Differential Evolution based Fuzzy Clustering Technique: Application to Satellite Image Segmentation

Ujjwal Maulik  
Professor  
Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, West Bengal, India  
E-mail : drumaulik@cse.jdvu.ac.in

Indrajit Saha  
Lecturer  
Department of Information Technology, Academy of Technology, Adisaptagram-712121, West Bengal, India  
E-mail : indra_raju@yahoo.co.in

Abstract  
An important approach to unsupervised pixel classification in remote sensing satellite imagery is to use clustering in the spectral domain. The problem of classifying an image into different homogeneous regions is viewed as the task of clustering the pixels in the intensity space. In particular, satellite images contain landcover types of which cover significantly large areas, while some (e.g., bridges and roads) occupy relatively much smaller regions. Detecting regions or clusters of such widely varying sizes presents a challenging task. A Differential evolution based fuzzy clustering technique, is proposed in this article. Real coded encoding of the cluster centers is used for this purpose. Results demonstrating the effectiveness of the proposed technique are provided for several synthetic data set and also statistical significance tests have been performed to establish the superiority of the proposed algorithm. Different landcover regions in remote sensing imagery have also been classified using the proposed technique to establish its efficiency.

Keywords: Differential Evolution based Fuzzy Clustering, Fuzzy clustering, Cluster validity measures, Remote sensing satellite imagery, Statistical significant test.

1. Introduction  
Clustering [1-3] is a useful unsupervised data mining technique which partitions the input space into $K$ regions depending on some similarity/dissimilarity metric where the value of $K$ may or may not be known a priori. The main objective of any clustering technique is to produce a $K \times n$ partition matrix $U(X)$ of the given data set $X$, consisting of $n$ patterns, $X = \{x_1, x_2, \ldots, x_n\}$. The partition matrix may be represented as $U = \{u_{kj}\}, k = 1, \ldots, K$ and $j = 1, \ldots, n$, where $u_{kj}$ is the membership of pattern $x_j$ to the $k$th cluster. For fuzzy clustering of the data, $0 \leq u_{kj} \leq 1$, i.e., $u_{kj}$ denotes the degree of belongingness of pattern $x_j$ to the $k$th cluster. The objective of the Fuzzy C-means (FCM) algorithm [4-5] is to maximize the global compactness of the clusters. Fuzzy C-means [4,5] clustering algorithm be applied for clustering continuous data sets, where there is natural ordering among the elements of an attribute. Thus inherent distance measures, such as Euclidean distance, can be used to compute the distance between two feature vectors [6]. The algorithm optimizes a single objective function. Moreover motivated by this fact, here we are using a global optimization tools like Differential Evolution based clustering is trying to optimize the objective function ($J_{DE}$). The superiority of the proposed method over simulated annealing based fuzzy clustering [7], well known FCM [4,5] and hierarchical average linkage clustering [19] algorithms have been demonstrated on different synthetic and real life data sets on the basis of Xie-Beni (XB) [8] index and $I$ [3]index.

In 1995, Price and Storn proposed a new floating point encoded evolutionary algorithm for global optimization [9] and named it Differential Evolution (DE) [10,11] owing to a special kind of differential operator, which they invoked to create new offspring from parent vectors instead of classical crossover or mutation. DE algorithm is a stochastic optimization [12,13] method minimizing an objective function that can model the problem’s objectives while incorporating constraints [14]. The algorithm mainly has three advantages; finding the global minimum regardless of the initial parameter values, fast convergence while using a few control parameters. Being simple, fast, easy to use, quite effective in nonlinear constraint optimization including penalty functions and useful for optimizing multi-modal search spaces are the other important features of DE [15,16].

In this article we propose DE based fuzzy clustering algorithm. The proposed technique is used to find the final solution of clustering. The performance of the differential evolution based fuzzy clustering (DEFC) has been demonstrated on two artificial data sets. Statistical significance of the results is also established. The proposed technique is also applied on an IRS satellite image of the city of Calcutta. The superiority of the proposed technique, as compared to the well known FCM algorithm, simulated annealing based fuzzy clustering (SAFC) and widely used hierarchical average linkage clustering [19] is demonstrated both quantitatively and qualitatively.

2. Clustering Algorithms and Validity Measure

This section describes some well known clustering methods and cluster validity measure respectively.

2.1 Clustering Algorithms
2.1.1 Fuzzy C-means

Fuzzy C-means (FCM) [4,5] is a widely used technique that uses the principles of fuzzy sets to evolve a partition matrix \( U(X) \) while minimizing the measure

\[
J_m = \sum_{k=1}^{K} \sum_{i=1}^{n} u_{ik}^m D^2(z_i, x_k) \quad 1 \leq m \leq \infty \tag{1}
\]

where \( n \) is the number of data objects, \( K \) represents number of clusters, \( u \) is the fuzzy membership matrix (partition matrix) and \( m \) denotes the fuzzy exponent, is taken as 2. Here \( x_i \) is the \( i \)th data point and \( z_i \) is the center of \( i \)th cluster, and \( D(z_i, x_k) \) denotes the distance of point \( x_k \) from the center of the \( i \)th cluster.

FCM algorithm starts with random initial \( K \) cluster centers, and then at every iteration it finds the fuzzy membership of each data points to every cluster using the following equation

\[
u_{ik} = \frac{1}{\sum_{j=1}^{K} \left( \frac{1}{D(z_i, x_j)} \right)^{m-1}} \quad \text{for } 1 \leq i \leq K, 1 \leq k \leq n \tag{2}
\]

for \( 1 \leq i \leq K, 1 \leq k \leq n \), where \( D(z_i, x_k) \) and \( D(z_i, x_j) \) are the distances between \( x_i \) and \( z_i \), and \( x_k \) and \( z_k \) respectively, \( m \) is the weighting coefficient. (Note that while computing \( u_{ik} \) using Eqn. 2, if \( D(z_i, x_j) \) is equal to zero for some \( j \), then \( u_{ik} \) is set to zero for all \( i = 1, \ldots, K, i \neq j \), while \( u_{ik} \) is set equal to one.)

Based on the membership values, the cluster centers are recomputed using the following equation

\[
z_i = \frac{\sum_{k=1}^{n} u_{ik}^m x_k}{\sum_{k=1}^{n} u_{ik}^m} \quad 1 \leq i \leq K \tag{3}
\]

The algorithm terminates when there is no further change in the cluster centers. Finally, each data point is assigned to the cluster to which it has maximum membership.

2.1.2 Simulated Annealing based Fuzzy Clustering

Simulated annealing (SA) [17,18] is an optimization tool which has successful applications in a wide range of combinatorial optimization problems. The fact motivated researchers to use a SA to optimize the clustering problem where it provides near optimal solutions of an objective or fitness function in complex, large and multimodal landscapes. In SA based fuzzy clustering [7] a string or configuration encodes \( K \) cluster centers. For computing the fuzzy membership of all the points the encoded centers are used. Subsequently the string is updated using new centers. Thereafter the fitness \( J_a \) is computed as per Eqn. 1. The current string undergoes perturbation as follows: the position of perturbation is chosen randomly and the value of that position is replaced by some other value chosen randomly from the set of dimensional space. This way, perturbation of a string yields a new string. Fig. 1 demonstrates the SAFC algorithm.

2.2 Clustering Validity Indices

2.2.1 XB Index

The XB index [8] is defined as a function of the ratio of the total variation \( \sigma^2 \) to the minimum separation \( sep \) of the clusters. Here \( \sigma \) and \( sep \) can be written as

\[
\sigma(U, Z; X) = \sum_{k=1}^{K} \sum_{i=1}^{n} u_{ik}^m D^2(z_i, x_k) \tag{5}
\]

and

\[
sep = \min_{i \neq j} \| z_i - z_j \|^2 \tag{6}
\]

where \( \| \cdot \| \) is the Euclidean norm, and \( D(z_i, x_k) \), as mentioned earlier, is the distance between the pattern \( x_k \) and the cluster center \( z_i \). The XB index is then written as

\[
XB(U, Z; X) = \frac{\sigma(U, Z; X)}{n \times sep(z)} \tag{7}
\]
Note that when the partitioning is compact and good, value \( \sigma \) of should be low while sep should be high, thereby yielding lower values of the Xie-Beni (XB) index. The objective is therefore to minimize the XB index for achieving proper clustering.

### 2.2.2 I Index

A cluster validity index \( I \), proposed in [3] is defined as follows:

\[
I(K) = \left( \frac{1}{K} \times \frac{E_k}{E_K} \times D_k \right)^p
\]

(8)

where \( K \) is the number of clusters. Here,

\[
E_K = \sum_{i=1}^{K} \sum_{k=1}^{m} u_{j,k} \| x_j - z_k \|,
\]

(9)

and

\[
D_k = \max_{j \neq k} \| z_j - z_k \|.
\]

(10)

The index \( I \) is a composition of three factors, namely, \( \frac{1}{K} \), \( E_k \), and \( D_k \). The first factor will try to reduce index \( I \) as \( K \) is increased. The second factor consists of the ratio of \( E_k \), which is constant for a given data set, to \( E_K \), which decreases with increase in \( K \). To compute \( E_k \) the value of \( K \) in Eqn. 9 is taken as one i.e all the points are considered to be in the same cluster. Hence, because of this term, index \( I \) increases as \( E_K \) decreases. This, in turn, indicates that formation of more number of clusters, which are compact in nature, would be encouraged. Finally, the third factor, \( D_k \) (which measures the maximum separation between two clusters over all possible pairs of clusters), will increase with the value of \( K \). However, note that this value is upper bounded by the maximum separation between two points in the data set. Thus, the three factors are found to compete with and balance each other critically. The power \( p \) is used to control the contrast between the different cluster configurations. In this article, we have taken \( p = 2 \).

### 3. Proposed Differential Evolution based Fuzzy Clustering

Differential Evolution is a relatively recent heuristic designed to optimize problems over continuous domains. In DE, each decision variable is represented in the vector by a real number. As in any other evolutionary algorithm, the initial population of DE is randomly generated, and then evaluated. After that, the mutation process takes place. During the mutation stage, three parents are chosen and they generate a single offspring which competes with a parent to determine who passes to the following generation. DE generates a single offspring by adding the weighted difference vector between two parents to a third parent. During crossover, each offspring and parent vectors are participated for creation of trail vectors depending on crossover rate. The crossover rate (CR) has been provided by the user in the range [0, 1]. In the context of single-objective optimization, if the trail vector yields a lower objective function value than a predetermined population member, the newly generated vector replaces the vector with respect to which it was compared.

### 3.1 DEFC algorithm

The basic steps of DE are also followed in the DEFC algorithm. These are now described in detail.

#### 3.1.1 Vector representation and Population Initialization

Each vector is a sequence of real numbers representing the \( K \) cluster centres. For an \( d \)-dimensional space, the length of a vector is \( d \times K \), where the first \( N \) positions (or, genes) represent the \( d \) dimensions of the first cluster centre, the next \( N \) positions represent those of the second cluster centre, and so on. The \( K \) cluster centres encoded in each vector are initialized to \( K \) randomly chosen points from the data set. This process is repeated for each of the \( P \) vectors in the population, where \( P \) is the size of the population.

#### 3.1.2 Fitness computation

An objective/fitness function is associated with each vector that represents the degree of goodness of the solution encoded in the vector. The fitness of each vector \((J_m/ \text{XB})\) is computed either using Eqn. 1 or Eqn. 7 respectively. Subsequently, the centers encoded in a vector are updated using Eqn. 3.

#### 3.1.3 Mutation

The \( i \)th individual vector of the population at time step (generation) \( t \) has \( d \) components (dimensions), i.e.,

\[
\mathcal{G}_i(t) = [G_{i,1}(t), G_{i,2}(t), ..., G_{i,d}(t)]
\]

(12)

For each individual vector \( \mathcal{G}_i(t) \) that belongs to the current population, we use DE for randomly samples three other individuals, i.e., \( \mathcal{G}_m(t), \mathcal{G}_j(t), \text{ and } \mathcal{G}_k(t) \) from the same generation (for distinct \( k, i, j, \text{ and } m \)). It then calculates the (component wise) difference, scales it by a scalar \( F \) (usually \( \in [0,1] \)), and creates a trail offspring \( \mathcal{G}_k(t + 1) \) by adding the result to \( \mathcal{G}_m(t) \). Thus, for the \( n \)th component of each vector

\[
\mathcal{G}_{k,n}(t + 1) = \mathcal{G}_{m,n}(t) + F(\mathcal{G}_{r,n}(t) - \mathcal{G}_{j,n}(t))
\]

(13)

#### 3.1.4 Crossover

In order to increase the diversity of the perturbed parameter vectors, crossover is introduced. To this end, the trial vector:

\[
\hat{\mathcal{G}}(t + 1) = [G_{i,1}(t + 1), G_{i,2}(t + 1), ..., G_{i,d}(t + 1)]
\]

(14)

is formed, where

\[
\hat{U}_{k,d}(t + 1) = \begin{cases} \hat{\mathcal{G}}_{k,d}(t+1) & \text{if } \text{rand}_i(0,1) < \text{CR} \text{ and } k \neq \text{rand}(d) \\ \hat{\mathcal{G}}_{j,d}(t) & \text{if } \text{rand}_i(0,1) > \text{CR} \text{ and } k \neq \text{rand}(d) \end{cases}
\]

(15)
In Eqn. 15, \( rand(0, 1) \) is the \( j \)th evaluation of a uniform random number generator with outcome \( \in [0, 1] \). CR is the crossover rate \( \in [0, 1] \) which has to be determined by the user. \( rand(d) \) is a randomly chosen index \( \in 1, 2, \ldots, d \) which ensures that \( \vartheta_k(t+1) \) gets at least one parameter from \( \mathcal{U}_k(t+1) \).

3.1.5 Selection
To decide whether or not it should become a member of generation \( G + 1 \), the trial vector \( \mathcal{U}_k(t+1) \) is compared to the target vector \( \vartheta_k(t) \) using the greedy criterion. If vector \( \mathcal{U}_k(t+1) \) yields a smaller cost function value than \( \vartheta_k(t) \) then \( \mathcal{U}_k(t+1) \) is set to \( \vartheta_k(t) \), otherwise, the old value \( \vartheta_k(t) \) is retained.

3.1.6 Termination Criteria
The processes of mutation, crossover and selection are executed for a fixed number of iterations. The best vector seen up to the last generation provides the solution to the clustering problem. Elitism has been implemented at each generation by preserving the best vector seen up to that generation in a location outside the population. Thus on termination, this location contains the centres of the final clusters. The algorithm is outlined in Fig. 2.

![Figure 2: DEFC Algorithm](image)

4. Experimental Result
The performance of the described DEFC scheme has been compared with that of a SA based fuzzy clustering, FCM and hierarchical average linkage clustering [19] is provided for two artificial data sets.

4.1 Artificial Data Set

Data 1: This is an overlapping two dimensional data set where the number of clusters is five. It has 250 points. The value of K is chosen to be 5. The data set is shown in Fig. 3(a).

Data 2: This is also a two dimensional data set consisting of 900 points. The data set has 9 classes. The data set is shown in Fig. 3(b).

\[ MS(T,C) = \frac{\|r - C\|}{\|r\|} \]  

(16)

where

Figure 3: Two Artificial Data Set (a) and (b).
\[ T = \sqrt{\sum_{i} \sum_{j} T_{i,j}} \] (17)

The Minkowski score is the normalized distance between the two matrices. Lower Minkowski score implies better clustering solution, and a perfect solution will have a score zero.

### 4.4 Results

The Tables 1, and 2 reports the average values for MS, \( J_m \), XB and \( I \) indices obtained by different algorithms over 50 runs on synthetic data sets. It is evident from the tables that the DEFC method consistently outperform the others algorithms. It is also evident from the Fig. 4 that DEFC performs better than the other algorithms and provides lowest Minkowski Scores in all the runs for all the data sets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>MS</th>
<th>( J_m )</th>
<th>XB</th>
<th>( I )</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFC</td>
<td>0.3329</td>
<td>340.4261</td>
<td>0.6391</td>
<td>6.2103</td>
</tr>
<tr>
<td>SAFC</td>
<td>0.3509</td>
<td>345.4621</td>
<td>0.7981</td>
<td>6.5021</td>
</tr>
<tr>
<td>FCM</td>
<td>0.3616</td>
<td>360.7510</td>
<td>0.8457</td>
<td>4.6103</td>
</tr>
<tr>
<td>AL</td>
<td>0.4361</td>
<td>450.8215</td>
<td>1.6210</td>
<td>2.3245</td>
</tr>
</tbody>
</table>

Table 1: Average values MS, \( J_m \), XB and \( I \) indices for synthetic data set Data1.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>MS</th>
<th>( J_m )</th>
<th>XB</th>
<th>( I )</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFC</td>
<td>0.4432</td>
<td>278.4501</td>
<td>2.2331</td>
<td>1.3256</td>
</tr>
<tr>
<td>SAFC</td>
<td>0.4562</td>
<td>282.3215</td>
<td>2.4962</td>
<td>1.2261</td>
</tr>
<tr>
<td>FCM</td>
<td>0.5223</td>
<td>298.3215</td>
<td>3.4689</td>
<td>0.9626</td>
</tr>
<tr>
<td>AL</td>
<td>0.6516</td>
<td>389.0329</td>
<td>4.0983</td>
<td>0.2810</td>
</tr>
</tbody>
</table>

Table 2: Average values MS, \( J_m \), XB and \( I \) indices for synthetic data set Data2.

### 4.5 Statistical Significance Test

A non-parametric statistical significance test called Wilcoxon’s rank sum test for independent samples [21] has been conducted at the 5% significance level. Three groups, corresponding to the three algorithms (1. SAFC, 2. FCM, and 3. AL), have been created for each data set. Each group consists of the Minkowski Scores (MS) for the data sets produced by 50 consecutive runs of the corresponding algorithm. The median values of each group for all the data sets are shown in Table 3.

It is evident from Table 3 that the median values for DEFC are better than that for other algorithms. To establish that this goodness is statistically significant, Table 4 reports the \( P\)-values produced by Wilcoxon’s rank sum test for a time.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Data1</th>
<th>Data2</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFC</td>
<td>0.3329</td>
<td>0.4432</td>
</tr>
<tr>
<td>SAFC</td>
<td>0.3509</td>
<td>0.4562</td>
</tr>
<tr>
<td>FCM</td>
<td>0.3616</td>
<td>0.5223</td>
</tr>
<tr>
<td>AL</td>
<td>0.4361</td>
<td>0.6516</td>
</tr>
</tbody>
</table>

Table 3: Median value of the MS for the data sets over 50 consecutive runs of different algorithms.

Similar results are obtained for other data set and for all other algorithms compared to DEFC, establishing the significant superiority of the proposed technique.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>SAFC</th>
<th>FCM</th>
<th>AL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data1</td>
<td>1.4077e-004</td>
<td>1.6527e-004</td>
<td>5.3075e-005</td>
</tr>
<tr>
<td>Data2</td>
<td>1.2934e-004</td>
<td>1.3253e-004</td>
<td>4.9177e-005</td>
</tr>
</tbody>
</table>

Table 4: \( P\)-value produced by the Wilcoxon’s Rank Sum Test comparing DEFC with other algorithm.
5. Application for pixel classification

In this section, an IRS satellite image of a part of the city of Calcutta has been used for demonstrating unsupervised pixel classification. The results obtained by application of DEFC clustering have been reported and compared with that of SAFC and FCM clustering. The results are shown both graphically and numerically. To show the effectiveness of the DEFC scheme, a cluster validity index $J_m$, XB and $I$ [3] has been examined.

5.1 IRS image of Calcutta

The data used here was acquired from the Indian Remote Sensing Satellite (IRS-1A) [22] using the LISS-II sensor that has a resolution of 36.25m×36.25m. The image is contained in four spectral bands namely, blue band of wavelength 0.45-0.52 m, green band of wavelength 0.52-0.59 m, red band of wavelength 0.62-0.68 m, and near infrared band of wavelength 0.77-0.86 m.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>$J_m$</th>
<th>XB</th>
<th>$I$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFC</td>
<td>3666117.42</td>
<td>1.3246</td>
<td>54.7321</td>
</tr>
<tr>
<td>SAFC</td>
<td>3667282.28</td>
<td>1.6609</td>
<td>40.5021</td>
</tr>
<tr>
<td>FCM</td>
<td>3667871.81</td>
<td>2.1409</td>
<td>18.0436</td>
</tr>
</tbody>
</table>

Table 5: Average values $J_m$, XB and $I$ indices for IRS image of Calcutta over 50 runs.

Fig. 5 shows the Calcutta image in the near infrared band. Some characteristic regions in the image are the river Hooghly cutting across the middle of the image, several fisheries observed towards the lower-right portion, a township, Salt Lake, to the upper-left hand side of the fisheries. This township is bounded on the top by a canal. Two parallel lines observed towards the upper right hand side of the image correspond to the airstrips in the Dumdum airport. Other than these there are several water bodies, roads etc. in the image. It has been shown in [23] that this image can be well clustered into four classes which correspond to the classes turbid water (TW), pond water (PW), concrete (Concr.) and open space (OS).

Fig. 6, 7, and 8 shows the Calcutta image partitioned using the proposed DEFC, SAFC and the FCM algorithms respectively. From Fig. 6, it appears that the water class has been differentiated into turbid water (the river Hooghly) as combination of concrete and open space, which appears to be correct, since this particular region is known to have several open spaces. The canal bounding Salt Lake from the upper portion has also been correctly classified as PW. Also, the airstrip of Dumdum airport is classified rightly as belonging to the class concrete. Presence of some small areas of PW beside the airstrips is correct as these correspond to the several ponds around the region. The predominance of concrete on both sides of the river, particularly towards the bottom of the image is also correct. This region corresponds to the central part of the city of Calcutta.

Fig. 7 indicates that though SAFC also clustered reasonably well on this data set, the performance of DEFC is clearer and DEFC is able to distinguish among several classes more prominently. The FCM algorithm performs the worst. From Fig. 7, it can be noted that the river Hooghly and the city region has been incorrectly classified as belonging to the same class. It also appears from the figure that the whole of Salt Lake city has been put into one class. Hence the FCM result involves in a significant amount of confusion.

![Figure 5: IRS image of Calcutta in the NIR band with histogram equalization.](image)

The superiority of the DEFC technique can also be verified from the $I$ index values that are reported in Table 5. The $I$ index values for DEFC and FCM algorithms are tabulated along with SAFC. From the table, it is found that these values are 54.7321, 18.0436 and 40.5021 respectively. As a higher value of $I$ index indicates better clustering result, it follows that DEFC outperforms both its single objective counterpart and FCM algorithm.

5. Conclusion

The problem of fuzzy clustering has been modelled as optimization of cluster validity measure namely, $J_m$, Xie-Beni (XB) and also measure the performance of $I$ index. In this regard, the newly defined algorithm DEFC as been used. Results on different synthetic data sets indicate that DEFC consistently performing better. In this context, IRS satellite
images of Calcutta have been classified using the proposed technique and compared with other clustering algorithms to show its effectiveness. Good performance of DEFC method for such large image data sets shows that it may be motivating to use this algorithm in data mining applications also.

As a scope of further research, the DEFC based algorithm can also be extended to find the number of clusters automatically. Moreover, its extension for solving multiobjective optimization problems using the concept of amount of dominance [24] may be studied. The authors are currently working in these directions.

6. References


