A Heuristic Solution for Noisy Image Segmentation using Particle Swarm Optimization and Fuzzy Clustering

Saeed Mirghasemi, Ramesh Rayudu and Mengjie Zhang
School of Engineering and Computer Science, Victoria University of Wellington, Wellington, New Zealand

Keywords: Noisy Image Segmentation, Fuzzy C-Means, Particle Swarm Optimization, Impulse Noise.

Abstract: Introducing methods that can work out the problem of noisy image segmentation is necessary for real-world vision problems. This paper proposes a new computational algorithm for segmentation of gray images contaminated with impulse noise. We have used Fuzzy C-Means (FCM) in fusion with Particle Swarm Optimization (PSO) to define a new similarity metric based on combining different intensity-based neighborhood features. PSO as a computational search algorithm, looks for an optimum similarity metric, and FCM as a clustering technique, helps to verify the similarity metric goodness. The proposed method has no parameters to tune, and works adaptively to eliminate impulsive noise. We have tested our algorithm on different synthetic and real images, and provided quantitative evaluation to measure effectiveness. The results show that, the method has promising performance in comparison with other existing methods in cases where images have been corrupted with a high density noise.

1 INTRODUCTION

The concept of partitioning an image into homogeneous regions, usually referred to as *image segmentation*, is an important mid-level image analysis technique for many high-level afterwards applications such as object detection (Zhuang et al., 2012; AntúNez et al., 2013), image recognition (Ferrari et al., 2006; Kang et al., 2011), image retrieval (Mei and Lang, 2014), image compression (Zhang et al., 2014), and video control/surveillance (Mahalingam and Mahalakshmi, 2010; Zhang et al., 2009). Since its emersion in mid 20th century, it has been revolutionized a lot, not only to be applied to more practical problems, but also to cope with the unstoppable trend of demand for more accurate detection, classification, and recognition in a variety of applications. A small but quite practicable section of image segmentation is devoted to noisy image segmentation in which fuzzy clustering usually performs as a powerful tool (Cai et al., 2007). The common fuzzy clustering algorithm for this matter is Fuzzy C-Means (FCM) (Hathaway et al., 2000) which due to simplicity and applicability is one of the most used clustering algorithms. It is also known for better performance in case of poor contrast, overlapping regions, noise and intensity inhomogeneities (Benaichouche et al., 2013). The fuzzy membership of FCM allows each datapoint to belong to every existing cluster with different degrees of membership. This is especially of interest in noisy data clustering where it has been widely used to cluster noisy contaminated data.

Lots of the so far proposed FCM-based algorithms for noisy image segmentation are parameter dependent (Ahmed et al., 1999; Szilagyi et al., 2003; Chen and Zhang, 2004; Cai et al., 2007). Usually, these parameters make a trade-off between preserving the details in an image and eliminating the noise which makes the applicability of these methods limited to noisy images in which the amount of noise is known. This means that the best segmentation results are only obtained when a prior knowledge of the noise volume is available. Another issue is that they usually fail to perform well when the image is presented with heavy impulse noise as a common type of noise mainly because impulse noises are not easy to deal with.

This paper introduces a heuristic fuzzy algorithm for noisy image segmentation with no parameters to tune in advance according to the noise volume. The utilized features have been specifically chosen to compensate for impulse noise, and the algorithm has been designed to confront heavy noise. In this way, we incorporate spatial, texture, and fuzzy membership values to achieve better results.

The rest of this paper is organized as follows. Section 2 is devoted to the research background of this study. Section 3 describes the proposed method. Experimental results, datasets, and parameter settings

Mirghasemi, S., Rayudu, R. and Zhang, M..
A Heuristic Solution for Noisy Image Segmentation using Particle Swarm Optimization and Fuzzy Clustering.
In *Proceedings of the 7th International Joint Conference on Computational Intelligence (IJCCI 2015) - Volume 1: ECTA*, pages 17-27
ISBN: 978-989-758-157-1

are presented in section 4, and section 5 is dedicated to conclusions and future work.

2 BACKGROUND

This section starts with the introduction of the primary FCM and its variants. We then introduce PSO as the heuristic algorithm we have utilized in this paper, and then related work would be presented.

2.1 Fuzzy C-Means Related Algorithms

The Fuzzy C-Means (FCM) as a clustering algorithm was first introduced in (Dunn, 1973), and then extended in (Hathaway et al., 2000). The aim in FCM is to find c partitions via minimizing the following objective function:

$$J = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ji}^{m} d^2(x_i, v_j) \quad (1)$$

where $\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots \boldsymbol{x_N}\}$ is a dataset in which x_i represents a p-dimensional array datapoint in R^p, and N is the number of datapoints (p is the number of features attributed to each datapoint), C is the number of clusters, u_{ij} is the degree of membership of $\boldsymbol{x_i}$ to cluster j, which meets $u_{ij} \in [0,1]$ and $\sum_{i=1}^{C} u_{ij} = 1$, m is the degree of fuzziness, v_j is the prototype of cluster j, and $d^2(x_i, v_j)$ is the distance difference between datapoint $\boldsymbol{x_i}$ and cluster center $\boldsymbol{v_j}$. The two following iterative formulas are necessary but not sufficient for J to be at its local extrema. u and v get updated using these equations till termination threshold is satisfied:

$$v_j^k = \frac{\sum_{i=1}^{N} \left(u_{ji}^k\right)^m x_i}{\sum_{i=1}^{N} \left(u_{ji}^k\right)^m} \quad (2)$$

$$u_{ji}^{k+1} = \frac{1}{\sum_{l=1}^{C} \left(\frac{d_{ji}}{d_{li}}\right)^{2/m-1}} \quad (3)$$

Although FCM itself fails at the segmentation of noisy images, utilizing the basic concept a number of algorithms have been created to cover this failure (Ahmed et al., 1999; Chen and Zhang, 2004; Szilagyi et al., 2003; Cai et al., 2007; Krinidis and Chatzis, 2010). A common approach in this manner (Ahmed et al., 1999) known as FCM_S was introduced in which FCM objective function is modified to deal with intensity inhomogeneities posing in segmentation of magnetic resonance images. The new objective function is as follows:

$$J = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ji}^{m} d^2(x_i, v_j) + \frac{\alpha}{N_R}\sum_{i=1}^{N}\sum_{j=1}^{C} u_{ji}^{m} \sum_{r \in N_k} d^2(x_r, v_j) \quad (4)$$

where N_k is the set of neighbors within the neighboring window around x_i, N_R is its cardinality, and x_r represents the neighbor of x_i. How the neighbors are affecting the objective function is controlled by α. The new updating formulas for u and v are obtained according to Lagrange multipliers taking partial derivatives of the new objective function. The new objective function allows the neighboring pixels to affect the labeling procedure of a pixel. Then two modifications of FCM_S algorithm were proposed in (Chen and Zhang, 2004) mainly trying to reduce the computation of FCM_S. These two algorithms known as FCM_S1 and FCM_S2 use a pre-calculated mean and median-filtered image of the noisy image as a substitution for neighbor pixel labeling in each iteration which results in a faster algorithm. The modified objective function is:

$$J = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ji}^{m} d^2(x_i, v_j) + \alpha \sum_{j=1}^{C}\sum_{r \in N_k} u_{jr}^{m} d^2(\overline{x_r}, v_j) \quad (5)$$

where $\overline{x_r}$ is the mean or median average of the neighboring pixels around x_i.

For even a faster performance, EnFCM was proposed (Szilagyi et al., 2003). Here again, a linearly-weighted local filter is applied to image in advance according to:

$$\xi_i = \frac{1}{1+\alpha}\left(x_i + \frac{\alpha}{N_R}\sum_{r \in N_k} x_r\right) \quad (6)$$

in which ξ_i is the gray level value of the pixel i, and α plays the same role as before. Then the clustering procedure is performed on the gray-level histogram obtained from the filtered image. As there are only 256 gray levels in an image, and having in mind that the number of pixels in an image are generally much larger than 256, this algorithm is quite fast, and also has better performance in noisy image segmentation compared to FCM_S. The new objective function is introduced as:

$$J = \sum_{i=1}^{Q}\sum_{j=1}^{C} \gamma_i u_{ji}^{m} d^2(\xi_i, v_j) \quad (7)$$

where Q denotes the number of gray levels, and ξ_i is the number of pixels having a gray value equal to i.

The so far mentioned algorithms, although having great achievements dealing with noise, they all suffer from a common problem which is the parameter

α. This parameter keeps a trade-off between the volume of noise and the details of an image. In other words, it should be large enough to compensate for noise and should be small enough to preserve details of an image like edges. That is why the performance of these algorithms is α-dependent which is a disadvantage when dealing with generic image segmentation. To make up for this, FGFCM was proposed in (Cai et al., 2007) incorporating local spatial and gray information. A new filtering factor, S_{ij} is proposed which acts as a local similarity measure:

$$S_{ij} = \begin{cases} e^{-max(|p_i-p_j|,|q_i-q_j|/\lambda_s)-||x_i-x_j||^2/\lambda_g\sigma_i^2}, & \text{i} \neq \text{j} \\ 0 & \text{i} = \text{j} \end{cases} \quad (8)$$

$$\sigma_i = \sqrt{\frac{\sum\limits_{j \in N_k} \| x_i - x_j \|^2}{N_R}} \quad (9)$$

where i stands for the pixels in the center of the sliding window, and j is one of the neighboring pixels falling in the neighborhood window, and (p_k, q_k) is the coordinates of the pixel k in the neighboring window. λ_s and λ_g are parameters with functioning similar to α. Like EnFCM, FGFCM performs clustering on the basis of histogram information obtained from the proposed filtering factor. Although FGFCM acts less parameter-independent introducing λ_s and λ_g, again its performance is influenced by the variation of λ_g (Krinidis and Chatzis, 2010).

Regardless of the fact that all these methods perform reasonably well on noisy image segmentation, they have parameters that need to be tuned according to the volume and type of noise. Although λ_s could be fixed in 3 according to (Cai et al., 2007), α and λ_g have to be adjusted empirically. They have to be large to make up for the noise, and have to be small to keep the details of an image. This makes the applicability of them limited to images with prior information about the noise and its volume. Therefore, the need for parameter-free algorithms which can adaptively be used for noisy image segmentation is essential. In addition, the existing methods fail to produce accurate segmentation results when the noise is heavy, which is another motivation for the proposed method in this paper.

2.2 Particle Swarm Optimization

Particle Swarm Optimization is a computational optimization algorithm introduced in (Eberhart and Kennedy, 1995; Kennedy and Eberhart, 1995). Due to efficiency, robustness, and simplicity (Engelbrecht, 2007) the technique has been modified many times, for general and specific applications. The search algorithm is motivated by the social behaviors of organisms. Particularly, choreography of birds flock led to the design of PSO. The algorithm is initialized with a swarm of potential solutions in a multidimensional space. Each solution, also known as particle, has the ability to move. Therefore, particle i has two parameters as x and v which specify its location and speed in the search space, respectively. During the movement, each particle updates its position and velocity according to its own experience, and that of its neighbors. i is in interactive communication with neighboring particles in order to find the best position (final solution). The best so-far position of each particle is called *pbest*, and the best so-far position in the whole swarm is called *gbest*. What really determines the goodness of *pbest*, *gbest*, and basically all particles is a fitness function which is an essential part of PSO algorithm. The fitness function specifies the nature of the optimization problem, and is designed according to the application. Briefly, assuming a D-dimensional search space the ith particle is represented by $\boldsymbol{X_i = (x_{i1}, x_{i2} \ldots x_{iD})}$ and $\boldsymbol{V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})}$ as D-dimensional arrays for the positions and velocities. $\boldsymbol{x}$ and $\boldsymbol{v}$ are updated using these two equations:

$$v_{id}^{k+1} = w \times v_{id}^k + c_1 r_1 (pbest_d - x_d^k) + c_2 r_2 (gbest_d - x_d^k) \quad (10)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (11)$$

where $d = 1, 2, \ldots, D$, $i = 1, 2, \ldots, N$, are the sizes of dimension and swarm, c_1 and c_2 are positive constants, r_1 and r_2 are random numbers, uniformly distributed in the interval [0,1], $k = 1, 2, \ldots$, denotes the iteration number, $pbest_d$ and $gbest_d$ represent *pbest* and *gbest* in the dth dimension, and w is inertia weight which controls the influence of previous velocities on the new velocity. Larger inertia weights indicate larger exploration through the search space while smaller values of the inertia weight restrict the search on a smaller space (Engelbrecht, 2007). Typically, PSO starts with a larger w, and the decreases gradually over the iterations. We have adopted the following equation for w to simulate its descending property:

$$w = (w_{initial} - w_{final}) \times \frac{(k_{max} - k)}{k_{max}} + w_{final} \quad (12)$$

where $w_{initial}$, is the preliminary value of w, w_{final} is the final value of w, k is the iteration number, and k_{max} is the maximum number of iterations.

2.3 Related Work

A couple of research works have been proposed in an attempt to combine heuristic algorithms particularly PSO with FCM. The common approach (Zhang et al., 2012; Benaichouche et al., 2013; Tran et al., 2014) is that potential solutions (particles) are possible values for cluster centers within the intensity diversity of pixels. They take the objective function of a FCM-based clustering method as the fitness function, and then try to find the optimum positions of the cluster centers that minimize the objective function the most. This will omit the updating formula for cluster centers, but again, they need the fuzzy membership updating formula to obtain the value of membership. That is why these approaches do not really create a new algorithm, rather they just try to optimize an existing one. Knowing that, the FCM-based clustering methods are already computationally optimized, these approaches fall effective in special cases that the objective function of an already existing FCM-based algorithm is not simple enough to be completely optimized by its iterative procedure itself. Since the improvement is not significant often for such an optimization case, the better performance is achieved by including other criteria to the existing algorithm (Tran et al., 2014; Benaichouche et al., 2013).

Another trend in the literature is investigating the effect of Gaussian noise in noisy image segmentation. This paper, unlike the common trend, is focused on impulse noises as another common type of noise in image processing. Impulse or fat-tail distributed noise, which sometimes is referred to as *salt and pepper* noise, can be produced by malfunctioning pixels in camera sensors, faulty memory locations in hardware, analog-to-digital converter errors or bit errors in a transmission (Bovik, 2005). This means, images are usually damaged by impulsive noises during acquisition or transmission. It appears as sparsely occurring white and black pixels. Since the corrupted pixel by impulsive noise contains no information about the present image, impulse noisy image segmentation is a challenging issue.

The proposed approach in this paper, unlike the existing approaches, uses PSO to define a new similarity criterion for FCM in which segregation between two datapoints (pixels here) happens by combining different features extracted from a neighboring window around each pixel. For this, the new measurement criterion not only utilizes gray and spatial information, but also uses the fuzzy membership value to achieve a better performance. Our method introduces a new optimization process in which FCM clustering performance on noisy image segmentation gets improved by modifying the classic Euclidean metric. This is different from the common trend that uses PSO for a better initialization of FCM.

3 THE PROPOSED METHOD

Fig. 1 shows a block diagram of the proposed method. This figure shows three main steps in the algorithm: initialization and pre-processing, PSO search procedure, and final clustering and segmentation. The general idea is to use PSO to form a new similarity criterion based on simple texture properties of a local neighboring window. During the iterations of the PSO search procedure, FCM will be used to obtain the parameters related to similarity measure, and to cluster the noisy data based on the new similarity criterion. Simply saying, PSO along with FCM, creates a search space in which the parameters related to the new similarity measure will be gradually and automatically optimized.

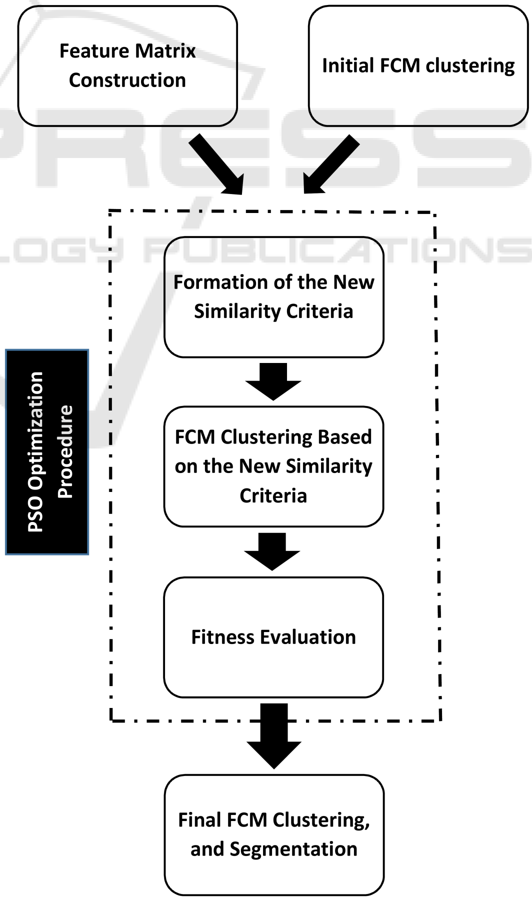

Figure 1: Block diagram of the proposed method.

3.1 Pre-processing and Initialization

The first step of the proposed algorithm deals with some initializations and feature construction. The initialization is carried out via a standard FCM according to (2) and (3). This initial clustering provides the initial cluster centers, V, and the initial membership values, U, for the PSO procedure in which the new similarity measurement is formed. This preprocessing not only leads to better segmentation results at the end, but also builds a deterministic algorithm with stable outputs.

Feature construction is done in advance for more simple and efficient implementation. Simple texture information of a local window around each pixel will be used to construct the feature matrix. The texture information is captured by four statistical-based filters which apply to the intensity values of the pixels inside the neighboring window. These sliding-window filters employ median, variance, standard deviation, and Wiener filtering.

The Wiener filter (Wiener, 1964) low-pass filters a grayscale image that has been degraded by additive noise. The filter uses a pixel-wise adaptive Wiener method based on statistics estimated from a local neighborhood of each pixel. The Wiener filtering is a general way of finding the best reconstruction of a noisy signal. More specially, it can generally be applied whenever you have a basis-in-function space that concentrates "mostly signal" in some components relative to "mostly noise" in others. It could be applied on spatial or wavelet function basis. In spatial (pixel) basis (as utilized in this paper), the Wiener filter is usually applied to the difference between an image and its smoothed version. It gives the optimal way of tapering off the noisy components, so as to give the best (L^2 norm) reconstruction of the original signal. Assuming an observation (noisy image) (C_i), which is composed of the original image (S_i) and noise (N_i), we have the measured components as:

$$C_i = S_i + N_i \tag{13}$$

The filter looks for a signal estimator that scales the individual components of what is measured as:

$$\hat{S}_i = C_i \Phi_i \tag{14}$$

and finds the signal estimator, Φ_i, such that it minimizes $\langle |\hat{S} - S|^2 \rangle$. Knowing that we are working on some orthogonal basis, the L^2 norm is just the sum of squares of the components of what is measured, we extend the latter, differentiate with respect to Φ_i, and set it to zero to obtain:

$$\Phi_{is} = \frac{\langle S_i^2 \rangle}{\langle S_i^2 \rangle + \langle N_i^2 \rangle} \tag{15}$$

where $\langle S_i^2 \rangle$ and $\langle N_i^2 \rangle$ are estimations of signal and noise power in each component. Using the estimator introduced in (Lim, 1990), Wiener estimates the local mean (ρ), and variance (σ^2) around each pixel:

$$\begin{aligned} \rho &= \tfrac{1}{MN} \sum_{n_1, n_2 \in \eta} a(n_1, n_2) \\ \sigma^2 &= \tfrac{1}{MN} \sum_{n_1, n_2 \in \eta} a^2(n_1, n_2) - \rho^2 \end{aligned} \tag{16}$$

where η is the $N \times M$ local neighborhood of each pixel in the image, and n_1 and n_2 are the coordinates of pixel a. Then, a pixel-wise Wiener filter using these estimates is created:

$$b(n_1, n_2) = \rho + \frac{\sigma^2 - v^2}{\sigma^2}(a(n_1, n_2) - \rho) \tag{17}$$

where v^2 is the noise variance related to the average of all the local estimated variances.

Although some of the filters for constructing the feature matrix have been individually used for FCM-based image segmentation (Chen and Zhang, 2004), this is the first time that their combination is used for noisy image segmentation. To be able to combine noise degradation properties of each filter on an image, we use them all to build a new similarity criterion. This not only allows us to benefit from the properties of each filter, but also gives our approach the ability to adaptively come up with a unique solution for each image.

3.2 A New Similarity Measure

The main contribution of this paper is that it incorporates simple statistical features of a neighboring window around each pixel into the distance calculation metric of the classical FCM using PSO. Motivated by the texture detection algorithm in (Tian et al., 2013), we modify $d^2(x_i, v_j)$ in (1) as below:

$$d^2(x_i, v_j) = \| x_i - v_j \| \, (1 - \sum_{p=1}^{4} \mu_p F_{ij(p)}) \tag{18}$$

where $\| x_i - v_j \|$ is the Euclidean distance between the gray (intensity) information of each pixel and jth cluster center, p is the number of features, and $\boldsymbol{\mu_p}$ are the coefficients to be obtained subjected to:

$$0 < \mu_p < 1 \tag{19a}$$

$$\sum_{p=1}^{4} \mu_p F_{ij(p)} \leq 1 \tag{19b}$$

F_{ij} is a fusion of one-at-a-time (one out of four) feature attributed to each pixel, and some other information from the rest of pixels falling inside the neighboring window as:

$$F_{ij} = \frac{\sum_{n=1}^{N_R} u_n \times (f_i^n - f_i) \times (e^{-|x_i^n - v_j|})}{\sum_{n=1}^{N_R} (f_i^n - f_i) \times (e^{-|x_i^n - v_j|})} \tag{20}$$

where n is the index for the pixels within the neighboring window for the similarity metric (N_R is the cardinality of neighboring pixels), u is the fuzzy membership value, f is the considering feature from the previously-built feature matrix, f_i is the corresponding feature value of the pixel x_i at the center of the neighboring window for the similarity metric, x is the intensity values, and v_j is the cluster center for cluster j. μ coefficients are determined and optimized gradually during the PSO procedure to produce a proper similarity measure that can mitigate the influence of noise to the greatest extend.

Equation (18) tries to reduce the distance between a pixel and a cluster center with respect to the information obtained from the neighboring window around the pixel. This is not limited to only one feature though, as (18) indicates. This is due to the requirement that one feature alone may not be able to attenuate the effect of noise. Having different features included in the new similarity measure, the search procedure provided by PSO enables this reduction to take the best out of each feature in an optimizing manner.

In addition, we have no parameters which need tuning according to the noise volume. In our method, all the related parameters (except for the neighboring window sizes) in the similarity criterion get tuned *automatically* during the iterations of PSO procedure according to the noise properties of the test image.

3.3 PSO Representation

Apart from general motivations to utilize PSO for optimization problems (mentioned in sub-section 2.2), simplicity of representing our optimizing problem in the form of a PSO-based process is another motivation. The particle representation of PSO suits well our objectives toward obtaining the optimum weights for the new similarity measure. Also, the encoding and decoding procedure is quite straightforward in our problem. Overall, PSO is utilizes to adaptively tune all the parameters associated with the new similarity measure specifically for each image, based on the volume of noise and feature properties.

The previously constructed feature matrix, initial fuzzy membership values and cluster centers are used as inputs for the PSO optimization procedure. As (18) and (20) indicate, the new similarity criterion is designed using four features. Associated with each feature is a μ coefficient. PSO is applied to find the best contribution of each feature by determining μ values. Therefore, each particle, $\boldsymbol{p_i}$, contains potential values for μ coefficients in form of $\boldsymbol{p_i = (\mu_1, \mu_2, \mu_3, \mu_4)}$ that demonstrates a 4-D search space.

The search space, as (19a) suggests, is defined as the smooth interval (0,1) for each dimension, and is restricted continuously according to (19b). During the PSO search, for each proposed combination of μ values, FCM clustering is performed using (2) and (3), and then the PSO fitness function is calculated to evaluate the combination. The fitness function (1) in which $d^2(x_i, v_j)$ has been substituted by the new similarity measure as in (18). The best $\boldsymbol{p_i}$ is the particle that minimizes the fitness function the most. The fitness function, which basically drives the search algorithm, conveys two important values to the next iteration:

1. The best solution that results in the minimum value of the fitness function.
2. The cluster centers that correspond to that solution.

When the PSO search finishes, the final combination of μ values is used for one final clustering. Then, the pixels will be labeled according to their biggest membership value to create a segmented image.

One disadvantage of FCM is that it easily gets trapped in local minima and fails to achieve the optimum results. To overcome this, the initial four dimensional array is set to very small positive values. The inertia factor, w, as a factor to control particle velocity during the search, has bigger values in the beginning and smaller values towards the end of the search. As mentioned in (Mirghasemi et al., 2012), selecting values between [1,0.5] with the mentioned updating formula leads to maximum velocity compatibility.

3.4 Summary of the Algorithm

Different steps of the proposed method could be summarized as below:

1. Load the noisy image.
2. Build the feature matrix based on Wiener, median, variance, and standard deviation measures of intensity values in the neighboring window.

3. Set the parameters for FCM clustering, including the number of clusters.
4. Perform a standard FCM clustering to obtain initial values for U and V.
5. Initialize PSO with parameters for x, v, iteration number, particle number, search space dimension size, and inertia weight.
6. Propose an initial solution for $\boldsymbol{\mu}$ values.
7. Form the new similarity measure according to (18).
8. Perform FCM clustering according to the new similarity measurement.
9. Evaluate the fitness function: calculate the objective function of FCM according to (1).
10. Form ***pbest*** and ***gbest*** according to the fitness value from step 9.
11. Update x and v according to (11) and (10).
12. Go back to step 6, unless it is the end of iterations.
13. Use the resultant optimized values of $(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\mu}_3, \boldsymbol{\mu}_4)$ to do the final similarity measure calculation and FCM clustering.
14. Use the U matrix for image segmentation.

For a clearer depiction of the proposed algorithm, the pseudocode is provided for the PSO search procedure in Algorithm 1. The parameters used in this pseudocode are as follow: ps, X, V, $Pbest$, C, k, k_{max}, and w stand for population size, population's position matrix, population's velocity matrix, the population's *pbest* matrix, matrix of population's corresponding cluster centres, iteration, iteration number, and inertia weight, respectively.

3.5 Parameter Design

Both PSO and FCM algorithms have intrinsic parameters to set. Also, our method has measures for the sizes of the local neighboring windows in both filtering and new distance forming steps. This sub-section provides details for all these parameters. As mentioned before, parameters related to the new similarity criterion get tuned automatically, and do not need prior setting. Based on experiments on two different datasets of gray images, we came up with the parameter adjustment shown in table 1. These parameters are fixed for every test image of each dataset, and none of them need to be changed.

Algorithm 1: The PSO search steps.

```
Set the PSO parameters: x, v, k, k_max, and w;        ▷ x is initially [0.001,0.001,0.001,0.001]
Set particle one as gbest;
k = 0;
while k < k_max do
    Update w using(12);
    for each particle i ∈ pi do
        Form the new similarity metric according to (18);
        Perform FCM based on the new similarity metric;
        Evaluate f(x) according to (1);
    end for
    if k = 0 then
        pbest = f(x);
        gbest = min(Pbest);
        c = C(gbest);
    else
        pbest = f(x) < pbest;
        gbest = min(Pbest);
        c = C(gbest);
        Update V using (10) and restrict its growth;
        Update X using (11) and restrict its growth;
        k = k + 1;
    end if
end while
Return gbest and c;
```

Table 1: Parameter Setting.

Parameter	Value
Neighboring window for filtering	5×5
Neighboring window for similarity criterion	5×5
Weighting exponent (m)	2
Termination threshold for FCM	0.001
Maximum number of iterations for FCM	100
Population size (ps)	20
Iteration number	50
The initial value of the first solution	0.001
c_1 and c_2 in (10)	1
$w_{initial}$ and w_{final} in (12)	1 and 0.5

4 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we compare the robustness of our method in heavy impulse noisy image segmentation against four other methods. The first one is the hard clustering method K-means, and the other three are fuzzy clustering methods named as FCM (Hathaway et al., 2000), EnFCM (Szilagyi et al., 2003), and FGFCM (Cai et al., 2007). EnFCM needs tuning for α, and FGFCM needs tuning for λ_g according to the type and volume of noise. We take $\alpha = 1.8$, $\lambda_s = 3$, and $\lambda_g = 6$ by investigating the interval [0.5,6] for λ_g,

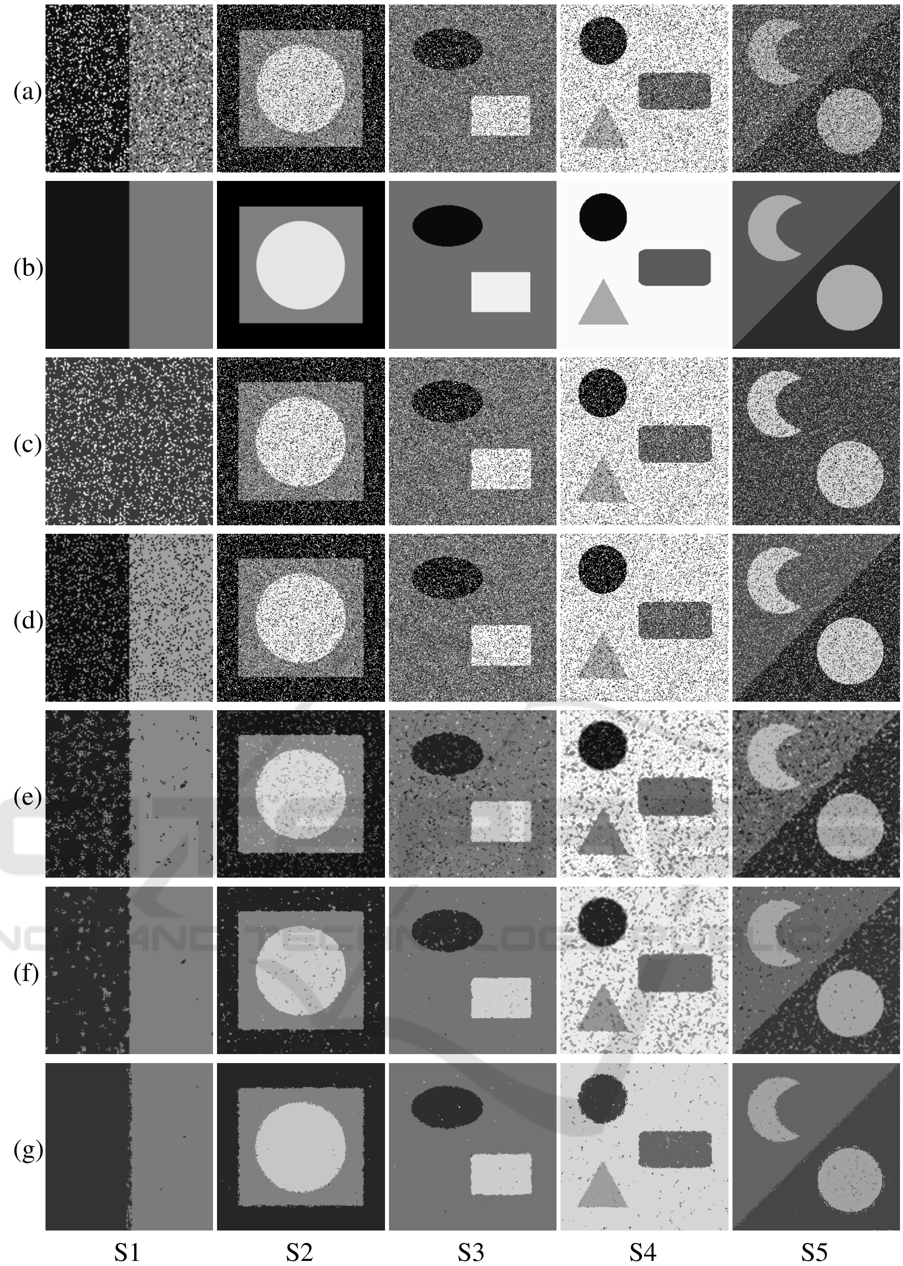

Figure 2: The segmentation results of the proposed algorithm on some sample synthetic images. Rows (a) through (g) are the noisy corrupted images, the ground truths, K-means, FCM, EnFCM, FGFCM, and our methods segmentation results, respectively.

and the interval [0.2,8] for α which, according to our experiments comes with the best performance of EnFCM and FGFCM methods. To carry out quantitative evaluation, we choose the Segmentation Accuracy (SA) introduced in (Ahmed et al., 1999):

$$SA = \sum_{i=1}^{C} \frac{A_i \cap S_i}{\sum_{j=1}^{C} S_j} \tag{21}$$

in which A_i represents the number of segmented pixels from the ith cluster and, S_i is the number of pixels belonging to the cluster i in the ground truth image.

To evaluate our method from different perspectives, we have utilized two different datasets. The first dataset is composed of synthetic images in which the tested images are 256×256 pixels, except for the image S1 in Fig. 2 which is 128×128 pixels. The

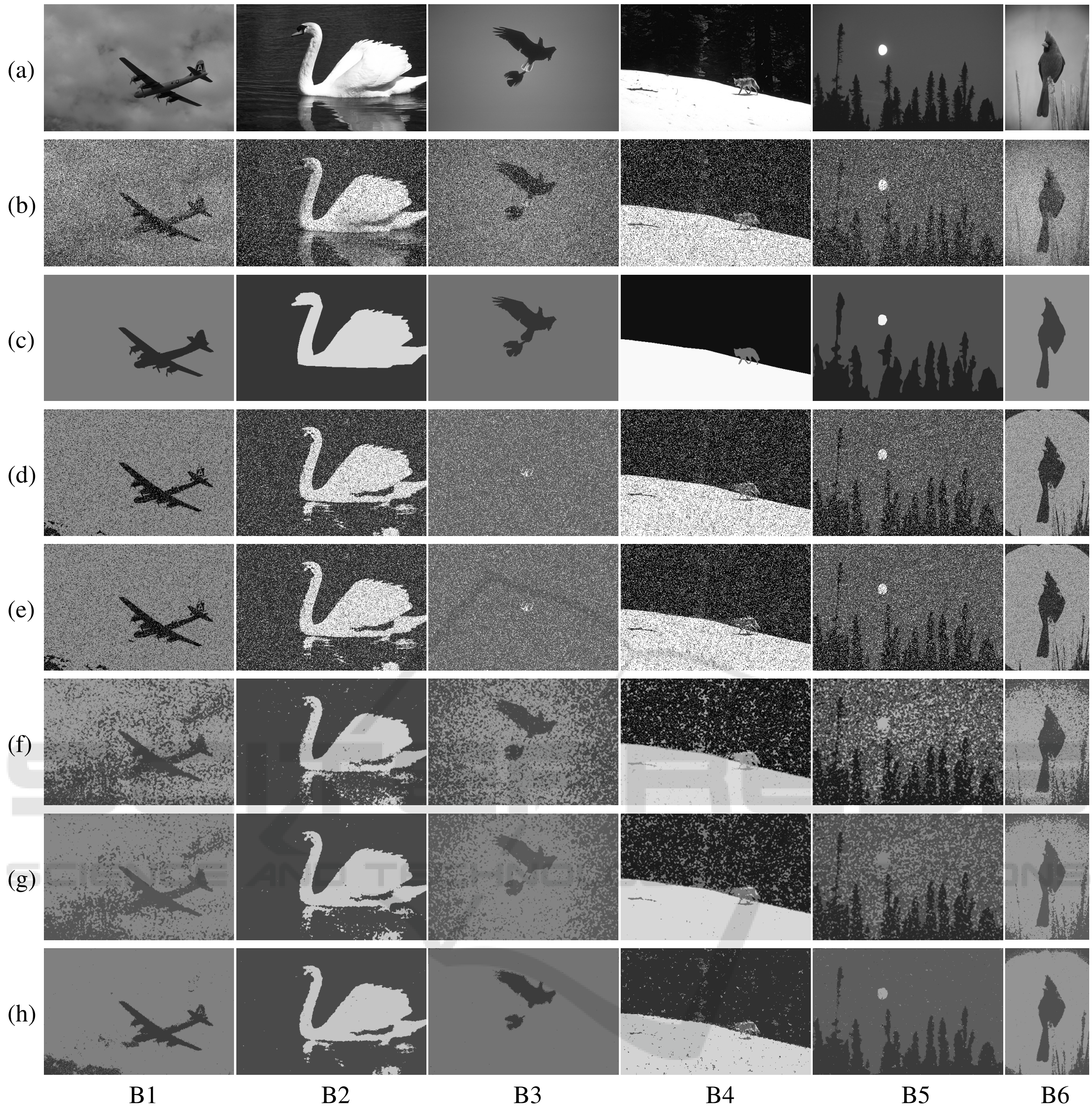

Figure 3: Segmentation results on Berkeley dataset. Rows (a) through (h) are the original images, the noisy corrupted images, the ground truths, K-means, FCM, EnFCM, FGFCM, and our methods segmentation results, respectively.

second database is the Berkeley image segmentation (Martin et al., 2001) which is composed of two sets of images, namely BSDS300 and BSDS500. These datasets are specifically created for image segmentation and boundary detection, providing ground truth for each image. Here, the images are 481×321 pixels.

Salt and pepper noise with heavy density of 30% has been applied through all the testing procedure. The time performance of our method varies depending on the image size and its complexity. FCM needs various numbers of iterations for different images to satisfy the specified termination threshold. On average, for images of 256×256, on a Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz machine with 8GB of RAM, it takes 25 minutes for an image to get processed.

Fig. 2 shows the segmentation results of some sample synthetic noisy images where the proposed method performs better in the segmentation of important regions. The numbers of specified clusters are 2, 3, 3, 4, and 3 for S1 through S5 images, respectively. Table 2 also shows the SA metric evaluation of the re-

Table 3: Quantitative comparison for Fig. 3, according to the SA metric. Bold numbers indicate the best performance for each image.

Algorithm	**B1**	**B2**	**B3**	**B4**	**B5**	**B6**
K-means	85.3580	86.8103	19.6629	82.9203	70.9614	79.1877
FCM	85.0750	86.8103	19.6629	81.7412	70.9614	79.1877
EnFCM	67.2460	96.5247	58.7644	84.7205	65.4835	73.2668
FGFCM	79.4635	97.3976	66.8882	85.1418	67.2698	79.4748
Our method	**96.3309**	**97.8440**	**98.8243**	**97.3401**	**95.2972**	**88.4101**

Table 2: Quantitative comparison for Fig. 2, according to the SA metric. Bold numbers indicate the best performance for each image.

Algorithm	**S1**	**S2**	**S3**	**S4**	**S5**
K-means	50.0585	84.8183	72.6301	82.7271	52.3180
FCM	85.0676	84.8183	72.6301	82.7271	78.6958
EnFCM	93.0931	95.3562	92.0351	75.4330	79.8469
FGFCM	97.3660	98.2033	99.1229	81.2957	89.4574
Our method	**99.3301**	**98.9701**	**99.2961**	**98.2316**	**97.0128**

sults in Fig. 2 on all five methods with bold numbers representing the best performance for each test image. This table shows that the proposed method performs significantly better than K-means, FCM, and EnFCM. When it comes to FGFCM, our method still performs visibly better on S1, S2, S4, and S5. To see the segmentation difference of FGFCM and the proposed method on image S3, one might need to have a closer look to see the better performance of our method. Qualitative evaluation also confirms better segmentation results obtained by our method. Only FGFCM has close performance to our method specially on image S3.

Testing our method with the second database comes with the segmentation results shown in Fig. 3. Here, six sample images named as B1-B6 are selected. The number of clusters has been set to two for the fuzzy clustering part in all of them except for B4 and B5 in which the number of clusters is three. Again, the proposed method performs both qualitatively and quantitatively better than the other four methods. Although FGFCM has somewhat comparable segmentation results with our method on synthetic images, Fig. 3 shows that the performance difference of the proposed method in real images is even greater. Our method performs better in the segmentation of the most compact regions. Table 3 provides the SA metric evaluation of the segmented images shown in Fig. 3. According to this table, the method that has close performance to our method is not merely FGFCM. Surprisingly, FCM has the second best performance in B5, and also a close performance to FGFCM in B6.

5 CONCLUSIONS AND FUTURE WORK

A noisy image segmentation method was proposed using PSO and FCM. The main objectives were to introduce a new algorithm that is parameter free, has good performance on impulse noises, and can deal with heavy density noise. In this way, modifying the traditional Euclidean similarity measure in FCM using different intensity-based features extracted from a neighboring window around each pixel was the main objective. PSO was utilized to produce an optimum combination of these features, and FCM was used to deal with the clustering problem. Spatial, intensity, and fuzzy criteria are considered in the new similarity metric simultaneously for better performance. The proposed method introduces a new algorithm to combine different features extracted from the local neighboring window. This puts forward a new way to fuse features of different types. A future work in this manner is to use more effective features extracted form the neighboring window. Features than can extend the applicability of the proposed method to other types of noises as well. The qualitative and quantitative evaluation showed better performances compared with a few state-of-the-art methods.

REFERENCES

Ahmed, M., Yamany, S., Mohamed, N., and Farag, A. (1999). A Modified Fuzzy C-Means Algorithm for MRI Bias Field Estimation and Adaptive Segmentation. In Taylor, C. and Colchester, A., editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI'99*, volume 1679 of *Lecture Notes in Computer Science*, pages 72–81. Springer Berlin Heidelberg.

AntúNez, E., Marfil, R., Bandera, J. P., and Bandera, A. (2013). Part-based object detection into a hierarchy of image segmentations combining color and topology. *Pattern Recogn. Lett.*, 34(7):744–753.

Benaichouche, A., Oulhadj, H., and Siarry, P. (2013). Improved spatial fuzzy c-means clustering for image segmentation using {PSO} initialization, mahalanobis distance and post-segmentation correction. *Digital Signal Processing*, 23(5):1390 – 1400.

Bovik, A. C. (2005). *Handbook of Image and Video Processing (Communications, Networking and Multimedia)*. Academic Press, Inc., Orlando, FL, USA.

Cai, W., Chen, S., and Zhang, D. (2007). Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. *Pattern Recognition*, 40(3):825–838.

Chen, S. and Zhang, D. (2004). Robust image segmentation using fcm with spatial constraints based on new kernel-induced distance measure. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 34(4):1907–1916.

Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3):32–57.

Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*.

Engelbrecht, A. P. (2007). *Computational Intelligence: An Introduction*. Wiley Publishing, 2nd edition.

Ferrari, V., Tuytelaars, T., and Van Gool, L. (2006). Simultaneous Object Recognition and Segmentation by Image Exploration. In Ponce, J., Hebert, M., Schmid, C., and Zisserman, A., editors, *Toward Category-Level Object Recognition*, volume 4170 of *Lecture Notes in Computer Science*, pages 145–169. Springer Berlin Heidelberg.

Hathaway, R., Bezdek, J., and Hu, Y. (2000). Generalized fuzzy c-means clustering strategies using lp norm distances. *Fuzzy Systems, IEEE Transactions on*, 8(5):576–582.

Kang, Y., Yamaguchi, K., Naito, T., and Ninomiya, Y. (2011). Multiband image segmentation and object recognition for understanding road scenes. *Intelligent Transportation Systems, IEEE Transactions on*, 12(4):1423–1433.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4.

Krinidis, S. and Chatzis, V. (2010). A robust fuzzy local information c-means clustering algorithm. *Image Processing, IEEE Transactions on*, 19(5):1328–1337.

Lim, J. S. (1990). *Two-dimensional Signal and Image Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Mahalingam, T. and Mahalakshmi, M. (2010). Vision based moving object tracking through enhanced color image segmentation using haar classifiers. In *Proceedings of the 2nd International Conference on Trendz in Information Sciences and Computing, TISC-2010*, pages 253–260.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423.

Mei, X. and Lang, L. (2014). An image retrieval algorithm based on region segmentation. *Applied Mechanics and Materials*, 596:337341. cited By 0.

Mirghasemi, S., Sadoghi Yazdi, H., and Lotfizad, M. (2012). A target-based color space for sea target detection. *Applied Intelligence*, 36(4):960–978.

Szilagyi, L., Benyo, Z., Szilagyi, S., and Adam, H. (2003). Mr brain image segmentation using an enhanced fuzzy c-means algorithm. In *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE*, volume 1, pages 724–726 Vol.1.

Tian, X., Jiao, L., and Zhang, X. (2013). A clustering algorithm with optimized multiscale spatial texture information: application to SAR image segmentation. *International Journal of Remote Sensing*, 34(4):1111–1126.

Tran, D., Wu, Z., and Tran, V. (2014). Fast Generalized Fuzzy C-means Using Particle Swarm Optimization for Image Segmentation. In Loo, C., Yap, K., Wong, K., Teoh, A., and Huang, K., editors, *Neural Information Processing*, volume 8835 of *Lecture Notes in Computer Science*, pages 263–270. Springer International Publishing.

Wiener, N. (1964). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. The MIT Press.

Zhang, J.-Y., Zhang, W., Yang, Z.-W., and Tian, G. (2014). A novel algorithm for fast compression and reconstruction of infrared thermographic sequence based on image segmentation. *Infrared Physics & Technology*, 67(0):296–305.

Zhang, Q., Huang, C., Li, C., Yang, L., and Wang, W. (2012). Ultrasound image segmentation based on multi-scale fuzzy c-means and particle swarm optimization. In *Information Science and Control Engineering 2012 (ICISCE 2012), IET International Conference on*, pages 1–5.

Zhang, Q., Kamata, S., and Zhang, J. (2009). Face detection and tracking in color images using color centroids segmentation. In *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*, pages 1008–1013.

Zhuang, H., Low, K.-S., and Yau, W.-Y. (2012). Multichannel pulse-coupled-neural-network-based color image segmentation for object detection. *Industrial Electronics, IEEE Transactions on*, 59(8):3299–3308.