICM result, (h) KWFLICM result, (i) IIFCM result, (j) WIPFCM result, (k) PFLSCM result.

Table 3
The PR, SP and SE values of different methods applied to noisy two-cluster images.

	EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
PR for 5% Gaussian noise	0.9673	0.9796	0.9737	0.9704	0.9856	0.9971	0.9931	0.9925	**0.9972**
PR for 10% Gaussian noise	0.9030	0.9204	0.9079	0.8948	0.9419	0.9755	0.9689	0.9422	**0.9791**
PR for 15% Gaussian noise	0.8537	0.8623	0.8583	0.8376	0.9092	0.9117	0.9110	0.9106	**0.9177**
PR for 20% Gaussian noise	0.7259	0.8371	0.8236	0.7990	0.8382	0.8812	0.8533	0.8439	**0.8975**
PR for 5% salt-and-pepper noise	0.9391	0.9894	0.9636	0.9542	0.9972	0.9996	0.9983	0.9978	**0.9999**
PR for 10% salt-and-pepper noise	0.9247	0.9876	0.9544	0.9521	0.9961	0.9987	0.9979	0.9971	**0.9996**
PR for 15% salt-and-pepper noise	0.8894	0.9865	0.9530	0.9492	0.9927	0.9975	0.9967	0.9939	**0.9995**
PR for 20% salt-and-pepper noise	0.8726	0.9802	0.9393	0.9488	0.9913	0.9972	0.9940	0.9922	**0.9993**
PR for 5% impulse noise	0.9723	0.9786	0.9579	0.9661	0.9782	0.9872	0.9862	0.9799	**0.9999**
PR for 10% impulse noise	0.9485	0.9782	0.9523	0.9491	0.9558	0.9820	0.9808	0.9793	**0.9998**
PR for 15% impulse noise	0.9472	0.9516	0.9496	0.9468	0.9539	0.9551	0.9540	0.9527	**0.9998**
PR for 20% impulse noise	0.9397	0.9465	0.9180	0.9288	0.9491	0.9542	0.9497	0.9503	**0.9994**
SP for 5% Gaussian noise	0.8451	0.9579	0.9384	0.9273	0.9745	0.9971	0.9757	0.9769	**0.9972**
SP for 10% Gaussian noise	0.7980	0.9165	0.9030	0.9022	0.9395	0.9750	0.9726	0.9522	**0.9788**
SP for 15% Gaussian noise	0.7328	0.8521	0.8459	0.8412	0.8523	0.8733	0.8574	0.8524	**0.9108**
SP for 20% Gaussian noise	0.6795	0.8363	0.8090	0.7755	0.8557	0.8598	0.8559	0.8557	**0.8629**
SP for 5% salt-and-pepper noise	0.9592	0.9895	0.9675	0.9688	0.9923	0.9996	0.9943	0.9930	**0.9999**
SP for 10% salt-and-pepper noise	0.9587	0.9877	0.9639	0.9636	0.9923	0.9992	0.9935	0.9925	**0.9996**
SP for 15% salt-and-pepper noise	0.9325	0.9856	0.9555	0.9479	0.9871	0.9903	0.9886	0.9885	**0.9995**
SP for 20% salt-and-pepper noise	0.9261	0.9807	0.9409	0.9441	0.9835	0.9892	0.9872	0.9843	**0.9995**
SP for 5% impulse noise	0.9259	0.9930	0.9782	0.9849	0.9961	0.9986	0.9969	0.9964	**0.9999**
SP for 10% impulse noise	0.9212	0.9909	0.9600	0.9533	0.9940	0.9955	0.9947	0.9941	**0.9998**
SP for 15% impulse noise	0.9143	0.9837	0.9297	0.9405	0.9874	0.9903	0.9882	0.9877	**0.9998**
SP for 20% impulse noise	0.8093	0.9776	0.9286	0.9389	0.9832	0.9898	0.9865	0.9841	**0.9995**
SE for 5% Gaussian noise	0.9425	0.9680	0.9614	0.9722	0.9855	0.9935	0.9872	0.9860	**0.9990**
SE for 10% Gaussian noise	0.9072	0.9661	0.9570	0.9333	0.9748	0.9780	0.9763	0.9760	**0.9962**
SE for 15% Gaussian noise	0.8729	0.9263	0.9230	0.8962	0.9433	0.9656	0.9488	0.9442	**0.9952**
SE for 20% Gaussian noise	0.8684	0.9115	0.8922	0.8922	0.9207	0.9404	0.9375	0.9324	**0.9915**
SE for 5% salt-and-pepper noise	0.9734	0.9982	0.9833	0.9889	0.9983	0.9996	0.9995	0.9993	**0.9999**
SE for 10% salt-and-pepper noise	0.9501	0.9917	0.9303	0.9870	0.9972	0.9982	0.9979	0.9977	**0.9998**
SE for 15% salt-and-pepper noise	0.9595	0.9747	0.8944	0.9799	0.9929	0.9956	0.9952	0.9938	**0.9994**
SE for 20% salt-and-pepper noise	0.9072	0.9418	0.8821	0.9738	0.9851	0.9900	0.9882	0.9864	**0.9991**
SE for 5% impulse noise	0.8926	0.9617	0.9274	0.9579	0.9995	0.9996	0.9705	0.9680	**0.9999**
SE for 10% impulse noise	0.7943	0.9598	0.9265	0.9555	0.9977	0.9986	0.9651	0.9647	**0.9999**
SE for 15% impulse noise	0.7792	0.9522	0.9002	0.9362	0.9928	0.9950	0.9634	0.9527	**0.9993**
SE for 20% impulse noise	0.7601	0.9397	0.8799	0.9090	0.8817	0.9830	0.9616	0.9345	**0.9988**

Table 4
The SA and S values of different methods applied to noisy three-cluster images.

	EnFCM	FGFCM	FGFCM_S1	FGFCM_S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
SA for 5% Gaussian noise	0.9529	0.9878	0.9841	0.9794	0.9935	0.9968	0.9961	0.9952	**0.9971**
SA for 10% Gaussian noise	0.8629	0.9569	0.9486	0.9296	0.9767	0.9818	0.9793	0.9779	**0.9870**
SA for 15% Gaussian noise	0.7936	0.9201	0.9071	0.8696	0.9352	0.9552	0.9415	0.9396	**0.9609**
SA for 20% Gaussian noise	0.7553	0.8763	0.8576	0.8245	0.8946	0.9251	0.9056	0.8963	**0.9336**
SA for 5% salt-and-pepper noise	0.9705	0.9943	0.9856	0.9528	0.9982	0.9995	0.9995	0.9983	**0.9997**
SA for 10% salt-and-pepper noise	0.9435	0.9885	0.9686	0.9507	0.9980	0.9992	0.9991	0.9981	**0.9995**
SA for 15% salt-and-pepper noise	0.9132	0.9744	0.9392	0.9336	0.9875	0.9990	0.9987	0.9979	**0.9992**
SA for 20% salt-and-pepper noise	0.8855	0.9538	0.9093	0.8925	0.9861	0.9987	0.9977	0.9943	**0.9990**
SA for 5% impulse noise	0.9625	0.9894	0.9806	0.9885	0.9957	0.9990	0.9970	0.9959	**0.9993**
SA for 10% impulse noise	0.9471	0.9886	0.9754	0.9883	0.9918	0.9937	0.9929	0.9926	**0.9992**
SA for 15% impulse noise	0.9479	0.9870	0.9732	0.9655	0.9913	0.9922	0.9920	0.9914	**0.9986**
SA for 20% impulse noise	0.9298	0.9759	0.9595	0.9430	0.9875	0.9891	0.9888	0.9882	**0.9978**
S for 5% Gaussian noise	2.2438	2.8903	2.8586	2.8186	2.9613	2.9714	2.9638	2.9627	**2.9741**
S for 10% Gaussian noise	2.1217	2.6452	2.5895	2.4748	2.7926	2.8328	2.8095	2.7998	**2.8805**
S for 15% Gaussian noise	1.8493	2.4105	2.3401	2.1683	2.4801	2.5969	2.5102	2.5024	**2.6499**
S for 20% Gaussian noise	1.7189	2.1809	2.1047	1.9784	2.2452	2.3769	2.3215	2.2511	**2.4252**
S for 5% salt-and-pepper noise	2.7554	2.9480	2.8727	2.7973	2.9752	2.9962	2.9957	2.9765	**2.9971**
S for 10% salt-and-pepper noise	2.5699	2.8987	2.7407	2.7519	2.9717	2.9948	2.9922	2.9754	**2.9958**
S for 15% salt-and-pepper noise	2.3930	2.7869	2.5425	2.4602	2.9475	2.9926	2.9884	2.9616	**2.9939**
S for 20% salt-and-pepper noise	2.2525	2.6460	2.3756	2.4585	2.9400	2.9915	2.9835	2.9592	**2.9924**
S for 5% impulse noise	1.9941	2.8935	2.7065	2.8869	2.9298	2.9723	2.9699	2.9629	**2.9983**
S for 10% impulse noise	1.9908	2.8799	2.6833	2.8637	2.9004	2.8926	2.9476	2.9155	**2.9997**
S for 15% impulse noise	1.9753	2.8359	2.6740	2.8376	2.8500	2.8636	2.8592	2.8537	**2.9961**
S for 20% impulse noise	1.9246	2.8082	2.5677	2.7862	2.8410	2.8628	2.8543	2.8495	**2.9937**

one class (corresponding to right part) is well segmented. The IIFCM and KWFLICM algorithm achieves superior results with good results for two classes (corresponding to middle and right parts). Finally, the PFLSCM algorithm performs best, where good results are obtained for all three categories.

In Tables 2 to 6, we show five indexes (SA, S, PR, SP and SE) of different methods applied to two-cluster image and three-cluster image with one kind of noise or mixed noise. Obviously our proposed PFLSCM algorithm and the WIPFCM algorithm (which both use the idea of image patch) are always better than FLICM, which means that the way of replacing a pixel with an image patch is conducive to improving the image segmentation effect. Moreover, the proposed PFLSCM algorithm is better than the WIPFCM algorithm in image segmentation, which verifies that the proposed image patch distance formula shows a positive improvement of the segmentation algorithm.

Table 5
The PR, SP and SE values of different methods applied to noisy three-cluster images.

	EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
PR for 5% Gaussian noise	0.9460	0.9774	0.9721	0.9661	0.9843	0.9913	0.9896	0.9885	**0.9938**
PR for 10% Gaussian noise	0.9207	0.9214	0.9153	0.7904	0.9599	0.9887	0.9722	0.9708	**0.9929**
PR for 15% Gaussian noise	0.8489	0.8697	0.8521	0.8227	0.9230	0.9679	0.9623	0.9241	**0.9785**
PR for 20% Gaussian noise	0.7945	0.8104	0.8014	0.7770	0.8956	0.9623	0.9598	0.9235	**0.9749**
PR for 5% salt-and-pepper noise	0.9631	0.9869	0.9710	0.9833	0.9885	0.9898	0.9772	0.9238	**0.9991**
PR for 10% salt-and-pepper noise	0.9318	0.9707	0.9406	0.9587	0.9719	0.9725	0.9551	0.9218	**0.9988**
PR for 15% salt-and-pepper noise	0.9114	0.9486	0.9019	0.9240	0.9566	0.9674	0.9335	0.9163	**0.9990**
PR for 20% salt-and-pepper noise	0.8756	0.9197	0.8883	0.7854	0.9200	0.9225	0.9108	0.9058	**0.9982**
PR for 5% impulse noise	0.8489	0.9891	0.9802	0.9785	0.9923	0.9970	0.9955	0.9929	**0.9986**
PR for 10% impulse noise	0.8014	0.9830	0.9669	0.9768	0.9882	0.9947	0.9903	0.9897	**0.9983**
PR for 15% impulse noise	0.6732	0.9727	0.9512	0.9609	0.9799	0.9896	0.9874	0.9853	**0.9972**
PR for 20% impulse noise	0.5728	0.9524	0.9271	0.9483	0.9765	0.9824	0.9782	0.9029	**0.9958**
SP for 5% Gaussian noise	0.9104	0.9945	0.9433	0.9212	0.9957	0.9967	0.9965	0.9958	**0.9984**
SP for 10% Gaussian noise	0.8872	0.9781	0.9429	0.9142	0.9821	0.9926	0.9863	0.9844	**0.9954**
SP for 15% Gaussian noise	0.8860	0.9629	0.9077	0.8995	0.9819	0.9922	0.9846	0.9841	**0.9910**
SP for 20% Gaussian noise	0.8155	0.9427	0.8731	0.8963	0.9652	0.9896	0.9828	0.9762	**0.9825**
SP for 5% salt-and-pepper noise	0.9326	0.9977	0.9944	0.9969	0.9980	0.9992	0.9988	0.9984	**0.9998**
SP for 10% salt-and-pepper noise	0.9105	0.9954	0.9875	0.9893	0.9888	0.9959	0.9975	0.9941	**0.9997**
SP for 15% salt-and-pepper noise	0.8791	0.9893	0.9761	0.9892	0.9886	0.9921	0.9905	0.9893	**0.9996**
SP for 20% salt-and-pepper noise	0.8665	0.9816	0.9635	0.9782	0.9830	0.9904	0.9889	0.9837	**0.9992**
SP for 5% impulse noise	0.9233	0.9428	0.9262	0.9198	0.9432	0.9931	0.9838	0.9444	**0.9997**
SP for 10% impulse noise	0.9215	0.9419	0.9236	0.9093	0.9429	0.9818	0.9691	0.9440	**0.9997**
SP for 15% impulse noise	0.9139	0.9368	0.8999	0.8989	0.9412	0.9679	0.9524	0.9428	**0.9994**
SP for 20% impulse noise	0.8842	0.9302	0.8980	0.8971	0.9399	0.9518	0.9365	0.9207	**0.9990**
SE for 5% Gaussian noise	0.9271	0.9637	0.9600	0.9329	0.9815	0.9938	0.9925	0.9864	**0.9955**
SE for 10% Gaussian noise	0.9297	0.9314	0.9341	0.9229	0.9783	0.9815	0.9806	0.9799	**0.9825**
SE for 15% Gaussian noise	0.8772	0.8941	0.8809	0.8975	0.9678	0.9800	0.9792	0.9755	**0.9804**
SE for 20% Gaussian noise	0.8262	0.8416	0.8342	0.8359	0.8465	0.8797	0.8609	0.8479	**0.9293**
SE for 5% salt-and-pepper noise	0.9325	0.9823	0.9802	0.9723	0.9865	0.9979	0.9958	0.9949	**0.9994**
SE for 10% salt-and-pepper noise	0.9084	0.9815	0.9717	0.9687	0.9822	0.9957	0.9863	0.9842	**0.9993**
SE for 15% salt-and-pepper noise	0.8950	0.9766	0.9445	0.9681	0.9820	0.9936	0.9829	0.9825	**0.9991**
SE for 20% salt-and-pepper noise	0.8779	0.9602	0.9138	0.9063	0.9808	0.9910	0.9827	0.9811	**0.9983**
SE for 5% impulse noise	0.9653	0.9958	0.9915	0.9873	0.9975	0.9990	0.9987	0.9979	**0.9994**
SE for 10% impulse noise	0.9579	0.9933	0.9860	0.9828	0.9964	0.9979	0.9972	0.9968	**0.9992**
SE for 15% impulse noise	0.9278	0.9886	0.9784	0.9805	0.9933	0.9955	0.9948	0.9935	**0.9987**
SE for 20% impulse noise	0.9009	0.9794	0.9651	0.9792	0.9891	0.9930	0.9915	0.9892	**0.9979**

Table 6
The SA, S, PR, SP and SE values of different methods applied to three-cluster images with mixed noises.

	EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
SA for 5% Gaussian noise & 10% salt-and-pepper noise	0.8085	0.8683	0.8193	0.8535	0.8787	0.9860	0.9680	0.8916	**0.9927**
SA for 10% Gaussian noise & 15% impulse noise	0.7887	0.8427	0.8590	0.8295	0.8641	0.9753	0.9377	0.8914	**0.9924**
SA for 5% salt-and-pepper noise & 20% impulse noise	0.6978	0.7823	0.7315	0.7811	0.8333	0.9232	0.9036	0.8907	**0.9375**
S for 5% Gaussian noise & 10 salt-and-pepper noise	2.1490	2.2564	2.1741	2.2497	2.6357	2.9242	2.9082	2.7654	**2.9653**
S for 10% Gaussian noise & 15% impulse noise	2.0002	2.2361	2.1533	2.0338	2.3195	2.8562	2.7117	2.6457	**2.9549**
S for 5% salt-and-pepper noise & 20% impulse noise	1.6384	1.8562	1.6628	1.7477	1.9751	2.7608	2.5360	2.2277	**2.8971**
PR for 5% Gaussian noise & 10 salt-and-pepper noise	0.7222	0.8918	0.8440	0.8810	0.8933	0.9819	0.9537	0.9126	**0.9888**
PR for 10% Gaussian noise & 15% impulse noise	0.6799	0.8733	0.8328	0.8210	0.8795	0.9699	0.9126	0.9044	**0.9819**
PR for 5% salt-and-pepper noise & 20% impulse noise	0.6636	0.8670	0.7153	0.7420	0.7945	0.9020	0.8717	0.8319	**0.9449**
SP for 5% Gaussian noise & 10 salt-and-pepper noise	0.9058	0.9287	0.9166	0.9251	0.9328	0.9929	0.9529	0.9425	**0.9965**
SP for 10% Gaussian noise & 15% impulse noise	0.8408	0.9285	0.9022	0.8891	0.9267	0.9872	0.9512	0.9323	**0.9951**
SP for 5% salt-and-pepper noise & 20% impulse noise	0.7782	0.8840	0.8195	0.8545	0.9131	0.9607	0.9526	0.9426	**0.9683**
SE for 5% Gaussian noise & 10 salt-and-pepper noise	0.6822	0.8405	0.7247	0.7499	0.8786	0.9747	0.9694	0.9218	**0.9884**
SE for 10% Gaussian noise & 15% impulse noise	0.6667	0.7975	0.7047	0.6779	0.7732	0.9521	0.9039	0.8819	**0.9850**
SE for 5% salt-and-pepper noise & 20% impulse noise	0.5721	0.6187	0.5994	0.6025	0.6584	0.9203	0.8453	0.7426	**0.9657**

Furthermore, for the real-world images, the ground truth normally does not exist. Therefore, the no-reference evaluation method is more convenient here. Thus we use the entropy-based evaluation criterion E and signal-to-noise ratio SNR to compare the result. Aiming at three classical real-world images, i.e., the Coins image, the House image, and the Cameraman image [46], we utilized the eight algorithms to carry a segmentation process. Fig. 9 shows the segmentation results for the 5% salt-and-pepper noise impacted Coins image. Fig. 10 shows the results for the 5% Gaussian-noised House image. Following that, Fig. 11 provides the results for the 10% salt-and-pepper-noisy Cameraman image. Table 7 shows the corresponding E values for different algorithms and images, and Table 8 provides the SNR values for different methods on noisy real-world images.

For the 5% salt-and-pepper-noisy Coins image (see Fig. 9), EnFCM exhibits the worst performance and FGFCM_S1 is the next one. The structure grasped in the segmented image of FGFCM_S1 and EnFCM is rather unclear. Then, FLICM, FGFCM and FGFCM_S2 perform better than FGFCM_S1 and EnFCM. KWFLICM, IIFCM and WIPFCM also exhibit poor performance which lose some local structure. The proposed PFLSCM algorithm yields the best segmentation results, see Fig. 9; its E values are the smallest as well, see Table 7; its SNR values are the biggest one, see Table 8. Therefore, the PFLSCM algorithm comes with the best segmentation performance. For the 5% Gaussian-noise impacted House image (Fig. 10) and the 10% salt-and-pepper noise impacted Cameraman image (see Fig. 11), the conclusions are similar. The worst result is reported for the EnFCM method, whereas the best is produced by the PFLSCM.

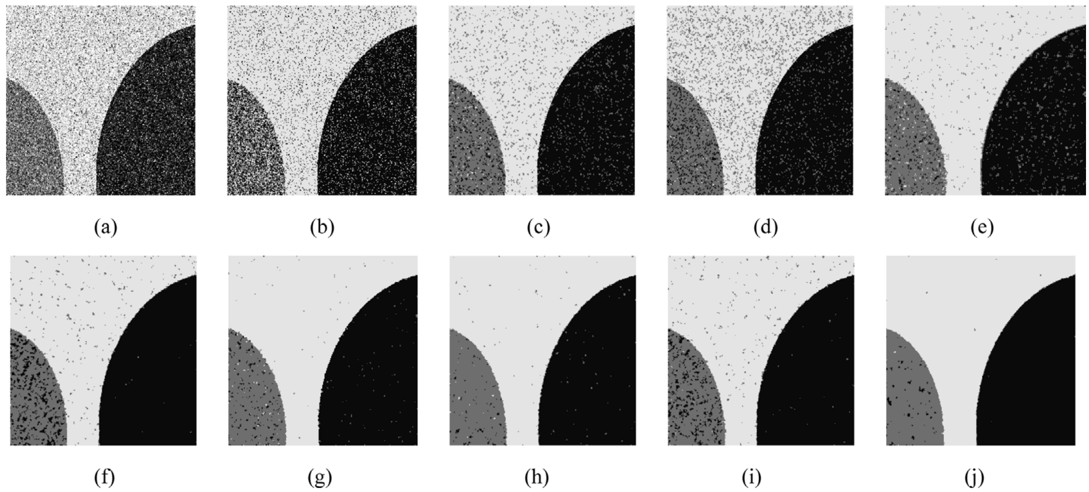

Fig. 8. Three-cluster synthetic image with 5% Gaussian noise & 10% salt-and-pepper noise. (a) Noisy image, (b) EnFCM result, (c) FGFCM result, (d) FGFCM_S1 result, (e) FGFCM_S2 result, (f) FLICM result, (g) KWFLICM result, (h) IIFCM result, (i) WIPFCM result, (j) PFLSCM result.

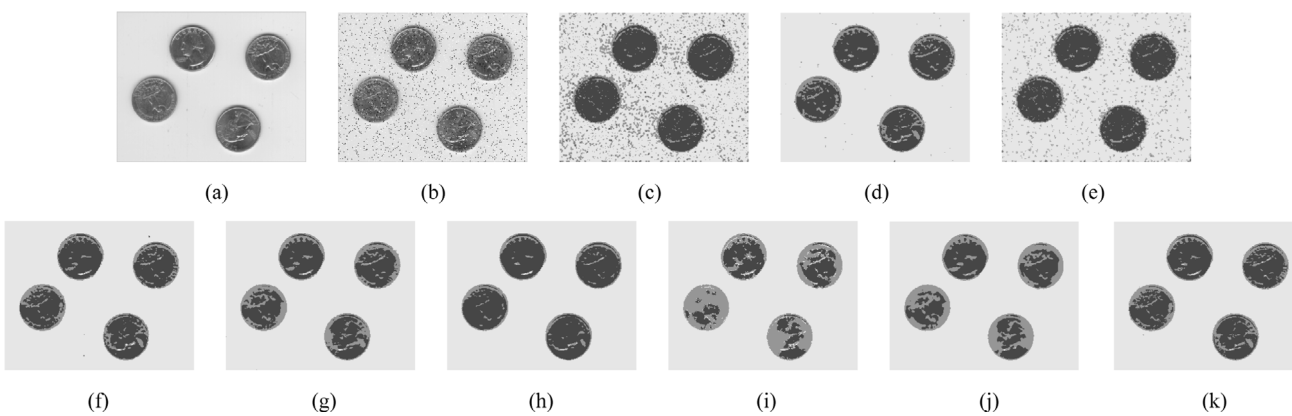

Fig. 9. 5% salt-and-pepper-noisy Coins image. (a) Original image, (b) Noisy image, (c) EnFCM result, (d) FGFCM result, (e) FGFCM_S1 result, (f) FGFCM_S2 result, (g) FLICM result, (h) KWFLICM result, (i) IIFCM result, (j) WIPFCM result, and (k) PFLSCM result.

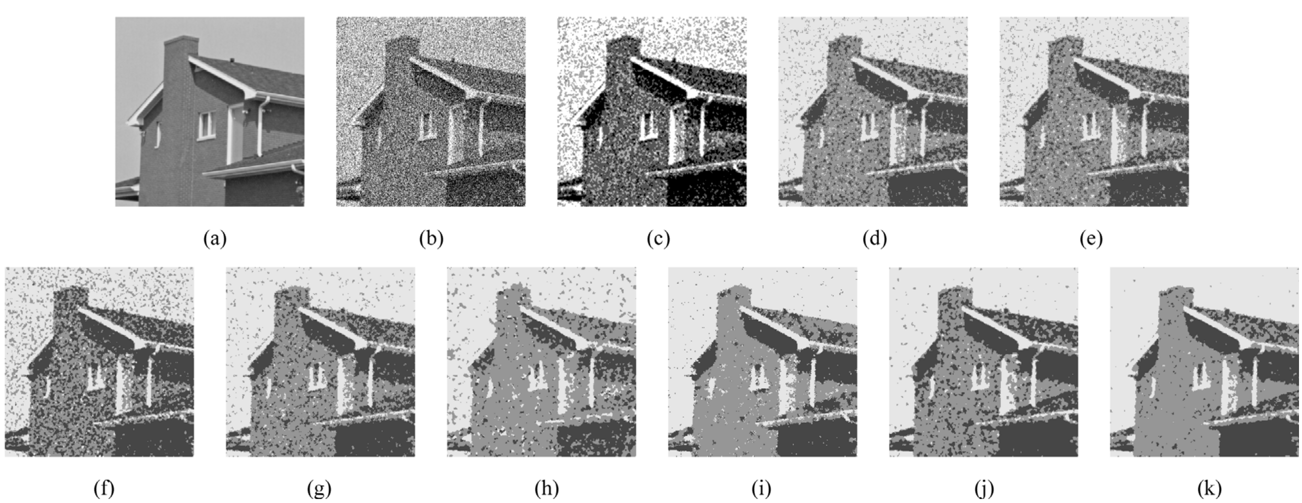

Fig. 10. 5% Gaussian-noisy House image. (a) Original image, (b) Noisy image, (c) EnFCM result, (d) FGFCM result, (e) FGFCM_S1 result, (f) FGFCM_S2 result, (g) FLICM result, (h) KWFLICM result, (i) IIFCM result, (j) WIPFCM result, and (k) PFLSCM result.

Finally, we complete experiments for medical images. Here we use the BrainWeb image [47], which is high-resolution T2_weighted phantom with slice thickness of 1 mm resolution, 40% intensity non-uniformity, 9% Rician noise, leading to a size of 181×217×181 voxels. It is noted that reference images are available on the website. We use two slices in the axial plane, as shown in Figs. 12(a) and 13(a).

The results produced by the nine algorithms are given in Figs. 12 and 13. Table 9 reports the SA and S values of different methods on the noisy image. We find that the EnFCM algorithm

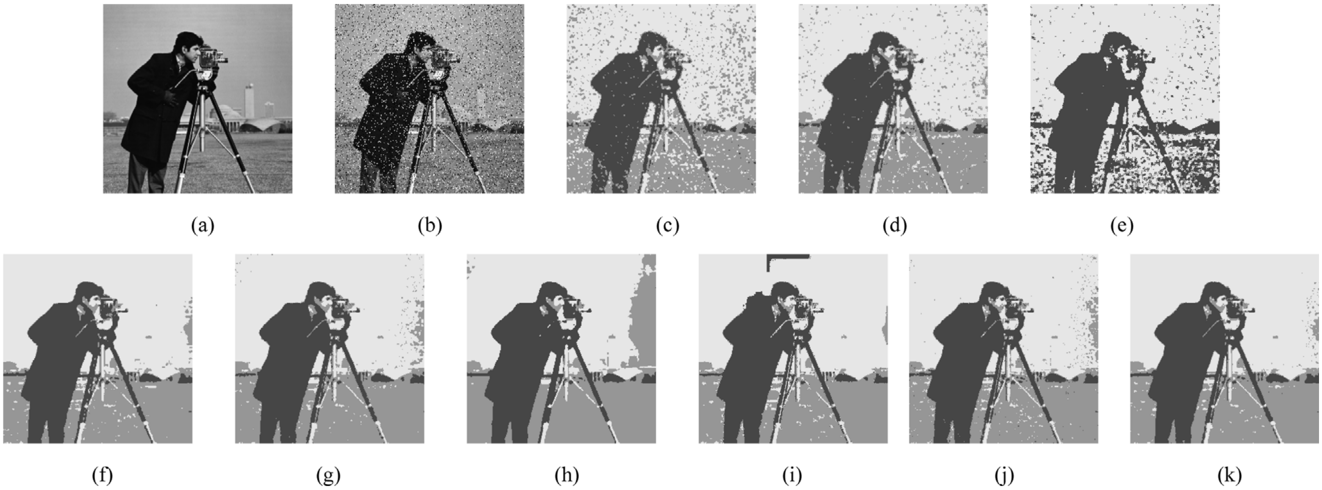

Fig. 11. 10% salt-and-pepper-noisy Cameraman image. (a) Original image, (b) Noisy image, (c) EnFCM result, (d) FGFCM result, (e) FGFCM_S1 result, (f) FGFCM_S2 result, (g) FLICM result, (h) KWFLICM result, (i) IIFCM result, (j) WIPFCM result, and (j) PFLSCM result.

Table 7
The E values for different methods on noisy real-world images.

Image	Metric	EnFCM	FGFCM	FGFCM_S1	FGFCM_S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
Coins	E	5.7265	5.4412	5.6919	5.4342	5.4088	5.3887	5.3911	5.4078	**5.3762**
	H_r	4.3831	4.4210	4.5339	4.4323	4.3850	4.4104	4.3970	4.3853	4.3448
	H_l	1.3434	1.0202	1.1580	1.0019	1.0238	0.9783	0.9941	1.0225	1.0314
House	E	9.1160	9.0024	9.0050	8.9515	8.7962	8.7901	8.7943	8.7957	**8.7886**
	H_r	7.1482	7.4436	7.4473	7.3671	7.2306	7.3247	7.2525	7.2385	7.2412
	H_l	1.9678	1.5588	1.5577	1.5844	1.5656	1.4653	1.5418	1.5572	1.5474
Cameraman	E	7.4423	7.3798	7.4011	7.2603	7.2466	7.2424	7.2458	7.2446	**7.2349**
	H_r	5.9032	5.8508	6.3383	5.7347	5.7144	5.6810	5.7195	5.7124	5.7081
	H_l	1.5391	1.5290	1.0628	1.5256	1.5322	1.5614	1.5263	1.5322	1.5268

Table 8
The SNR values for different methods on noisy real-world images.

Image	EnFCM	FGFCM	FGFCM_S1	FGFCM_S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
Coins	4.1583	4.5138	4.1727	4.2482	4.5284	5.2934	4.9464	4.6658	**5.6970**
House	2.5691	3.7201	3.0339	2.7550	4.0641	6.0840	5.8103	5.2946	**6.9147**
Cameraman	6.8518	7.3040	7.3011	7.3038	7.4090	8.2602	7.4383	7.4114	**8.7739**

comes with the worst performance and that FGFCM_S1 is the next one. Some detailed structural features are missed by the KWFLICM, IIFCM and WIPFCM. The proposed PFLSCM algorithm achieves the best performance by retaining the details of the image and eliminating noise present in the original image.

Fig. 14 illustrates the computational cost for images of different sizes when running EnFCM, FGFCM, FGFCM_S1, FGFCM_S2, FLICM, KWFLICM, and PFLSCM. We added Gaussian together with salt-and-pepper noise to these testing images. All experiments were performed on a Pentium IV (2.6 GHz) under Windows 10 Professional using MATLAB and VC++ 2008. FGFCM_S1 is much faster than other algorithms, while KWFLICM is the slowest one and IIFCM is the next one. The running time of the proposed PFLSCM algorithm is similar to FLICM and shorter than KWFLICM as well as IIFCM.

To investigate the effect of different image patches of the PFLSCM method, Table 10 shows the SA values for the PFLSCM method applied to noisy three-cluster synthetic images (Fig. 7(a)) with different patches. From Table 10, aiming at different salt-and-pepper noise, the image patch with size 3*3 can let PFLSCM produce the highest SA value. Under the same noise density condition, with the increase of patch size, the performance of PFLSCM algorithm decreases gradually. Especially when 20% salt and pepper noise is added, the performance decline trend is the most obvious. This is because as the patch size increases, the noise which it contains increases and naturally it gradually interferes with the PFLSCM algorithm for image segmentation. Based on this reason, here we all use the size of 3*3 to reveal the performance of PFLSCM.

Table 11 shows the t-test of the PFLSCM method with regard to other algorithms, where the parameter α of t-test employs 0.05 and the index is SA. Here 3.21E−07 means $3.21^{*}10^{-7}$ and the others are similar. The data tested here shows that the SA value of the proposed PFLSCM is higher than other algorithms in each image. After one-tail t-test, the probability P is less than 0.01 when PFLSCM is compared with the other 8 comparison algorithms. That is to say, it is very significant that the SA value of PFLSCM is higher than other algorithms, especially for FGFCM_S. Moreover, after two-tail t-test, the probability P is less than 0.01 when PFLSCM is compared with other eight comparison algorithms except KWFLICM. Thus, the SA value of the PFLSCM is significantly higher than that of EnFCM, FGFCM, FGFCM_S1, FGFCM_S2, FLICM, IIFCM and WIPFCM. As for KWFLICM, the probability P is bigger than 0.01 and smaller than 0.05, which means that the SA value of the PFLSCM is also significantly better than KWFLICM. To sum up, the t-test proves that the performance of the proposed PFLSCM method is significantly better than other algorithms.

Table 12 shows the Monte Carlo simulation of these algorithms, which depends on the SA value. Here we do 10,000 Monte Carlo simulations, and take its mean value for these nine

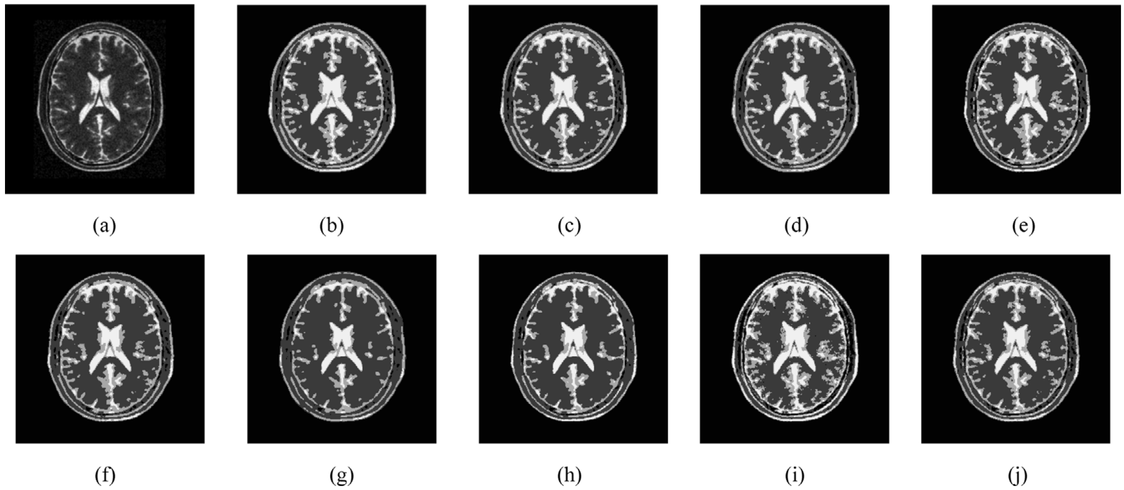

Fig. 12. Medical image (I). (a) Original image, (b) EnFCM result, (c) FGFCM result, (d) FGFCM_S1 result, (e) FGFCM_S2 result, (f) FLICM result, (g) KWFLICM result, (h) IIFCM result, (i) WIPFCM result, and (j) PFLSCM result.

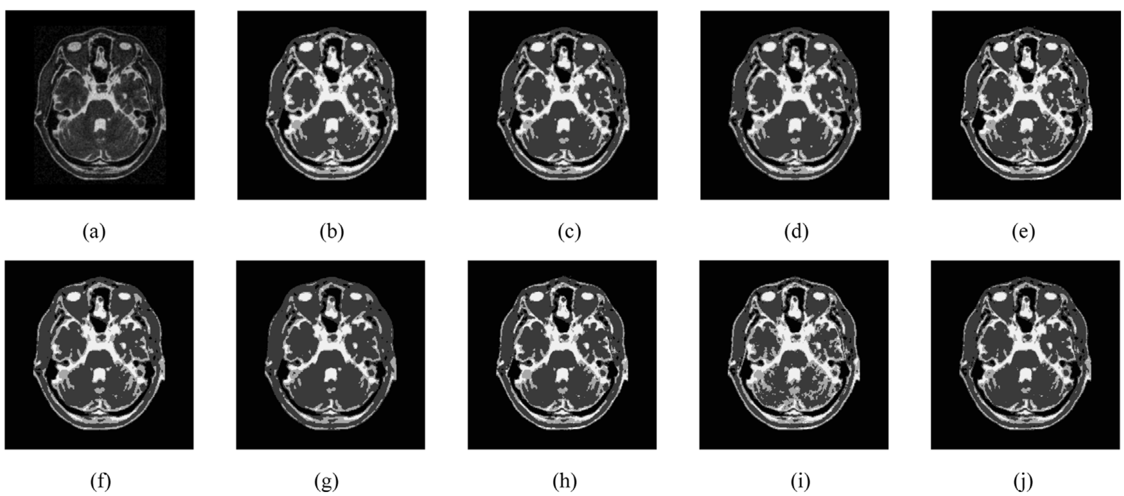

Fig. 13. Medical image (II). (a) Original image, (b) EnFCM result, (c) FGFCM result, (d) FGFCM_S1 result, (e) FGFCM_S2 result, (f) FLICM result, (g) KWFLICM result, (h) IIFCM result, (i) WIPFCM result, and (j) PFLSCM result.

Table 9
The SA and S values for different methods on noisy medical images.

Image	Metric	EnFCM	FGFCM	FGFCM_S1	FGFCM_S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
Noisy Medical Images (i)	SA	0.9194	0.9328	0.9281	0.9301	0.9371	0.9395	0.9389	0.9387	**0.9446**
	S	2.9475	3.0688	3.0321	3.0323	3.1146	3.1221	3.1152	3.1150	**3.2070**
Noisy Medical Images (ii)	SA	0.9255	0.9388	0.9320	0.9365	0.9409	0.9420	0.9411	0.9413	**0.9512**
	S	3.0292	3.1339	3.0704	3.1135	3.1505	3.4169	3.2061	3.1902	**3.5917**

Table 10
The SA values for the PFLSCM method applied to noisy three-cluster synthetic images with different patches.

	Size of image patch				
	3*3	5*5	7*7	9*9	11*11
5% salt-and-pepper noise	**0.9997**	0.9996	0.9996	0.9995	0.9993
10% salt-and-pepper noise	**0.9995**	0.9993	0.9990	0.9985	0.9981
15% salt-and-pepper noise	**0.9992**	0.9991	0.9987	0.9964	0.9950
20% salt-and-pepper noise	**0.9990**	0.9988	0.9938	0.9882	0.9874

algorithms. It can be found that our proposed PFLSCM method has the average SA value of 0.9858, which is the biggest one. All algorithm performance relations satisfy EnFCM < FGFCM_S2 < FGFCM_S1 < FGFCM < FLICM < WIPFCM < IIFCM < KWFLICM < PFLSCM. From the statistical point of view, it is reliable that our PFLSCM method has better performance than other comparative algorithms.

3.3. Discussion

The comprehensive experiments lead to a number of well documented conclusions:

(i) There exists some information loss in the EnFCM algorithm, since it uses a linearly weighted sum image to accelerate the computing process. This new image is only obtained from gray

Table 11
The t-test of the PFLSCM method with regard to other algorithms.

P	EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM
One-tail test	3.21E−07	4.67E−06	6.51E−08	2.44E−06	3.3E−06	0.00111	0.00034	0.00010
Two-tail test	6.41E−07	9.33E−06	1.3E−07	4.89E−06	6.6E−06	0.00222	0.00069	0.00020

Table 12
The Monte Carlo simulation of these algorithms.

EnFCM	FGFCM	FGFCM _S1	FGFCM _S2	FLICM	KWFLICM	IIFCM	WIPFCM	PFLSCM
0.9101	0.9637	0.9479	0.9442	0.9728	0.9818	0.9777	0.9757	0.9858

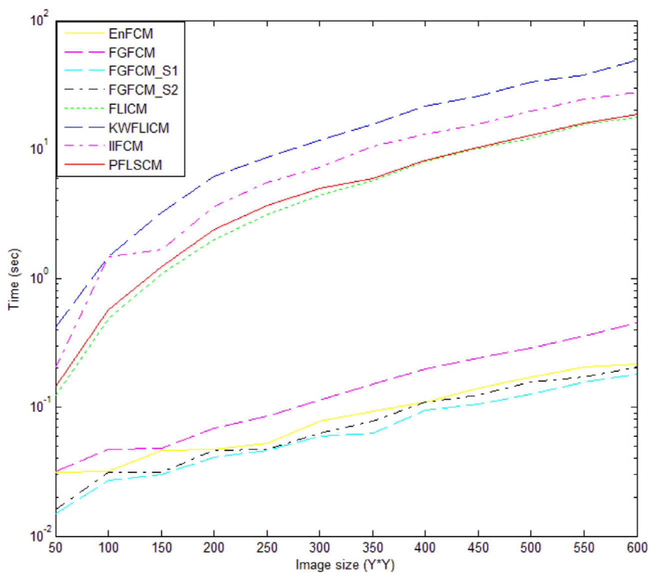

Fig. 14. Computational cost of the algorithms.

level information, which does not consider spatial information. From the experimental results, this effect on the EnFCM is the worst one.

(ii) Similar to EnFCM, the FGFCM algorithm also utilizes the linearly weighted sum image. However, this new image is derived from a local similarity measure combining both spatial and gray level information. Thus FGFCM, has better performance than EnFCM. Information loss in FGFCM cannot be avoided. Besides, FGFCM_S1 and FGFCM_S2 are two reduced versions of the FGFCM method. Generally speaking, FGFCM, FGFCM_S1 and FGFCM_S2 are superior over the EnFCM.

(iii) For the FLICM algorithm, there are no information losses as they are avoided by using the original image. The experimental results show that the method is better than EnFCM, FGFCM, FGFCM_S1, and FGFCM_S2. However, as mentioned in Section 1, it is not proper to depict the relationship between the image pixel and the cluster center, and also the relationship between two pixels in the FLICM algorithm.

(iv) As an improvement to FLICM, the KWFLICM algorithm inherits its main features and demonstrates better performance. KWFLICM uses a non-Euclidean distance to express the relationship between the image pixel and the cluster center, however it also depends on the gray values of these two points. And it employs the local coefficient of variation to represent the relationship between the two pixels. This is helpful but not sufficient to fully capture the relationship between two pixels. KWFLICM is better than FLICM to characterize the relationship for the image elements. Besides, the computing time of KWFLICM is the highest in comparison with other algorithms studied here.

(v) IIFCM is another improvement of FLICM. Its main contribution lies in that it depends on the intuitionistic fuzzy value to represent image elements, however it is only related to image gray, and the configuration of intuitionistic fuzzy value is simple and lack of clear interpretation. Thus the relationship for the basic image element is also not adequately expressed. From experiment results, IIFCM is better than FLICM while it is not as good as KWFLICM. Meanwhile, IIFCM seems a little difficult to distinguish detailed structure features for noised images.

(vi) WIPFCM employs image patches to replace pixels with a weighting scheme. From experiment results, WIPFCM is better than FLICM while it is not as good as KWFLICM and IIFCM. Moreover, we show the difference of WIPFCM and PFLSCM. First, in WIPFCM, only the idea of image patch is employed. But, in PFLSCM, the computing method for image patch is different from WIPFCM, and a new distance measure and similarity measure are put forward to characterize the spatial relationships between two image pixels but also the dependencies of two image patches revolving around them. Second, WIPFCM is basically positioned along the lines of FCM, which leads to its standard computing mechanism. However, PFLSCM inherits the advantages of FLICM and utilizes fuzzy coefficient to make the computing model perform more reasonable. Finally, by experiments we find that the performance of PFLSCM is better than WIPFCM.

(vii) PFLSCM fully characterizes the relationship among the elements of the image. For one thing, to properly analyze the relationship between the image pixel and the cluster center, the idea of image patch is introduced for both the current image pixel and the cluster center, because the image patches imply more general information than image pixels. For another, aiming at the relationship between two image pixels, we establish a new similarity measure to compute the similarity between two image pixels in the whole image. The new similarity measure expresses not only the spatial relationships between two image pixels but also the dependencies related with luminance, contrast, and structure of two image patches formed around them. Consequently, PFLSCM fully characterizes the relationship among the elements of the image, and then overcomes the weaknesses of FLICM. These also demonstrate where PFLSCM improves on FLICM. The computing cost of PFLSCM is similar to the one of the FLICM, and is lower than the one encountered in KWFLICM and IIFCM. The PFLSCM method has the highest ability to preserve the details of the image and eliminate noise.

In sum, the proposed PFLSCM algorithm has the highest ability to retain the details of the image and eliminate noise. It performs the best among these algorithms from the angles of design mechanism, evaluation metrics and visual perceptions.

4. Conclusions

In the FCM clustering algorithm used for image segmentation, a crucial issue is how to appropriately characterize the relationship between the image pixel and the cluster center, as well as the relationship between two image pixels. To properly characterize these two relationships, a novel FCM algorithm called PFLSCM algorithm is put forward and investigated.

The research of the proposed PFLSCM approach is summarized as follows. To begin with, the idea of image patches is introduced for not only the current image pixel but also the cluster center. The basic distance is restructured into a weighted sum of image patch. Furthermore, a novel local distance measure via the structural similarity (SSIM) index is presented to calculate the distance between two pixels, and based on it a new similarity measure is constructed. Such similarity measure includes the spatial relationships between two pixels as well as the dependencies related with luminance, contrast, and structure of two patches revolved around them. Following that, a new fuzzy coefficient is provided using the new similarity measure as well as the weighted sum distance of image patch, and then the PFLSCM algorithm is established. Finally, from the algorithm design mechanism and testing for the synthetic, real-world and medical images, the proposed PFLSCM algorithm is better than related comparative algorithms from the angles of both performance indexes and visual effects.

In the future, the idea of possibilistic c-means clustering can be introduced into the PFLSCM algorithm, then new object function and segmentation strategy can be obtained. What is more, we can combine the PFLSCM algorithm with an advanced denoising algorithm and then integrate it into a new algorithm, which may lead to the better segmentation results for images with significant levels of noise. In further studies, it is worth investigating how to adjust the PFLSCM algorithm to segment synthetic aperture radar images and color images. Furthermore it may be a good way to employ granular information to express image pixel or patch. How to utilize granular expression [48,49] to investigate image segmentation based on fuzzy clustering, could be of interest in further studies.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2019.105928.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. 61673156, 61432004, U1613217, 61672202), the Fundamental Research Funds for the Central Universities of China (No. ACAIM190101), the Natural Science Foundation of Anhui Province (Nos.1408085MKL15, 1508085QF129), and the China Postdoctoral Science Foundation (Nos. 2012M521218, 2014T70585).

References

[1] K.K. Maninis, J. Pont-Tuset, P. Arbeláez, L.V. Gool, Convolutional oriented boundaries: From image segmentation to high-level tasks, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 819–833.
[2] C. Chen, A. Zare, H.N. Trinh, et al., Partial membership latent Dirichlet allocation for soft image segmentation, IEEE Trans. Image Process. 26 (12) (2017) 5590–5602.
[3] Y. Pan, Y. Xia, T. Zhou, M. Fulham, Cell image segmentation using bacterial foraging optimization, Appl. Soft Comput. 58 (2017) 770–782.
[4] Y.M. Tang, X.H. Hu, W. Pedrycz, X.C. Song, Possibilistic fuzzy clustering with high-density viewpoint, Neurocomputing 329 (2019) 407–423.
[5] C. Rupprecht, L. Peter, N. Navab, Image segmentation in twenty questions, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. CVPR, Boston, MA, USA, 2015, pp. 3314–3322.
[6] F.L. Hernández, E.G. de Ory, S.R. Aguilar, R.G. Crespo, Residue properties for the arithmetical estimation of the image quantization table, Appl. Soft Comput. 67 (2018) 309–321.
[7] P. Kushwaha, R.R. Welekar, Feature selection for image retrieval based on genetic algorithm, Int. J. Interact. Multimed. Artif. Intell. 4 (2) (2017) 16–21.
[8] A.A. Rezaie, A. Habiboghli, Detection of lung nodules on medical images by the use of fractal segmentation, Int. J. Interact. Multimed. Artif. Intell. 4 (5) (2017) 15–19.
[9] U. Javed, M.M. Riaz, A. Ghafoor, T.A. Cheema, SAR image segmentation based on active contours with fuzzy logic, IEEE Trans. Aerosp. Electron. Syst. 52 (1) (2016) 181–188.
[10] W. Pedrycz, A. Bargiela, An optimization of allocation of information granularity in the interpretation of data structures: Toward granular fuzzy clustering, IEEE Trans. Syst. Man Cybern. B 42 (3) (2012) 582–590.
[11] Y.M. Tang, F.J. Ren, Variable differently implicational inference for R- and S-implications, Int. J. Inf. Tech. Decis. 15 (5) (2016) 1235–1264.
[12] H. Izakian, W. Pedrycz, I. Jamal, Clustering spatiotemporal data: An augmented fuzzy c-means, IEEE Trans. Fuzzy Syst. 21 (5) (2013) 855–868.
[13] S. Balla-Arabe, X.B. Gao, B. Wang, A fast and robust level set method for image segmentation using fuzzy clustering and lattice Boltzmann method, IEEE Trans. Cybern. 43 (3) (2013) 910–920.
[14] Y. Tolias, S. Panas, Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions, IEEE Trans. Syst. Man Cybern. A 28 (3) (1998) 359–369.
[15] D.L. Pham, J.L. Prince, An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities, Pattern Recognit. Lett. 20 (1) (1999) 57–68.
[16] Z.M. Wang, Q. Song, Y.C. Soh, K. Sim, An adaptive spatial information-theoretic fuzzy clustering algorithm for image segmentation, Comput. Vis. Image Underst. 117 (10) (2013) 1412–1420.
[17] D.G. Zhou, H. Zhou, A modified strategy of fuzzy clustering algorithm for image segmentation, Soft Comput. 19 (11) (2015) 3261–3272.
[18] Z.X. Ji, J.Y. Liu, G. Cao, Q.S. Sun, Q. Chen, Robust spatially constrained fuzzy c-means algorithm for brain MR image segmentation, Pattern Recognit. 47 (7) (2014) 2454–2466.
[19] S.K. Adhikari, J.K. Sing, D.K. Basu, M. Nasipuri, Conditional spatial fuzzy C-means clustering algorithm for segmentation of MRI images, Appl. Soft Comput. 34 (2015) 758–769.
[20] S.P. Chatzis, T.A. Varvarigou, A fuzzy clustering approach toward hidden Markov random field models for enhanced spatially constrained image segmentation, IEEE Trans. Fuzzy Syst. 16 (5) (2008) 1351–1361.
[21] G. Liu, Y. Zhang, A. Wang, Incorporating adaptive local information into fuzzy clustering for image segmentation, IEEE Trans. Image Process. 24 (11) (2015) 3990–4000.
[22] R.P. Li, M. Mukaidono, A maximum entropy approach to fuzzy clustering, in: Proc. IEEE Int. Conf. Fuzzy Syst. FUZZ-IEEE, Yokohama, Japan, 1995, pp. 2227–2232.
[23] S. Miyamoto, K. Umayahara, Fuzzy clustering by quadratic regularization, in: Proc. IEEE Int. Conf. Fuzzy Syst. FUZZ-IEEE, Anchorage, USA, 1998, pp. 1394–1399.
[24] Z. Hou, W. Qian, S. Huang, Q. Hu, W.L. Nowinski, Regularized fuzzy c-means method for brain tissue clustering, Pattern Recognit. Lett. 28 (13) (2007) 1788–1794.
[25] Y. He, M.Y. Hussaini, J. Ma, B. Shafei, G. Steidl, A new fuzzy c-means method with total variation regularization for segmentation of images with noisy and incomplete data, Pattern Recognit. 45 (9) (2012) 3463–3471.
[26] D.L. Pham, Fuzzy clustering with spatial constraints, in: Proc. Int. Conf. Image Process. ICIP, New York, USA, 2002, pp. 65–68.
[27] M.N. Ahmed, S.M. Yamany, N. Mohamed, A.A. Farag, T. Moriarty, A modified fuzzy C-means algorithm for bias field estimation and segmentation of MRI data, IEEE Trans. Med. Imaging 21 (3) (2002) 193–199.
[28] S. Chen, D. Zhang, Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure, IEEE Trans. Syst. Man Cybern. B 34 (4) (2004) 1907–1916.
[29] L. Szilagyi, Z. Benyo, S. Szilagyii, H. Adam, MR brain image segmentation using an enhanced fuzzy C-means algorithm, in: Proc. 25th Annu. Int. Conf. IEEE EMBS, Nov. 2003, pp. 17–21.
[30] W. Cai, S. Chen, D. Zhang, Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation, Pattern Recognit. 40 (3) (2007) 825–838.
[31] F. Zhao, L.C. Jiao, H.Q. Liu, Fuzzy c-means clustering with non-local spatial information for noisy image segmentation, Front. Comput. Sci. China 5 (1) (2011) 45–56.
[32] J.J. Ma, D.Y. Tian, M.G. Gong, L.C. Jiao, Fuzzy clustering with non-local information for image segmentation, Int. J. Mach. Learn. Cybern. 5 (6) (2014) 845–859.
[33] S. Krinidis, V. Chatzis, A robust fuzzy local information C-means clustering algorithm, IEEE Trans. Image Process. 19 (5) (2010) 1328–1337.
[34] N. Li, H. Huo, Y.M. Zhao, X. Chen, T. Fang, A spatial clustering method with edge weighting for image segmentation, IEEE Geosci. Remote Sens. Lett. 10 (5) (2013) 1124–1128.
[35] M. Gong, Z. Zhou, J. Ma, Change detection in synthetic aperture radar images based on image fusion and fuzzy clustering, IEEE Trans. Image Process. 21 (4) (2012) 2141–2151.

[36] M. Gong, Y. Liang, J. Shi, W. Ma, J. Ma, Fuzzy C-means clustering with local information and kernel metric for image segmentation, IEEE Trans. Image Process. 22 (2) (2013) 573–584.
[37] H. Verma, R.K. Agrawal, A. Sharan, An improved intuitionistic fuzzy c-means clustering algorithm incorporating local information for brain image segmentation, Appl. Soft Comput. 46 (2016) 543–557.
[38] Z.X. Ji, Y. Xia, Q. Chen, et al., Fuzzy c-means clustering with weighted image patch for image segmentation, Appl. Soft Comput. 12 (6) (2012) 1659–1667.
[39] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[40] D. Brunet, E. R.Vrscay, Z. Wang, On the mathematical properties of the structural similarity index, IEEE Trans. Image Process. 21 (4) (2012) 1488–1499.
[41] T.C. Chang, S.S.D. Xu, S.F. Su, SSIM-based quality-on-demand energy-saving schemes for OLED displays, IEEE Trans. Syst. Man Cybern. A 46 (5) (2016) 623–635.
[42] H.H. Chang, A.H. Zhuang, D.J. Valentino, et al., Performance measure characterization for evaluating neuroimage sSegmentation algorithms, NeuroImage 47 (1) (2009) 122–135.
[43] F. Masulli, A. Schenone, A fuzzy clustering based segmentation system as support to diagnosis in medical imaging, Artif. Intell. Med. 16 (2) (1999) 129–147.
[44] H. Zhang, J. Fritts, S. Goldman, An entropy-based objective evaluation method for image segmentation, in: Proc. SPIE, Storage Retrieval Methods Appl. Multimedia, vol. 5307, 2004, pp. 38–49.
[45] R.D. Fiete, Comparison of SNR image quality metrics for remote sensing systems, Opt. Eng. 40 (4) (2001) 574–585.
[46] M. Mathworks, Natick, Image Processing Toolbox [Online] Available: http://www.mathworks.com.
[47] C.A. Cocosco, V. Kollokian, R.K.S. Kwan, A.C. Evans, BrainWeb: Online interface to a 3D MRI simulated brain database [online], 2011, Available: http://www.bic.mni.mcgill.ca/brainweb/.
[48] W. Pedrycz, Granular Computing: Analysis and Design of Intelligent Systems, CRC Press/Francis & Taylor, Boca Raton, FL, USA, 2013.
[49] W. Pedrycz, X.M. Wang, Designing fuzzy sets with the use of the parametric principle of justifiable granularity, IEEE Trans. Fuzzy Syst. 24 (2) (2016) 489–496.

Yiming Tang received the Ph.D. degree from Hefei University of Technology in 2011. He is now an Associate Professor in Hefei University of Technology, and his research interests include clustering, image processing, fuzzy system, fuzzy logic, and affective computing.

He has published more than 50 papers in many journals and conferences. He serves an Associate Editor of JOURNAL OF MATHEMATICS AND INFORMATICS, a member of the Editorial Board of Mathematics and Computer Science, and also a senior member of CAAI (Chinese Association for Artificial Intelligence) and a member of IEEE, CCF (China Computer Federation). He serves as a Professional Committee of Multiple-Valued and Fuzzy Logic of CCF, a Professional Committee of Rough Sets and Soft Computing of CAAI, a Professional Committee of Machine Learning of CAAI. He is a reviewer of several journals (including IEEE TRANSACTIONS ON FUZZY SYSTEMS, INFORMATION SCIENCES, ISPRS JOURNAL OF PHOTOGRAMMETRY AND REMOTE SENSING, SIGNAL PROCESSING). He is now a visiting professor at University of Alberta, Canada.

Fuji Ren received the Ph.D. Degree in 1991 from Faculty of Engineering, Hokkaido University, Japan. He is a Professor and the Director of the Faculty of Engineering, the University of Tokushima. His research interests include artificial intelligence, fuzzy system, image processing, natural language processing, and affective computing.

He serves as the Chang Jiang Scholar Endowed Chair Professor of Ministry of Education, the winner of Funds for Overseas Distinguished Young Scientists, the overseas evaluation expert of Chinese Academy of Sciences, and Haizhi experts of China Association for Science and Technology. Moreover, he is the conference founder and Chairman of International Conference on Natural Language Processing and Knowledge Engineering. He is also a member of the CAAI, IEEJ, IPSJ, JSAI, AAMT and a senior member of IEEE, IEICE.

He is the Editor-in-Chief of INTERNATIONAL JOURNAL OF ADVANCED INTELLIGENCE, and a member of the Editorial Board of INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY AND DECISION MAKING, INTERNATIONAL JOURNAL OF INFORMATION ACQUISITION. He is a Fellow of the Japan Federation of Engineering Societies.

Witold Pedrycz received the M.Sc., Ph.D., and D.Sc. degrees from Silesian University of Technology, Gliwice, Poland, in 1977, 1980, and 1984, respectively. He is a Professor and the Canada Research Chair of computational intelligence with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. He is also with the Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland, where he was elected as a foreign member in 2009. He also holds an appointment as Special Professor with the School of Computer Science, University of Nottingham, Nottingham, U.K. His current research interests include computational intelligence, image processing, fuzzy modeling and granular computing, knowledge discovery and data mining, fuzzy control, pattern recognition, knowledge-based neural networks, relational computing, and software engineering. He has authored or coauthored numerous papers. He is also the author of 14 research monographs that cover various aspects of computational intelligence and software engineering.

He is the Editor-in-Chief of INFORMATION SCIENCES. He is also an Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS and APPLIED SOFT COMPUTING. He has edited a number of volumes, including Handbook of Granular Computing (Wiley, 2008). Prof. Pedrycz has been a member of numerous program committees of IEEE conferences in the area of fuzzy sets and neurocomputing. He received the prestigious Norbert Wiener Award from the IEEE Systems, Man, and Cybernetics Council in 2007. He received the IEEE Canada Computer Engineering Medal in 2008 and the Cajastur Prize for Soft Computing in 2009 from the European Centre for Soft Computing for pioneering and multifaceted contributions to granular computing.