ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume **20**, Number 3, 2017, 271–285

An Efficient Particle Swarm Optimization for MRI Fuzzy Segmentation

Moussa SEMCHEDINE and Abdelouahab MOUSSAOUI

Department of Computer Science, University of Ferhat Abbas Setif 1, 19000, Sétif, Algeria,
Email: moussa.semchedine@univ-setif.dz,
abdelouahab.moussaoui@univ-setif.dz

Abstract. In this paper, we propose a novel initialization approach for the *Fuzzy C-Means Algorithm* (FCM) based on *Fuzzy Particle Swarm Optimization* (FPSO) applied to brain MR image segmentation. The proposed method, named FPSOFCM (*Fuzzy Particle Swarm Optimization for FCM*) uses the FPSO algorithm to get the initial cluster centers of FCM according to a new fitness function which combines fuzzy cluster validity indices. The FPSOFCM was evaluated on several MR brain images corrupted by different levels of noise and intensity non-uniformity. Experiment results show the proposed approach improves segmentation results.

Key-words: fuzzy c-means (FCM), MRI segmentation, swarm intelligence, particle swarm optimization (PSO).

1. Introduction

Image segmentation is an important task in image processing and computer vision applications. It can be considered as the first step in image processing and pattern recognition [1]. Image segmentation refers to the process of partitioning an image into many regions of pixels corresponding to different objects or parts of objects according to some homogeneity criteria (e.g. pixel intensity, color,or texture) [2] . Each region is homogeneous and the union of adjacent regions is not homogeneous. Several techniques for image segmentation have been proposed in literature [1, 3]. In general, these techniques are classified into four categories: thresholding, edge based, region growing and clustering techniques [4].

Image segmentation can be considered as a clustering problem where the features characterizing each pixel correspond to a pattern, and each image region corresponds to a cluster [2]. The FCM algorithm, which was first proposed by Dunn [5] and extended as the general FCM clustering algorithm by Bezdek [6], is the most widely used fuzzy clustering method for MR brain images segmentation because of its simplicity and applicability [7–9]. However, the application of the standard FCM algorithm to image segmentation, especially to brain MRI often performs inefficiently since the performance of FCM extremely depends on the initialization of cluster centers, where the random selection in these centers makes the algorithm falling into local optimal solution easily [8, 10]. On the other hand FCM is very sensitive to MR image artifacts [8], such as noise, intensity inhomogeneity, and artifacts due to image acquisition [11].

To overcome the shortcomings of the standard FCM, many works using bio-inspired techniques were proposed such as *Genetic Algorithm* (GA) [12], *Simulated Annealing* (SA) [13], Ant Colony Optimization (ACO) [14] and PSO.

PSO algorithm [15] is a global search strategy which avoids falling into local optimum solution. PSO has been successfully adapted to solve fuzzy clustering problems [16–21] and the problems related to image segmentation [22–28].

Das *et al.* [22] used a modified version of the basic PSO algorithm for the fuzzy image segmentation. The algorithm incorporated spatial information into the membership function. Wensheng et al. [23] proposed a fuzzy PSO approach for solving image clustering problem. In their method, each particle is considered as a candidate cluster center and particles fly in the solution space to search suitable cluster centers. Forghani et al. [11] applied an *Improved Fuzzy C-Means* IFCM and PSO method to find optimal values of two influential factors in segmentation which are the feature difference between neighboring pixels in the image and the relative location of the neighboring pixels. Chun and Fang [24] described a hybridized FCM-PSO approach for image segmentation. FCM algorithm is used to find cluster centres that maximizes a similarity function or minimizes the dissimilarity function and PSO is applied for assigning each pixel to a cluster. In order to solve the problems of the inadequacy of FCM algorithm which is sensitive to initial cluster centers and easily traps into the local optimum, Zhou [10] introduced an image segmentation algorithm based on the predator-prey (PPPSO) and FCM algorithm. Zang and Bo [25] used a fast FCM method together with PSO algorithm which automatically determines the number of clusters as well as the centre of the clusters for image segmentation. Gopal *et al.* [26] presented two phases for the detection of brain tumor in MR images: preprocessing and enhancement in the first phase and segmentation based on FCM with PSO in the second phase. In [27], Yanling and Shen proposed a FCM cluster segmentation algorithm based on PSO algorithm combined with Piecewise Linear Chaotic Map (PWLCPSO). Firstly, the PWLCPSO algorithm is used to get the initial cluster centers. Then, the images are segmented using standard FCM algorithm. Recently, Benaichouche et al. [8] proposed an improved method for image segmentation by acting at three different levels. The first level was related to the FCM algorithm itself by improving the initialization step using PSO algorithm. The second one concerned the classification criterion which was improved by introducing the local information and the Mahalanobis distance. Finally, a postsegmentation stage was used to refine the results, by using a new local criterion. In [28] , the *Fuzzy C-Means* (FCM) algorithm was applied in fusion with *Particle Swarm Optimization* (PSO) to define a new similarity metric based on combining different intensity-based neighborhood features. The algorithm was tested on both synthetic and real noisy images. Experiments show that the method has promising performance in comparison with other state-of-the-art methods.

In most of existing methods, the objective function of FCM given in Eq.1 was generally used as fitness function of PSO algorithm. Therefore, in our method, we present a FPSO algorithm based on a new fitness function which combines fuzzy cluster validity indices. The proposed algorithm takes the result of FPSO as the initialization of the FCM algorithm.

The remaining of this paper is organized as follows. In Section 2, the standard FCM algorithm is introduced with presentation of cluster validity indices used to evaluate the quality of clustering. In Section 3, the basic PSO algorithm and the fuzzy PSO algorithm (FPSO) are presented briefly. The proposed algorithm is described in Section 4. Section 5 reports the experimental results. Finally, conclusion and future works are summarized in section 6.

2. FCM algorithm and cluster validity indices

2.1. Fuzzy c-means algorithm

FCM algorithm and cluster validity indices The FCM algorithm assigns pixels to each category by using fuzzy memberships. Let $O = \{o_1, o_2, \ldots, o_N\}$ denotes an image with *N* pixels to be partitioned into $c(2 \leq c \leq N)$ classes (clusters), where o_i represents the feature value of pixel *i*, the most commonly used feature is the gray-level value. The algorithm is an iterative optimization that minimizes the objective function defined as follows [29]:

$$J = \sum_{k=1}^{c} \sum_{i=1}^{N} u_{ki}^{m} \|o_i - Z_k\|^2 \quad (1)$$

with the following constraints:

$$\forall i \in \{1..N\}, \forall k \in \{1..c\} \quad \sum_{k=1}^{c} u_{ki} = 1; \quad 0 \leq u_{ki} \leq 1; \quad \sum_{i=1}^{N} u_{ki} > 0 \quad (2)$$

where u_{ki} represents the membership of pixel o_i in the *k-th* cluster, Z_k is the *k-th* cluster center. $\|.\|$ denotes the Euclidean distance. The parameter $m(m > 1)$ controls the fuzziness of the resulting partition. The membership functions and cluster centers are updated by Eq.3 and Eq.4 respectively.

$$u_{ki} = \frac{1}{\sum_{l=1}^{c} \left(\frac{\|o_i - Z_k\|}{\|o_i - Z_l\|} \right)^{2/(m-1)}} \quad (3)$$

$$Z_k = \frac{\sum_{i=1}^{N} u_{ki}^{m} o_i}{\sum_{i=1}^{N} u_{ki}^{m}} \quad (4)$$

The FCM algorithm is as follows:

2.2. Cluster validity Indices

To evaluate the quality of a partition provided by FCM algorithm, it is necessary to have cluster validity indices. We describe four well known measures, which are presented as follows:

- The partition coefficient (PC) [30] defined as:

$$PC = \frac{\sum_{i=1}^{N} \sum_{k=1}^{c} u_{ki}^{2}}{N} \quad (5)$$

PC index has maximum value while the cluster partition is the optimal.

Algorithm 1 : FCM

Fix c, m, maximum iterations *itermax* and stop criterion ε
Initialize randomly the cluster centers Z_k, $k = 1, \ldots, c$,
for $t \leftarrow 1$ **to** $itermax$ **do**
  Update the membership function u_{ki}, $k = 1, \ldots, c$; $i = 1, \ldots, N$ according to Eq.3
  Compute the cluster centers Z_k, $k = 1, \ldots, c$, according to Eq. eq4
  Compute the objective function according to Eq. 1
  if $\left| J^{(t)} - J^{(t+1)} \right| < \varepsilon$ **then**
    break
  end if
end for

- The partition entropy (PE) [31] defined as:

$$PE = \frac{-\sum_{i=1}^{N}\sum_{k=1}^{c} u_{ki}\log(u_{ki})}{N} \tag{6}$$

The best clustering is achieved when the value PE is minimal.

- The modified partition coefficient (MPC)

Both PC and PE possess monotonic evolution tendency with c. A modification of the PC index proposed by Dave [32] can decrease the monotonic tendency and it is defined as:

$$MPC = 1 - \frac{c}{c-1}(1 - PC) \tag{7}$$

- The Xie-Beni index (XB) [33] defined as:

$$XB = \frac{\sum_{k=1}^{c}\sum_{i=1}^{N} u_{ki}^2 \|o_i - Z_k\|^2}{N \times \min_{k \neq j} \|Z_k - Z_j\|^2} \tag{8}$$

XB index reaches its minimum value when the partition is the best.

3. Particle swarm optimization (PSO)

The PSO is a population-based stochastic method inspired by bird flocking and fish schooling to find optimal or near-optimal solutions. It was first introduced in 1995 by social-psychologist Eberhart and electrical engineer Kennedy [15]. In very short time the PSO has made the great progress and has been used in many fields of engineering optimization.

3.1. Standard PSO algorithm

The PSO algorithm starts with a population of particles. Each particle *i* consists of potential solutions called positions *X*, and velocities *V* and maintains the following information [34]:

- x_i, the current position of the particle.

- v_i, the current velocity of the particle.
- y_i, the personal best position of the particle (*pbest*); the best position visited so far by the particle.
- $\widehat{y}$, the global best position of the swarm (*gbest*); the best position visited so far by the entire swarm.

In each iteration *t*, the performance of each particle *i* is measured using a predefined fitness function *f*. The personal best position (*pbest*) is obtained as follows [34]:

$$y_i(t+1) = \begin{cases} y_i(t) \text{if} f(x_i(t+1)) \geq f(y_i(t)) \\ x_i(t+1) \text{if} f(x_i(t+1)) < f(y_i(t)) \end{cases} \tag{9}$$

There are two different topologies of PSO algorithm to find best solutions: global and local topologies. In global topology, the position of each particle is affected by the best-fitness particles of the whole swarm in the search space while each particle is influenced by the best-fitness particles of its neighbors in the local topology. In this study we use the global topology whose the global best position $\widehat{y}$ is obtained as follows [34]:

$$\widehat{y}\,(t) \in \{y_1, y_2, \ldots, y_p\} = \min \{f\,(y_1\,(t))\,, f\,(y_2\,(t))\,, \ldots, f\,(y_p\,(t))\} \tag{10}$$

The particle's velocity and position are updated as follows:

$$v_i\,(t+1) = wv_i\,(t) + c_1 r_1\,(t)\,(y_i\,(t) - x_i) + c_2 r_2\,(t)\left(\widehat{y}\,(t) - x_i\,(t)\right); \mathrm{i} = 1, 2, \ldots, p \tag{11}$$

$$x_i\,(t+1) = x_i\,(t) + v_i\,(t+1)\,; \mathrm{i} = 1, 2, \ldots, p \tag{12}$$

where, *w* is inertia weight that controls the impact of previous velocity of particle on its current one, *c1* and *c2* are positive constants, called acceleration coefficients which control the influence of *pbest* and *gbest* on the search process, *p* is the number of particles in the swarm, and *r1* and *r2* are random values in range [0,1]. The PSO algorithm is repeated until a specified number of iterations has been exceeded or the velocity changes are close to zero.

3.2. Fuzzy particle swarm optimization for fuzzy clustering

The FPSO algorithm was initially proposed by Pang *et al.* [35] to solve Traveling Salesman Problem (TSP). In this sub-section, we present the FPSO algorithm for fuzzy clustering problem [19].

In FPSO algorithm, X the position of particle, represents the fuzzy memberships of pixels $\{o_1, o_2, \ldots, o_N\}$,to set of cluster centers $\{Z_1, Z_2, \ldots, Z_c\}$, *X* can be expressed as follows:

$$X = \begin{bmatrix} u_{11} & \cdots & u_{1c} \\ \vdots & \ddots & \vdots \\ u_{N1} & \cdots & u_{Nc} \end{bmatrix} \tag{13}$$

In which u_{ij} is the membership function of the *i*-th pixel to the *j*-th cluster with constraints stated in Eq. 2. Therefore, the position matrix of each particle is similar to the fuzzy matrix *U* in FCM algorithm. Also, the velocity of each particle is stated using a matrix *V* with the size *N* rows and *c* columns. We get Eq. 14 and Eq. 15 for updating the positions and velocities [19].

$$V\left(t+1\right)=\omega\otimes V\left(t\right)\oplus\left(c_{1}r_{1}\right)\otimes\left(y_{i}\left(t\right)\ominus X\left(t\right)\right)\oplus\left(c_{2}r_{2}\right)\otimes\left(\widehat{y}\left(t\right)\ominus X\left(t\right)\right) \tag{14}$$

$$X\left(t+1\right)=X\left(t\right)\oplus V\left(t+1\right) \tag{15}$$

where the symbols $\oplus$ and $\ominus$ denote the addition and subtraction between matrices respectively. The symbol $\otimes$ denote a multiplication of all elements in the matrix by a real number.

After updating the position matrix *X*, it may violate the constraints given in Eq. 2. So it is necessary to normalize the position matrix *X* [19]. First, all negative elements in matrix *X* become zero. If all elements in a row of the matrix *X* are zero, they need to be re-evaluated using series of random numbers within the interval [0,1] and then the matrix undergoes the following transformation without violating the constraints [19]:

$$X_{normal}=\begin{bmatrix} u_{11}\Big/\sum\limits_{j=1}^{c}u_{1j} & \cdots & u_{1c}\Big/\sum\limits_{j=1}^{c}u_{1j} \\ \vdots & \ddots & \vdots \\ u_{N1}\Big/\sum\limits_{j=1}^{c}u_{Nj} & \cdots & u_{Nc}\Big/\sum\limits_{j=1}^{c}u_{Nj} \end{bmatrix} \tag{16}$$

In FPSO algorithm the same as standard PSO, we need a function for evaluating the generalized solutions called fitness function. The FPSO algorithm will be described in detail in next section.

4. Proposed method

In order to overcome the problem of random initialization of FCM, we propose a FCM segmentation algorithm based on FPSO. Firstly, the FPSO algorithm is used to minimize a new fitness function given in Eq.17 to get the initial cluster centers. Then, these centers are used as the initial seed of the FCM. We shall use the notation presented in table 1.

Table 1.: The notation used in this paper.

N	*Number of image pixels*	*itermax*	*Maximum iterations*
c	*Number of clusters*	*p*	*Number of particles*
t	*Iterative counter*	*w*	*Inertia weight*
ε	*Stop criterion*	*Z*	*Cluster centers ($c \times p$ matrix)*
c1,c2	*Acceleration coefficients*	v_{max}	*Maximum velocity*
X	*Particles positions ($N \times c \times p$ matrix)*	*pbest*	*Individual best solution*
V	*Particles velocities ($N \times c \times p$ matrix)*	*gbest*	*Global best solution*

4.1. Fitness definition

In this paper, we propose a new fitness function defined as follows:

$$Fitness = \frac{PE + \alpha}{PC + \beta} \tag{17}$$

where PE is the partition entropy given in Eq. 5, PC is the partition coefficient given in Eq. 6.

α is the sum of distances between each pixel $o_{i,}, i = 1, ..., N$ and cluster centers $Z_{k,}, k = 1, ..., c$ defined as follows:

$$\alpha = \sum_{k=1}^{c} \sum_{i=1}^{N} \|o_i - Z_k\|^2 \tag{18}$$

β is the sum of distances between different clusters defined as follows [36]:

$$\beta = \sum_{\forall k,j,k \neq j} \|Z_k - Z_j\|^2 \tag{19}$$

The proposed algorithm aims to minimize the new fitness function Eq. 17, so the smaller is the fitness function Eq. 17, the better is the clustering result. Note that *Fitness* is minimized when the value of $PE + \alpha$ should be low, and the value of $PC + \beta$ should be high.

4.2. Algorithm description

In the proposed algorithm FPSOFCM, first, p particles are initialized. Each of particle positions represents a matrix *U* containing the membership function u_{ij} of the *i*-th pixel to the *j*-th cluster. In each of iterations, particles are displaced in the problem space with minimizing the fitness function parameterized by α and β parameters using Eq. 17. In order to restrict the particles from moving too far beyond the search space, a technique called velocity clamping [37] is used to limit the maximum velocity of each particle to the range$[-v_{max}, +v_{max}]$. After convergence of FPSO, the cluster centers $\overline{Z}$ corresponding to *gbest* solution which have been found by particles so far is considered as the input of FCM algorithm. Then, FCM is applied to segment the original image.

When FCM algorithm converged, a defuzzyfication process takes place in order to convert the fuzzy partition matrix U to a crisp partition. In this work, the defuzzyfication is performed using the maximum membership procedure which assigns object i to the class c with the highest membership as follows:

$$C_i = \arg_k\{\max(u_{ki})\}; \quad \mathrm{i} = 1, 2, \ldots, N, \quad \mathrm{k} = 1, 2, \ldots, c \tag{20}$$

In this paper, a linear decreasing inertia weight, which was introduced in [38], is used. The value of *w* is linearly decreased from an initial value w_{max} to a final value w_{min} according to the following equation:

$$w\left(t\right) = \frac{itermax - t}{itermax}\left(w_{\max} - w_{\min}\right) + w_{\min} \tag{21}$$

Algorithm 2 : FPSOFCM

input: original image
Fix the parameters c, m, *itermax*, ε , p, *c1*, *c2*, w, v_{max} , .
Create a swarm with p particles: initialize randomly X and V
Initialize randomly cluster centers Z ($c \times p$ matrix)
best-global-fitness $\leftarrow +\infty$
///Step 1: FPSO///
for $t \leftarrow 1$ **to** $itermax$ **do**
  Calculate the cluster centers Z for each particle using Eq. 4
  Calculate the fitness value *Fitness* of each particle according to Eq. 17
  Update the best global fitness *best-global-fitness*
  Calculate *pbest* and *gbest* using Eq. 9 and Eq. 10 respectively
  Update the velocity matrix V for each particle using Eq.14
  Limit the velocity to the range $[-v_{max}, +v_{max}]$
  Update the position matrix X for using Eq. 15
  Normalize the position matrix X using Eq. 16
  if $\left|\textit{best-global-fitnesss}^{(t)} - \textit{best-global-fitnesss}^{(t-1)}\right| < \varepsilon$ **then**
    break
  end if
end for
///Step 2: FCM///
Initialize cluster centers $Z_k \leftarrow \overline{Z}$
for $t \leftarrow 1$ **to** $itermax$ **do**
  Update the membership function $u_{ki}, k = 1, \ldots, c; i = 1, \ldots, N$ according to Eq. 3
  Update the cluster centers $Z_k, k = 1, \ldots, c$ according to Eq.4
  Compute the objective function according to Eq. 1
  if $\left|J^{(t)} - J^{(t+1)}\right| < \varepsilon$ **then**
    break
  end if
end for
return U the membership degrees of each pixel to c clusters
Defuzzyfication of the partition matrix U according to Eq. 20
output: segmented image

5. Experimental results

In this section, we compare the proposed algorithm FPSOFCM with the standard FCM algorithm on a set of simulated MRI brain images downloaded from Brainweb [39]. In order to study the robustness of the proposed algorithm for MRI brain segmentation, the testing images (181x217 pixels) are from two MRI modalities (T1 and PD), corrupted by different levels of white Gaussian noise (0%, 3%, 5%) and intensity non-uniformity (RF)(0%, 20%, 40%).

The FPSOFCM and FCM algorithms were implemented in MATLAB version 7 and run on Pentium(R) 4 CPU 2.66 GHz with 1.24 GB RAM. The study was performed using the following parameters: The number of cluster c is equal to 4: background, gray matter, white matter, cerebrospinal fluid (CSF), m =2, *itermax* =100, ε =10^{-6}, *c1*= *c2*=1.49 same value used in [38], p=20, v_{max} is adjusted to 255 (high grayscale level), w_{max} =0.9 and w_{min} =0.4 [38].

The quantitative results of FCM and FPSOFCM on T1- and PD- weighted images are given in terms of mean values of five indices PC, MPC, PE, XB respectively given in Eq. 5, Eq. 7, Eq. 6 and Eq. 8 . Results after 10 independent runs of simulation are listed in Table 2 and Table 4, and the best values are shown in bold. The standard deviation (indicated between parentheses) for each index is given to check the stability of algorithms. It can be seen in these tables that the PC and MPC values of FPSOFCM are larger than the FCM algorithm in different levels of noise (0%, 3%, 5%) and RF (0%, 20%, 40%) for both MRI modalities (T1, and PD). Meanwhile, the PE and XB values of the proposed algorithm are smallest than FCM algorithm, indicating that the proposed algorithm is capable of generating more compact and well-separated clusters. Additionally, FPSOFCM is more stable than the standard FCM algorithm, shown by low values of standard deviation.

Table 2.: Results of FCM and FPSOFCM algorithms on T1-weighted images.

Noise		0%		3%		5%	
	RF	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM
0%	*PC*	0.886496 (0.012591)	**0.890695 (0.000002)**	0.883778 (0.010506)	**0.887280 (0.000003)**	0.848620 (0.000007)	**0.848622 (0.000007)**
	MPC	0.848662 (0.016787)	**0.854259 (0.000002)**	0.845037 (0.014008)	**0.849707 (0.000004)**	0.798159 (0.000009)	**0.798163 (0.000009)**
	PE	0.221893 (0.025105)	**0.213521 (0.000004)**	0.228919 (0.020322)	**0.222144 (0.000006)**	0.296142 (0.000012)	**0.296138 (0.000012)**
	XB	0.046393 (0.005677)	**0.044498 (0.000003)**	0.047482 (0.004056)	**0.046130 (0.000004)**	0.057496 (0.000005)	**0.057494 (0.000005)**
20%	*PC*	0.861299 (0.015759)	**0.866553 (0.000003)**	0.879581 (0.011243)	**0.883330 (0.000003)**	0.849102 (0.000004)	**0.849110 (0.000001)**
	MPC	0.815066 (0.021012)	**0.822071 (0.000003)**	0.839442 (0.014991)	**0.844440 (0.000004)**	0.798803 (0.000005)	**0.798813 (0.000001)**
	PE	0.269972 (0.031138)	**0.259591 (0.000005)**	0.237008 (0.021933)	**0.229695 (0.000005)**	0.295453 (0.000007)	**0.295438 (0.000001)**
	XB	0.060420 (0.008511)	**0.057582 (0.000004)**	0.047140 (0.006761)	**0.044885 (0.000003)**	0.053753 (0.000003)	**0.053746 (0.000000)**
40%	*PC*	0.874644 (0.011248)	**0.878394 (0.000004)**	0.868202 (0.012215)	**0.872283 (0.000003)**	0. 831642 (0.000004)	**0. 831645 (0.000004)**
	MPC	0.832858 (0.014998)	**0.837859 (0.000005)**	0.824270 (0.016287)	**0.829710 (0.000004)**	0. 775523 (0.000005)	**0. 775527 (0.000005)**
	PE	0.245080 (0.021949)	**0.237761 (0.000008)**	0.256223 (0.022487)	**0.248710 (0.000006)**	0. 328346 (0.000007)	**0. 328341 (0.000007)**
	XB	0.046779 (0.004699)	**0.045211 (0.000004)**	0.052499 (0.001926)	**0.051848 (0.000004)**	0. 056433 (0.000002)	**0. 056432 (0.000002)**

Table 3.: Comparison of algorithm execution times (Time units: seconds) for FCM and the proposed algorithm (FPSOFCM) on T1-weighted images.

Noise	0%		3%		5%	
RF	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM
0%	1. 908594 (1.433156)	**1. 474219** **(0. 178855)**	1. 631250 (0. 697204)	**1. 525781** **(0. 241469)**	1. 550781 (0. 621472)	**1. 521094** **(0. 217085)**
20%	1. 759375 (1. 057437)	**1. 650781** **(0. 641658)**	1. 916406 (0. 941363)	**1. 550000** **(0. 257011)**	1. 862500 (0. 618868)	**1. 810156** **(0. 447276)**
40%	1. 975000 (0. 741017)	**1. 864063** **(0. 566589)**	1. 893750 (0. 410988)	**1. 855469** **(0. 352442)**	1. 660938 (0. 305671)	**1. 628125** **(0. 223585)**

Table 4.: Results of FCM and FPSOFCM algorithms on PD-weighted images.

Noise		0%		3%		5%	
RF		FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM
0%	*PC*	0. 899093 (0. 016681)	**0. 904653** **(0.000001)**	0. 855284 (0.017119)	**0. 860994** **(0. 000006)**	0. 847625 (0. 018340)	**0. 858855** **(0.011231)**
	MPC	0. 865457 (0. 022241)	**0. 872870** **(0.000001)**	0. 807045 (0. 022826)	**0. 814658** **(0. 000008)**	0. 796833 (0. 024453)	**0. 811807** **(0.014975)**
	PE	0. 189601 (0. 029850)	**0. 179653** **(0.000004)**	0. 269484 (0. 030294)	**0. 259378** **(0. 000013)**	0. 287870 (0. 031751)	**0. 268425** **(0.019438)**
	XB	0. 080108 (0. 063840)	**0. 058829** **(0.000000)**	0. 102092 (0. 055044)	**0. 083741** **(0. 000005)**	0. 110595 (0. 045011)	**0. 083088** **(0. 27733)**
20%	*PC*	0. 839769 (0. 029306)	**0. 863698** **(0.000002)**	0. 846118 (0. 020880)	**0. 851337** **(0. 015660)**	0. 844686 (0. 015961)	**0. 851652** **(0. 10449)**
	MPC	0. 786359 (0. 039075)	**0. 818264** **(0.000003)**	0. 794824 (0. 027840)	**0. 801783** **(0. 020880)**	0. 792915 (0. 021281)	**0. 802203** **(0. 13932)**
	PE	0. 296534 (0. 053427)	**0. 252909** **(0.000009)**	0. 286434 (0. 036751)	**0. 277248** **(0. 027563)**	0. 292822 (0. 027663)	**0. 280748** **(0. 18114)**
	XB	0. 147878 (0. 080492)	**0. 082156** **(0.000002)**	0. 100006 (0. 047398)	**0. 088157** **(0. 035551)**	0. 101019 (0. 040312)	**0. 083414** **(0. 26358)**
40%	*PC*	0. 855751 (0. 015633)	**0. 863568** **(0. 000001)**	0. 836529 (0. 013353)	**0. 847433** **(0.000003)**	0. 830740 (0. 000005)	**0. 830743** **(0. 000003)**
	MPC	0. 807668 (0. 020844)	**0. 818091** **(0. 000002)**	0. 782038 (0. 017805)	**0. 796577** **(0.000004)**	0. 774320 (0.000006)	**0. 774324** **(0.000004)**
	PE	0. 272492 (0. 026831)	**0. 259074** **(0. 000006)**	0. 305676 (0. 024243)	**0. 285877** **(0.000009)**	0. 329367 (0. 000009)	**0. 329361** **(0. 000005)**
	XB	0. 083554 (0. 035490)	**0. 065809** **(0.000002)**	0. 099874 (0. 026344)	**0. 078364** **(0.000002)**	0. 056476 (0. 000002)	**0. 056474** **(0. 000001)**

Table 5.: Comparison of algorithm execution times (Time units: seconds) for FCM and the proposed algorithm (FPSOFCM) on PD-weighted images .

Noise	0%		3%		5%	
RF	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM
0%	1.142187 (0.199969)	**1.066406 (0.139569)**	1.229688 (0.270683)	**1.217969 (0.170769)**	1.416406 (0.375399)	**1.276563 (0.137473)**
20%	1. 192188 (0. 222408)	**1. 137500 (0. 140851)**	1. 769531 (0. 309315)	**1. 760938 (0. 256987)**	1. 825000 (0. 239251)	**1. 713281 (0. 221538)**
40%	1. 295313 (0. 307733)	**1. 231250 (0. 159543)**	1. 561719 (0. 740227)	**1. 309375 (0. 151941)**	1. 734375 (0. 177191)	**1. 707031 (0. 159947)**

Table 3 and Table 5 report the mean execution time and the standard deviation taken by FCM and FPSOFCM respectively on T1- and PD- weighted images after 10 independent runs. From these results, we note that the FPSOFCM computation times for all testing images are clearly lower than that for the FCM method. This is because the random selection of cluster centers in FCM algorithm often leads to more iterations, whereas in the proposed method, FCM starts with an improved initial cluster centers requiring fewer iterations. Therefore, both the mean time and the standard deviation in FPSOFCM were reduced. Table 6 provides the comparison of objective function given in Eq.1 after 10 independent runs of simulation. As shown in this table, the value of the objective function of FPSOFCM is smaller than FCM in all tested images.

Table 6.: The comparison of objective function between FCM and FPSOFCM .

Noise		0%		3%		5%	
RF		FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM
0%	*T1*	68.87	**62.36**	70.09	**64.52**	74.18	**70.27**
	PD	56.85	**53.01**	71.71	**69.52**	80.81	**75.56**
20%	*T1*	75.86	**68.77**	71.29	**65.53**	68.870	**62.61**
	PD	75.54	**69.67**	75.55	**72.16**	79.26	**77.92**
	T1	74.33	**68.59**	86.83	**72.32**	76.36	76.36
40%	*PD*	82.21	**81.17**	83.51	**82.81**	86.24	86.24

Fig.1 and Fig.2 present a comparison of segmentation results on simulated MRI brain images with different Gaussian noise levels, respectively, as shown in Fig. 1(a)(d)(g) and Fig. 2(a)(d)(g). The segmentation results obtained by FCM, are shown in Figs. 1(b)(e)(h) and Figs. 2(b)(e)(h) . Figs. 1(c)(f)(i) and Figs. 2(c)(f)(i) show the segmented images provided by FPSOFCM. Visually, the proposed algorithm FPSOFCM achieves a good segmentation effect and provides more detail than FCM algorithm.

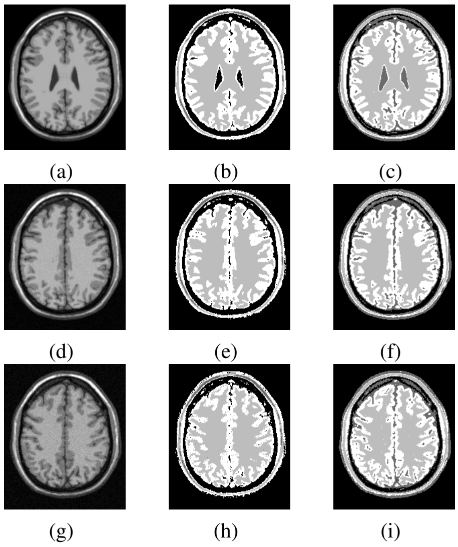

Fig. 1.: Comparison of segmentation results on T1-weighted images (RF=0%). (a) (d) (g) orignal images corrupted by Gaussian noise, respectively (0%, 3%, 5%). (b) (e) (h) FCM result. (c) (f) (i) FPSOFCM result.

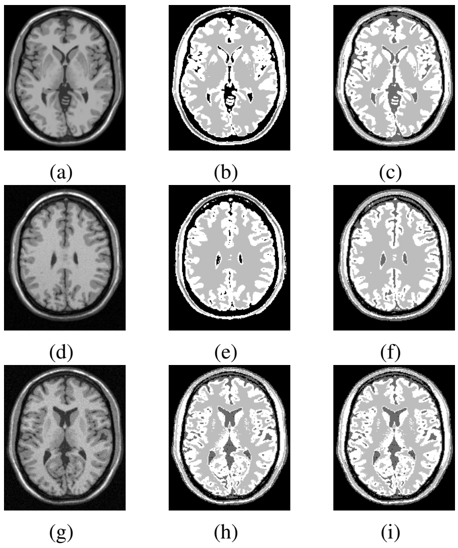

Fig. 2.: Comparison of segmentation results on T1-weighted images (RF=20%). (a) (d) (g) orignal images corrupted by Gaussian noise, respectively (0%, 3%, 5%). (b) (e) (h) FCM result. (c) (f) (i) FPSOFCM result.

6. Conclusion

In this paper we have presented an improvement of the FCM algorithm applied to image segmentation by overcoming its drawbacks. The improvement lies in the initialization step of the FCM algorithm by using the FPSO algorithm based on a new fitness function combined fuzzy cluster validity indices. The result from FPSO was used as the initial seed of the FCM algorithm. Experiment results show that the proposed algorithm FPSOFCM can segment MR images more accurately, and it outperforms the conventional FCM algorithm in terms of CPU time and the solution quality. Our future work will focus on the incorporation of the spatial neighbourhood information into the fitness function to make FPSOFCM more robust against noise.

References

[1] H. D. CHENG, X. H. JIANG, Y. SUN, and Jing Li WANG, *Color image segmentation: Advances and prospects,* Pattern Recognition, 34:2259–2281, 2001.

[2] A. K. JAIN, M. N. MURTY, and P. J. FLYNN, *Data clustering: A review.* ACM Comput. Surv., 31(3):264–323, 1999.

[3] Y. ZHANG A V. DEY and M. ZHONG B, *A review on image segmentation techniques with remote sensing perspective,* In Proceedings of the International Society for Photogrammetry and Remote Sensing Symposium (ISPRS), **38**, pp. 31–42, 2010.

[4] Rafael C. GONZALEZ and Richard E. WOODS, *Digital Image Processing (3rd Edition)*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.

[5] J. C. DUNN, *A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters,* Journal of Cybernetics, 3(3):32–57, 1973.

[6] James C. BEZDEK, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[7] S. R. KANNAN, S. RAMATHILAGAM, R. DEVI, and E. HINES, *Strong fuzzy c-means in medical image data analysis,* J. Syst. Softw., 85(11):2425–2438, November 2012.

[8] A. N. BENAICHOUCHE, H. OULHADJ, and P. IARRY, Improved spatial fuzzy c-means clustering for image segmentation using pso initialization, mahalanobis distance and post-segmentation correction, *Digit. Signal Process.*, 23(5):1390–1400, September 2013.

[9] Maoguo GONG, Yan LIANG, Jiao SHI, Wenping MA, and Jingjing MA, *Fuzzy c-means clustering with local information and kernel metric for image segmentation*, IEEE Transactions on Image Processing, 22(2):573–584, 2013.

[10] ZHOU Xian-cheng. *Image segmentation based on modified particle swarm optimization and fuzzy c-means clustering*, In Second International Conference on Intelligent Computation Technology and Automation, ICICTA, **1**, pp. 611–616, 2009.

[11] N. FORGHANI, M. FOROUZANFAR, and E. FOROUZANFAR, *Mri fuzzy segmentation of brain tissue using ifcm algorithm with particle swarm optimization*, In 22nd international symposium on Computer and information sciences, pp. 1–4, 2007.

[12] M.A. BALAFAR, Abd Rahman RAMLI, M. IQBAL SARIPAN, Rozi MAHMUD, Syahmsiah MASHOHOR, and Hakimeh BALAFAR, *Mri segmentation of medical images using fcm with initialized class centers via genetic algorithm*, In International Symposium on Information Technology, **4**, pp. 1–4, 2008.

[13] Xiao-long LIU, You-sheng ZHANG, and Ying XIE, *Image segmentation algorithm based on simulated annealing and fuzzy c-means clustering*, Journal of Engineering Graphics, 1:89–93, 2007.

[14] P.M. KANADE and L.O. HALL, *Fuzzy ant clustering by centroid positioning*, In *Proceedings of IEEE International Conference on Fuzzy Systems*, **1**, pp. 371–376, 2004.

[15] J. KENNEDY and R. EBERHART, *Particle swarm optimization*, In Proceedings of IEEE International Conference on Neural Networks, **4**, pp. 1942–1948, 1995.

[16] T.A. RUNKLER and C. KATZ, *Fuzzy clustering by particle swarm optimization,* In IEEE International Conference on Fuzzy Systems, pp. 601–608, 2006.

[17] Lili LI, Xiyu LIU, and Mingming XU, *A novel fuzzy clustering based on particle swarm optimization*, In First IEEE International Symposium on Information Technologies and Applications in Education ISITAE '07, pp. 88–90, 2007.

[18] S. SADI-NEZHAD, E. MEHDIZADEH and R. TAVAKKOLI-MOGHADDAM, *Optimization of fuzzy clustering criteria by a hybrid pso and fuzzy c-means clustering algorithm* Iranian Journal of Fuzzy Systems, 5:1–14, 2008.

[19] Hesam IZAKIAN and Ajith ABRAHAM, *Fuzzy c-means and fuzzy swarm for fuzzy clustering problem,* Expert Syst. Appl., 38(3):1835–1838, 2011.

[20] O.A.M. JAFAR and R. SIVAKUMAR, *A study on fuzzy and particle swarm optimization algorithms and their applications to clustering problems,* In IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), pp. 462–466, 2012.

[21] Telmo M. Silva FILHO, Bruno A. PIMENTEL, Renata M.C.R. SOUZA, and Adriano L. I. OLIVEIRA, *Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization,* Expert Syst. Appl., 42(17):6315–6328, 2015.

[22] S. DAS, A. ABRAHAM, and A. KONAR, *Spatial information based image segmentation using a modified particle swarm optimization algorithm,* In Sixth International Conference on Intelligent Systems Design and Applications ISDA, 2, pp. 438–444, 2006.

[23] Wensheng YI, Min YAO, and Zhiwei JIANG, *Fuzzy particle swarm optimization clustering and its application to image clustering,* In Proceedings of the 7th IEEE Pacific Rim Conference on Multimedia, volume 4261 of Lecture Notes in Computer Science, pp. 459–467, 2006.

[24] Wei CHEN and Kangling FANG, *A hybridized clustering approach using particle swarm optimization for image segmentation,* In International Conference on Audio, Language and Image Processing, pp. 1365–1368, 2008.

[25] Zang JING and Li BO, *Image segmentation using fast fuzzy c means based on particle swarm optimization,* In 3rd International Conference on Intelligent Networks and Intelligent Systems (ICINIS), pp. 370–373, 2010.

[26] N.N. GOPAL and M. KARNAN, *Diagnose brain tumor through mri using image processing clustering algorithms such as fuzzy c means along with intelligent optimization techniques,* In IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1–4, 2010.

[27] Yan ling LI and Yi SHEN, *Fuzzy c-means cluster segmentation algorithm based on hybridized particle swarm optimization,* In IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA), pp. 811–815, 2010.

[28] Saeed MIRGHASEMI, Ramesh RAYUDU, and Mengjie ZHANG, *A New Modification of Fuzzy C-Means via Particle Swarm Optimization for Noisy Image Segmentation*, pp. 147–159, Springer International Publishing, Cham, 2016.

[29] Jiayin KANG, Lequan MIN, Qingxian LUAN, Xiao LI, and Jinzhu LIU, *Novel modified fuzzy c-means algorithm with applications*, Digital Signal Processing, 19(2):309 – 319, 2009.

[30] J.C. BEZDEK, *Numerical taxonomy with fuzzy sets,* Journal of Mathematical Biology, 1(1):57–71, 1974.

[31] James C. BEZDEK, *Cluster validity with fuzzy sets,* Journal of Cybernetics, 3(3):58–73, 1973.

[32] Rajesh N. DAVE, *Validating fuzzy partitions obtained through c-shells clustering*, Pattern Recognition Letters, 17(6):613 – 623, 1996.

[33] Xuanli Lisa XIE and G. BENI, *A validity measure for fuzzy clustering*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(8):841–847, 1991.

[34] S.SAATCHI and C. HUNG, *Swarm intelligence and image segmentation, swarm intelligence, focus on ant and particle swarm optimization,* In: I-Tech Education and Publishing, 2007.

[35] Wei PANG, Kang-Ping WANG, Chun-Guang ZHOU, and Long-Jiang DONG, Fuzzy discrete particle swarm optimization for solving traveling salesman problem. In The Fourth International Conference on Computer and Information Technology, pp. 796–800, 2004.

[36] Sitao WU and Tommy W.S. CHOW, *Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density,* Pattern Recognition, 37(2):175–188, 2004.

[37] Russ EBERHART, Pat SIMPSON, and Roy DOBBINS, *Computational Intelligence PC Tools*. Academic Press Professional, Inc., San Diego, CA, USA, 1996.

[38] R.C. EBERHART and Y. SHI, *Comparing inertia weights and constriction factors in particle swarm optimization,* In Proceedings of the 2000 Congress on Evolutionary Computation, **1**, pp. 84–88, 2000.

[39] http://www.bic.mni.mcgill.ca/brainweb/. [Online; accessed 2016].