1

Multilevel Image Segmentation Using Modified Genetic Algorithm (MfGA)-based Fuzzy C-Means

Sourav De[1], Sunanda Das[2], Siddhartha Bhattacharyya[3], and Paramartha Dutta[4]

[1] *Department of Computer Science and Engineering, Cooch Behar Government Engineering College, Cooch Behar, West Bengal, India*
[2] *Department of Computer Science and Engineering, University Institute of Technology, The University of Burdwan, Burdwan, West Bengal, India*
[3] *Department of Information Technology, RCC Institute of Information Technology, Kolkata, West Bengal, India*
[4] *Department of Computer Science and System Sciences, Visva-Bharati University, Santiniketan, West Bengal, India*

1.1 Introduction

Image segmentation plays a key role in image analysis and pattern recognition and also has other application areas, like machine vision, biometric measurements, medical imaging, and so on for the purposes of detecting, recognzing or tracking an object. The objective of image segmentation is to segregate an image into a set of disjoint regions on the basis of uniform and homogeneous attributes such as intensity, color, texture, and so on. The attributes in the different regions are heterogeneous to each other, but the attributes in the same region are homogeneous to each other. On the basis of the inherent features of an image and some *a priori* knowledge and/or presumptions about the image, the pixels of that image can be classified successfully. Let an image be represented by I, and that image can be partitioned into n number of subimages (i.e., $I_1, I_2, ..I_n$), such that:

- $\bigcup_{i=1}^{K} I_i = I$;
- I_i is a connected segment;
- $I_i \neq \emptyset$ for $i = 1, \cdots, K$; and
- $I_i \cap I_j = \emptyset$ for $i = 1, \cdots, K, j = 1, \cdots, K$, and $i \neq j$.

Different image segmentation techniques have been developed on the basis of the discontinuity and similarity of the intensity levels of an image. Multilevel grayscale and color image segmentation became a perennial research area due to the diversity of the grayscale and color-intensity gamuts. In different research articles, the image segmentation problem is handled by different types of classical and nonclassical image-processing algorithms.

Among different types of classical image segmentation techniques [1–3], edge detection and region growing, thresholding, normalized cut, and others are well-known image segmentation techniques to segment the multilevel grayscale images. The image

Hybrid Intelligence for Image Analysis and Understanding, First Edition.
Edited by Siddhartha Bhattacharyya, Indrajit Pan, Anirban Mukherjee, and Paramartha Dutta.

Companion Website: www.wiley.com/go/bhattacharyya/hybridintelligence

segmentation by the edge detection techniques are done by finding out the boundaries of the objects in that image. But an incorrect segmentation may happen with edge detection algorithms as these processes are not assistive for segmenting any complicated images or the blur images. Region-growing techniques are not efficiently applied for multilevel image segmentation, as the different regions of an image are not fully separated. The image segmentation using thresholding techniques are fully dependent on the histogram of that image. The images that have the distinctive background and objects can be segmented efficiently by the thresholding techniques. This process may not work if the distribution of the pixels in the image is very complex in nature.

To solve the clustering problems, the *k*-means [4] and the fuzzy *c*-means [5] are two very renowned clustering algorithms. The common feature of these two algorithms is that both start with a fixed number of predefined cluster centroids. The meaning of *k* in the *k*-means algorithm is the same as that of *c* in the FCM algorithm, and both *k* and *c* signify the number of clusters. At the initial stage of the *k*-means algorithm, the *k* number of cluster centroids are generated randomly. The clusters are formed with the patterns on the basis of the minimum distance between the pattern and the cluster centroids. The cluster centriods update their positions iteratively by minimizing the sum of the squared error in between the corresponding members within the clusters. This clustering algorithm is considered as hard clustering as it is assumed that each pattern is a member of a single cluster. In FCM, the belongingness of a pattern within a cluster is defined by the degree of membership value of that pattern. The well-known least-squares objective function is used to minimize, and the degree of membership of the pattern will be updated to optimize the clustering problem. That is why FCM is considered a soft clustering algorithm.

Quality improvement of the *k*-means algorithm has been reported in different research articles. Luo *et al.* [6] proposed a spatially constrained *k*-means approach for image segmentation. Initially, the *k*-means algorithm is applied in feature space, and after that, the spatial constraints are considered in the image plane. Khan and Ahmad [7] tried to improve the *k*-means algorithm by modifying the initialization of the cluster centers. The high-resolution images are segmented by the *k*-means algorithm after optimization by the Pillar algorithm [8]. The *k*-means algorithm along with the improved watershed segmentation algorithm are employed to segment the medical images [9]. The noisy images can be segmented efficiently by the modified FCM algorithm, which is proposed by Ahmed *et al.* [10, 11]. In this process, the objective function for the standard FCM algorithm has been modified to allow the labels in the immediate neighborhood of a pixel to influence its labeling [10]. The spatial information of the image is incorporated in the traditional FCM algorithm to segment the noisy grayscale images [12]. In this method, the spatial function is considered as the sum of all the membership functions within the neighborhood of the pixel under consideration. Noisy medical images can be segmented by FCM [13]. In this approach, the input images are converted into multiscale images by smoothing them in different scales. The scaling operation, from high scale to low scale, is performed by FCM. The image with the highest scale is processed with the FCM, and after that, the membership of the current scale is determined by the cluster centers of the previous scale. Noordam *et al.* [14] presented a geometrically guided FCM (GG-FCM) algorithm in which geometrical information is added with the FCM algorithm during clustering. In this process, the clustering is guided by the condition of each pixel, which is decided by the local

neighborhood of that pixel. Different types of modified objective functions are applied to segment the corrupted images to derive a lower error rate [10, 15]. A modified FCM algorithm, considering spatial neighborhood information, is presented by Chen *et al.* [16]. In this method, a simple and effective 2D distance metric function is projected by considering the relevance of the current pixel to its neighbor pixels, and an objective function is also formed to update the cluster centers simultaneously in two dimensions of the pixel value and its neighboring value.

However, the *k*-means algorithm and fuzzy *c*-means algorithm have some common disadvantages. Both the algorithms need a predefined number of clusters, and the centroid initialization is also a problem. It may not be feasible to know the exact number of clusters beforehand in a large data set. It has been observed in different research articles that different initial numbers of clusters with different initializations had been applied to determine the exact number of clusters in a large data set [17]. Moreover, it may not be effective if the algorithm starts with a limited number of center of clusters. This type of problem is known as the *number of clusters dependency problem* [18]. Ultimately, it may happen that the solutions may be stuck in local minima instead of obtaining a global optimal solution.

In this real-world scenario, most of the problems can be considered as optimization problems. The differentiable functions are solved by the traditional heuristic algorithms, but the real-world optimization problems are nondifferentiable. It is rare that the nondifferentiable optimization functions are properly solved by a heuristic algorithm. In many research articles, it has been found that nondifferentiable optimization functions are solved by different types of metaheuristic approaches, and these algorithms are now becoming more popular in the research arena. Applying stochastic principles, evolutionary algorithms are now becoming an alternative way to solve optimization problems as well as clustering problems. These types of algorithms work on the basis of probabilistic rules to search for a near-optimal solution from a global search space. By inspiring the principle of natural genetics, genetic algorithms (GAs), differential evolution, particle swarm optimization, and so on are some of the examples of evolutionary algorithms.

GAs [19], as randomized search and optimization techniques, are applied efficiently to solve different types of image segmentation problems. Inspired by the principles of evolution and natural genetics, GAs apply three operators (viz., selection, crossover, and mutation over a number of generations) to derive potentially better solutions. Without having any *a priori* knowledge about the probable solutions to the problem or difficulties to formulate the problem, GAs can be the solution to solve those types of problems. Due to the generality characteristic, GAs demand a segmentation quality measurement to evaluate the segmentation technique. Another important characteristic of GAs is that they can derive the optimal or near-optimal solutions for their balanced global and local search capability.

The GA in combination with wavelet transformation is applied to segment multilevel images [20]. At first, the wavelet transform is applied to reduce the size of the original histogram, and after that, the number of thresholds and the threshold values are resolved with the help of a GA on the basis of the lower resolution version of the histogram. Fu *et al.* [21] presented an image segmentation method using a multilevel threshold of gray-level and gradient-magnitude entropy based on GAs. A hybrid GA is proposed by Hauschild *et al.* [22] for image segmentation on the basis of the *q*-state Potts spin

glass model to a grayscale image. In this approach, a set of weights for a q-state spin glass is formed from the test image, and after that, the candidate segmented images are generated using the GA until a suitable candidate solution is detected. The hierarchical local search is applied to speed up the convergence to an adequate solution [22]. The GA in combination with the multilayer self-organizing neural network (MLSONN) architecture are employed to segment the multilevel grayscale images [11]. In [11], it is claimed that the MLSONN architecture, with the help of multilevel sigmoidal (MUSIG) activation function, is not capable of incorporating the image heterogeneity property in the multilevel grayscale image segmentation process. To induce the image content heterogeneity in the segmentation process, the authors employed a GA to generate the optimized class levels for designing the optimized MUSIG (OptiMUSIG) activation function [11]. Now, the OptiMUSIG activation function based MLSONN architecture is capable of segmenting the multilevel grayscale images. The variable threshold–based OptiMUSIG activation function is also efficient for grayscale image segmentation [23]. To decrease the effect of isolated points on the k-means algorithm, a GA is applied to enhance the k-means clustering algorithm [24].

The performance and the shortcomings of FCM algorithms can be improved by different types of evolutionary algorithms. Biju *et al.* [25] proposed a genetic algorithm–based fuzzy c-means (GAFCM) technique to segment spots of complementary DNA (cDNA) microarray images for finding gene expression. An improved version of the GA is presented by Wei *et al.* [26]. In this approach, the genetic operators are modified to enhance the global searching capability of GAs. To improve the convergence speed, the improvised GA applies FCM optimization after each generation of genetic operation [26]. The GA inspired with the FCM algorithm is capable of segmenting the grayscale images [27]. In this approach, the population of GAs is generated with the help of FCM. Jansi and Subashini [28] proposed a GA integrated with FCM for medical image segmentation. The resultant best chromosome, derived by the GA, is applied as the input in the FCM. The drawback of the FCM algorithm (i.e., convergence to the local optima solution,) is overcome in this approach.

The main concern of this chapter is to overcome the shortcomings of the FCM algorithm as this algorithm is generally stuck to a local minima point. To eliminate this drawback, an evolutionary algorithm is the better choice to deal with these types of problems. A GA has the capability to find the global optimal or near–global optimal solutions in a large search space. That is why the authors considered the GA to handle this problem. In this chapter, a modified genetic algorithm (MfGA)-based FCM algorithm is proposed. For that reason, some modifications are made in the traditional GA. A weighted mean approach is introduced for the chromosome generation in the population initialization stage. Like the traditional GA, the crossover probability is not fixed throughout the generations in the MfGA. As crossover probability plays a vital role in GAs, the value of the crossover probability decreases as the number of generations increases. The resultant class levels, derived by the MfGA, are applied as the input of the FCM algorithm. The proposed MfGA-based FCM algorithm is compared with the traditional FCM algorithm [5] and the GA-based FCM algorithm [28]. The above-mentioned algorithms are employed on three benchmark images to determine the performance of those algorithms. Two standard efficiency measures, the correlation coefficient (ρ) [11] and the empirical measure Q [29], are applied as the evaluation functions to measure the quality of the segmented images. The experimental results show that the proposed

MfGA-based FCM algorithm outperforms the other two algorithms to segment the multilevel grayscale images.

This chapter is organized in the following ways. Section 1.2 discusses the traditional FCM algorithm. Two quality evaluation metrics for image segmentation, the correlation coefficient (ρ) and the empirical measure (Q), are narrated in Section 1.4. Before that, in Section 1.3, the proposed MfGA is illustrated. After that, the MfGA-based FCM is discussed in Section 1.5. The experimental results and the comparison with the FCM algorithm [5] and the GA-based FCM [28] algorithm are included in Section 1.6. Finally, Section 1.7 concludes the chapter.

1.2 Fuzzy *C*-Means Algorithm

The FCM, introduced by Bezdek [5], is the most widely applied fuzzy clustering method. Basically, this algorithm is an extension of the hard *c*-means clustering method and is based on the concept of fuzzy *c*-partition [30]. Being an unsupervised clustering algorithm, FCM is associated with the fields of clustering, feature analysis, and classifier design. The wide application areas of FCM are in astronomy, geology, chemistry, image processing, medical diagnosis, target recognition, and others.

The main objectives of the FCM algorithm are to update the cluster centers and to calculate the membership degree. The clusters are formed on the basis of the distance calculation between data points, and the cluster centers are formed for each cluster. The membership degree is applied to show the belongingness of each data point with each cluster, and the cluster centers are also updated with this information. This means that every data point in the data set is associated with every cluster, and among them, a data point may be considered in a cluster when it has high membership value with that cluster and low membership value with the rest of the clusters in that data set.

Let $X = x_1, x_2, \ldots, x_N, x_k \in R^n$ be a set of unlabeled patterns, where N is the number of patterns and each pattern has n number of features. This algorithm tries to minimize the value of an objective function that calculates the quality of the partitioning that divides the data set into C clusters. The distance between a data point x_k to a cluster center T_i is calculated by the well-known Euclidean distance, and it is represented as [5]:

$$D_{ik}^2 = \sum_{j=1}^{n} (x_{kj} - T_{ij})^2, \quad 1 \le k \le N, 1 \le i \le C \tag{1.1}$$

where the squared Euclidian distance in n-dimensional space is denoted as D_{ik}^2. The membership degree is calculated as [5]:

$$U_{ik} = \frac{1}{\sum_{j=1}^{C} \left(\frac{D_{ik}}{D_{jk}}\right)^{\frac{2}{(m-1)}}}, \quad 1 \le k \le N, 1 \le i \le C \tag{1.2}$$

where U_{ik} represents the degree of membership of x_k in the i^{th} cluster. The degree of fuzziness is controlled by the parameter $m > 1$. It signifies that every data point has a degree of membership in every cluster.

The centroids are updated according to the following equation [5]:

$$T_{ij} = \frac{\sum_{k=1}^{N} (U_{ik})^m x_k}{\sum_{k=1}^{N} (U_{ik})^m}, \quad 1 \le i \le C \tag{1.3}$$

Ultimately, the membership degree of each point is measured using equation (1.2) with the help of new centroid values.

1.3 Modified Genetic Algorithms

Conventional GAs [19] are well-known, efficient, adaptive heuristic search and global optimization algorithms. The basic idea of GAs took from the evolutionary ideas of natural selection and genetics. A GA provides a near-optimal solution in a large, complex, and multimodal problem space. It contains a fixed population size over search space. In GAs, the performance of the entire population can be upgraded instead of improving the performance of the individual members in the population-based optimization procedure. The evolution starts by generating the population through random creation of individuals. These individual solutions are known as *chromosomes*. The quality or goodness of each chromosome or individual is assessed by its *fitness* value. The potentially better solutions are evolved after applying three well-known genetically inspired operators like *selection, crossover*, and *mutation*.

In this chapter, a MfGA is proposed to improve the performance of the conventional GA. In the chromosome representation of the conventional GA, each chromosome contains N number of centroids to cluster a data set into N number of clusters. The initial values of the chromosomes are selected randomly. The clusters are formed between the cluster centroids and the individual data points on the basis of some criteria. In most of the cases, the distance between the data points is the selection criterion. A data point is included in a cluster when the distance between that data point and the particular cluster centroid is minimum rather than the same with other cluster centroids. It may happen that a cluster may contain very few data points compared to other clusters in that data set. The relevance of that small cluster may decrease in that data set. At the same time, the spatial information of the data points in the data set is also not considered. To overcome that, N+1 number of cluster centroids are selected in the initial stage to generate the N number of cluster centroids. Steps of MfGA are discussed here:

- *Population initialization*: In MfGA, the population size is fixed, and the lengths of individual chromosomes are fixed. At the initial stage, N+1 number of cluster centroids are generated randomly within the range of minimum and maximum values of the data set, and the cluster centroids are denoted as $R_1, R_2, R_3,..,R_i, .., R_{n+1}$. These cluster centroids are temporary, and they are used to generate the actual cluster centroids. The weighted mean between the temporary cluster centroids of R_i and R_{i+1} is applied to generate the actual cluster centroids L_i, and it is represented

as:

$$L_i = \frac{\sum_{I=R_i}^{R_{i+1}} f_I \times I}{\sum_{I=R_i}^{R_{i+1}} f_I} \tag{1.4}$$

where I represents the value of the data point and f_I denotes the frequency of the I^{th} data point. For example, the cluster centroid L_1 is generated after taking the weighted mean between the temporary cluster centroids R_1 and R_2. The cluster centroid L_i, having the most frequency value in between the cluster centroids R_i and R_{i+1}, will be selected by this process, and this will make a good chromosome for future generation. Ultimately, the spatial information is taken into consideration at the population initialization time.

- *Selection*: After population initialization, the effectiveness of the individual chromosomes is determined by a proper fitness function so that better chromosomes are taken for further usage. This is done in the perception that the better chromosomes may develop by transmitting the superior genetic information to new generations, and they will persist and generate offspring. In the selection step, the individuals are selected for mating on the basis of the fitness value of each individual. This fitness value is used to associate a probability of selection with each individual chromosome. Some of the well-known selection methods are roulette wheel selection, stochastic universal selection, Boltzmann selection, rank selection, and so on [19].
- *Crossover*: The randomly selected parent chromosomes exchange their information by changing parts of their genetic information. Two chromosomes, having the same probability of crossover rate, are selected to generate offspring for the next generation. The crossover probability plays a vital role in this stage, and it is used to show a ratio of how many parents will be picked for mating. In conventional GA, the crossover probability is same throughout the process. It may happen that a good chromosome may be mated with a bad chromosome, and that good chromosome is not stored in the next stage. To overcome this drawback, it is suggested in this process that the crossover probability rate will decrease as the number of iterations will increase. Here, the crossover probability is inversely proportional to the number of iterations so that the better chromosomes will remain unchanged and will go to the next generation of population. Mathematically, it is represented as:

$$C_p = C_{max} - \left(\frac{C_{max} - C_{min}}{IT_{max} - IT_{cur}} \right) \tag{1.5}$$

where C_p is the current crossover probability, and C_{max} and C_{min} are the maximum and minimum crossover probability, respectively. The maximum number of iterations and the current number of iterations are represented as IT_{max} and IT_{cur}, respectively.
- *Mutation*: The sole objective of the mutation stage is to introduce the genetic diversity into the population. Being a divergence operation, the frequency of the mutation operation is much less, and so the members of the population are affected much less. Mutation probability is taken as a very small value for this reason. Generally, a value in the randomly selected position in the chromosome is flipped.

1.4 Quality Evaluation Metrics for Image Segmentation

It is absolutely necessary to measure the quality of the final segmented images after segmenting the images by different types of segmentation algorithms. Usually, different statistical mathematical functions are employed to evaluate the results of the segmentation algorithms. Here, two unsupervised approaches are provided to measure the goodness of the segmented images in Sections 1.4.1 and 1.4.2.

1.4.1 Correlation Coefficient (ρ)

The degree of the similarity between the original and segmented images can be measured by using the standard measure of correlation coefficient (ρ) [11], and it is represented as [11]:

$$\rho = \frac{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(R_{ij}-\overline{R})(G_{ij}-\overline{G})}{\sqrt{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(R_{ij}-\overline{R})^2}\sqrt{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(G_{ij}-\overline{G})^2}} \tag{1.6}$$

where $R_{ij}, 1 \le i,j \le n$ and $G_{ij}, 1 \le i,j \le n$ are the original and the segmented images, respectively, each of dimensions $n \times n$. The respective mean intensity values of R_{ij} and G_{ij} are denoted as $\overline{R}$ and $\overline{G}$, respectively. The higher value of ρ signifies the better quality of image segmentation [11].

1.4.2 Empirical Measure $Q(I)$

A empirical measure, $Q(I)$, is used to evaluate the goodness of the segmented images, and it is suggested by Borsotti *et al.* [29, 31]. It is denoted as [29]:

$$Q(I) = \frac{1}{1000.S_I}\sqrt{N}\sum_{g=1}^{N}\left[\frac{e_g^2}{1+\log\ S_g} + \left(\frac{N(S_g)}{S_g}\right)^2\right] \tag{1.7}$$

where the area of the g^{th} region of an image (I) is denoted as S_g, and the number of regions having an area S_g is signified by $N(S_g)$. S_I is the area of an image (I) to be segmented. The squared color error of region g, e_g^2, is noted as [29, 31]:

$$e_g^2 = \sum_{v\in(r,g,b)}\sum_{pl\in RI_g}(C_v(pl) - \hat{C}_v(RI_g))^2 \tag{1.8}$$

The number of pixels in region g is presented as RI_g. The average value of feature v (red, green, or blue) of a pixel pl in region g is referred to as [29, 31]:

$$\hat{C}_v(RI_g) = \frac{\sum_{pl\in RI_g} C_v(pl)}{S_g} \tag{1.9}$$

where $C_v(pl)$ signifies the value of component v for pixel pl.

A smaller value of Q implicates better quality of segmentation [29]. In this chapter, the quality of the segmented images is evaluated by these measures, which are applied as different objective functions.

1.5 MfGA-Based FCM Algorithm

FCM has three major drawbacks: there must be *a priori* information of the number of classes to be segmented, this algorithm can only be applied to hyperspherical-shaped clusters, and it often converges to local minima [6]. In this chapter, we have tried to overcome the drawback about the convergence to the local minima. In many cases, it has been observed that a FCM algorithm will easily converge to the local minimum point and the clustering will be affected if the initial centroid values are not initialized correctly. The spatial relative information is not considered in the process of the FCM algorithm. The selection of the initial cluster centers and/or the initial membership value plays a vital role in the performance of the FCM algorithm. The quick convergence and drastic reduction of processing time can be possible in FCM if the selection of the initial cluster center is very close to the actual final cluster center.

The drawback of FCM can be overcome with the help of the proposed MfGA algorithm. The optimized class levels, obtained by the proposed MfGA algorithm, are applied as the initial class levels in the FCM algorithm. Another advantage of this proposed method is that the image content heterogeneity is also considered. The pixels, having most occurrence in the image, have the higher probability for being selected as the class levels in the initial stage. The flowchart in Figure 1.1 shows the steps of the MfGA-based FCM algorithm.

The pseudo-code of the GA-based FCM is as follows:

1. *Pop* ← Generate *P* number of feasible solutions randomly with *N* number of class levels.
2. Calculate fitness (*P*).
3. For i=1 to *itr*, do the following:
   A. Apply selection on *Pop*.
   B. Apply crossover operation with crossover probability C_p.
   C. Apply mutation operation with mutation probability μ_m.
   D. End for.
4. Return the best chromosome with the class levels.
5. Initialize cluster centers of the FCM algorithm with the best solution derived from GA.
6. For i=0 to itr2, do the following:
   A. Update the membership matrix (U_{ik}) using equation (1.2).
   B. Update the centroids [equation (1.3)].
7. Return the ultimate cluster centroids/class levels.

The pseudo-code of the MfGA-based FCM is as follows:

1. Generate *M* number of chromosomes randomly with $N + 1$ number of temporary cluster centroids individually.
2. Using weighted mean equation (1.4), generate *M* number of chromosomes with *N* number of actual cluster centroids individually.
3. Calculate the fitness computation of population using a relevant fitness function.
4. Repeat (for a predefined number of iterations or until a certain condition is satisfied).
   A. Select parents from population, on the basis of fitness values.

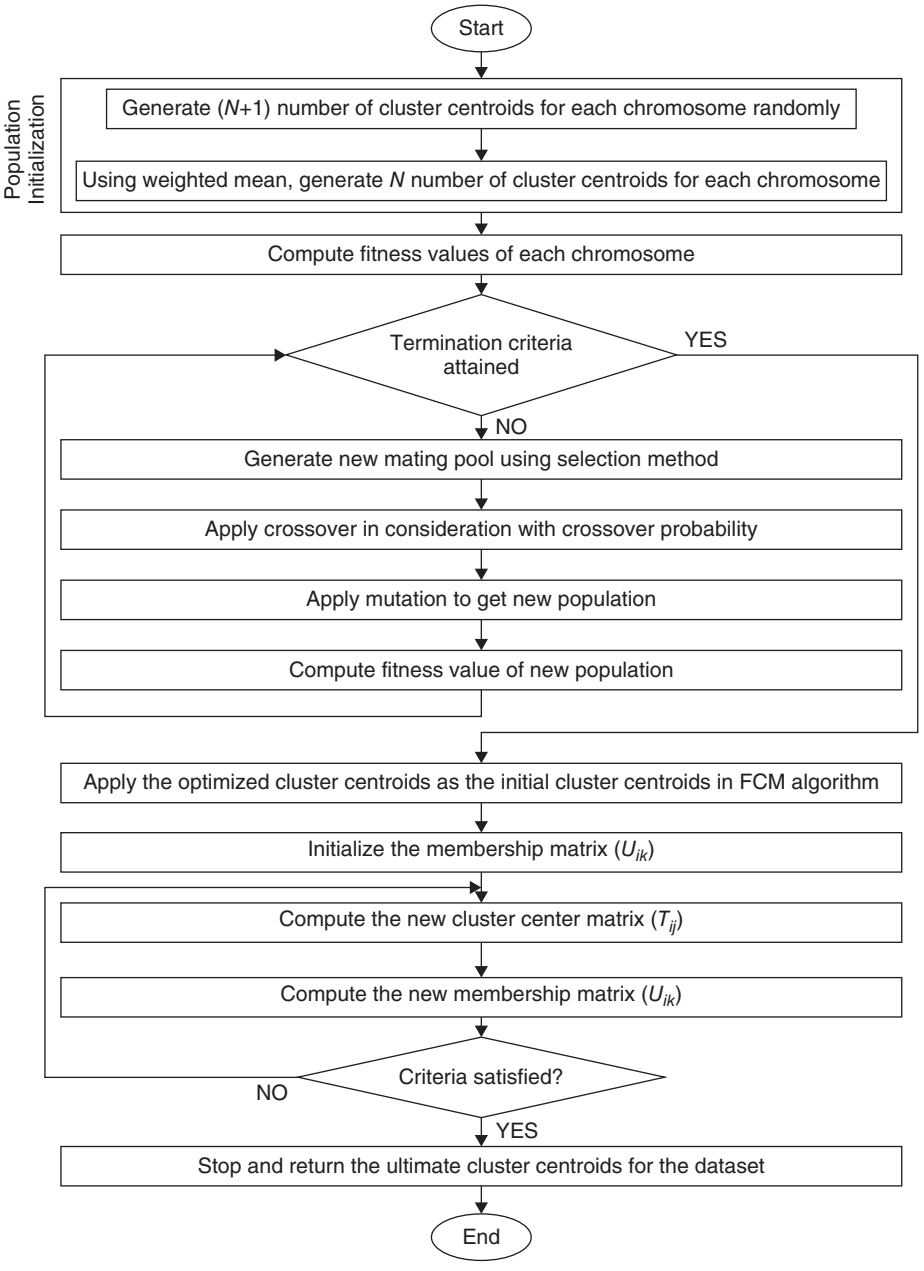

Figure 1.1 Flowchart of MfGA-based FCM algorithm.

   B. Execute crossover and mutation to create a new population [crossover probability is applied using equation (1.5)].
   C. Compute fitness of new population.
5. Return the best chromosome with the class levels.

6. Initialize cluster centers of the FCM algorithm with the best solution derived from the GA.
7. For i=0 to itr2, do the following:
   A. Update the membership matrix (U_{ik}) using equation (1.2).
   B. Update the centroids [equation (1.3)].
8. Return the ultimate cluster centroids/class levels.

1.6 Experimental Results and Discussion

Multilevel grayscale image segmentation with the MfGA-based FCM algorithm was demonstrated with three real-life grayscale images (viz., Lena, baboon, and peppers) of dimensions 256×256. Experimental results in quantitative and qualitative terms are reported in this section. The multilevel images are segmented in $K = \{6, 8, 10\}$ classes, but the results are reported for $K = \{6, 8\}$ classes. Results are also presented for the segmentation of the multilevel grayscale images with the GA-based FCM algorithm [5] and also with the FCM algorithm [28]. To measure the efficiency of the different algorithms qualitatively, two evaluation functions (ρ, Q), presented in equations (1.6) and (1.7), respectively, have been used in this chapter.

In the initial stage, the class levels are generated by the proposed MfGA algorithm, and after that the obtained class levels are supplied as the input in the FCM algorithm. For that, the pixel intensity levels of the multilevel grayscale images and the number of classes (K) to be segmented are supplied as inputs to this process. The randomly generated real numbers within the range of the minimum and maximum intensity values of the input image are applied to create the chromosomes for this process. These components of the chromosomes are treated as the class levels or class boundaries to segment the input image. For example, to segment an image into eight segments, the chromosomes with nine class levels are generated at the starting point. The process of generating the chromosomes and creating the chromosome pool in the GA-based FCM method is the same as the proposed method. A population size of 50 has been used for the proposed and GA-based FCM method. At the initial stage, the class levels are also generated randomly in the FCM method.

The GA operators (viz., selection, crossover, and mutation) are used in the proposed and GA-based FCM [28] approaches. The roulette wheel approach is employed for the selection phase for both the GA-based approaches. Afterward, the crossover and mutation operators are applied to generate the new population. Though the crossover probability value is fixed in the GA-based FCM approach, the crossover probability value in the MfGA-based FCM method is dependent on the number of iterations, and that works according to equation (1.5). The crossover probability value is 0.9 for all the stages in the GA-based FCM algorithm. The maximum crossover probability value (C_{max}) and minimum crossover probability value (C_{min}) are applied as 0.9 and 0.5, respectively, in the proposed MfGA-based FCM method. In both GA-based approaches, the single-point crossover operation is used in the crossover stage, and 0.01 is considered as the mutation probability. The new population is formed after the mutation operation.

The class levels generated by the proposed algorithm on the basis of the two fitness functions (ρ and Q) for different numbers of classes are tabulated in Tables 1.1 and 1.2 for Lena images. In this type of tables, the number of segments (# segments), the name

Table 1.1 Class boundaries and evaluated segmentation quality measures ρ by different algorithms for different classes of the Lena image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	1	7, 61, 107, 143, 179, 229	0.9133
		2	6, 64, 109, 144, 177, 229	0.9213
		3	9, 64, 107, 143, 179, 229	0.9194
		4	**7, 64, 110, 145, 180, 227**	**0.9227**
	GA-based FCM	1	47, 76, 105, 135, 161, 203	0.9823
		2	**50, 93, 124, 148, 173, 206**	**0.9824**
		3	47, 78, 105, 134, 161, 203	0.9823
		4	47, 77, 105, 135, 161, 203	0.9823
	Proposed	**1**	**47, 76, 105, 135, 161, 203**	**0.9828**
		2	50, 92, 123, 148, 173, 206	0.9824
		3	50, 94, 125, 149, 173, 204	0.9824
		4	**47, 76, 105, 135, 161, 203**	**0.9828**
8	FCM	**1**	**4, 37, 71, 105, 131, 155, 188, 231**	**0.9559**
		2	5, 37, 70, 105, 131, 155, 188, 231	0.9273
		3	4, 37, 71, 105, 131, 154, 188, 230	0.9251
		4	6, 37, 71, 105, 131, 155, 188, 231	0.9266
	GA-based FCM	1	42, 58, 84, 104, 130, 152, 178, 207	0.9893
		2	**42, 58, 83, 105, 131, 152, 176, 207**	**0.9894**
		3	46, 72, 99, 122, 140, 156, 179, 208	0.9893
		4	**42, 58, 83, 105, 130, 152, 176, 207**	**0.9894**
	Proposed	1	46, 71, 98, 122, 140, 157, 179, 208	0.9897
		2	**45, 72, 99, 124, 141, 158, 179, 208**	**0.9898**
		3	46, 72, 99, 123, 140, 156, 177, 207	0.9896
		4	**46, 72, 98, 122, 142, 157, 179, 208**	**0.9898**

of the algorithm (Algorithm), the serial number (#), class levels, and the fitness value (*fit val*) are accounted in the columns. The class levels evaluated by the FCM algorithm and the GA-based FCM algorithm are also reported in these tables. Each type of experiments is performed 50 times, but only four good results are tabulated in these tables in respect to the number of segments and in respect to the individual algorithms. The best results obtained by any process for each number of segments are highlighted by **boldface** type.

The mean and standard deviations are evaluated for different algorithm-based fitness values using different types of fitness functions, and these results for Lena images are reported in Table 1.3. The time taken by different algorithms is also preserved, and the mean of the time taken by the different algorithms is also mentioned in this table. The good results are marked in **bold**.

For segmenting the Lena image, the proposed algorithm outperforms the other two algorithms. In Table 1.1, the ρ values, derived by the MfGA-based FCM algorithm, are better than the other two algorithms for a different number of segments. The same observation can be obtained if anyone goes through the reported results in Table 1.2. The Q-based fitness values, obtained by the proposed algorithm, are much better than the results derived by the other two processes. In Table 1.3, the reported mean and standard

Table 1.2 Class boundaries and evaluated segmentation quality measures *Q* by different algorithms for different classes of the Lena image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	1	7, 62, 107, 144, 179, 228	24169.019
		2	8, 64, 107, 143, 177, 229	19229.788
		3	7, 64, 106, 140, 179, 231	24169.018
		4	**5, 66, 110, 142, 180, 229**	**15017.659**
	GA-based FCM	**1**	**47, 77, 105, 135, 161, 203**	**10777.503**
		2	50, 93, 124, 148, 173, 206	16017.944
		3	47, 74, 105, 135, 161, 203	12194.313
		4	47, 76, 108, 134, 161, 203	10909.195
	Proposed	1	47, 78, 105, 135, 161, 203	7117.421
		2	**47, 77, 105, 134, 161, 204**	**7020.245**
		3	50, 93, 124, 148, 173, 206	13146.389
		4	50, 94, 126, 158, 179, 204	10577.836
8	FCM	1	4, 38, 71, 105, 131, 155, 188, 229	64095.622
		2	5, 37, 69, 105, 132, 155, 188, 231	83743.507
		3	6, 37, 71, 105, 129, 155, 187, 230	108324.494
		4	**4, 37, 71, 105, 131, 155, 184, 231**	**48229.038**
	GA-based FCM	1	46, 72, 99, 122, 140, 157, 179, 208	49382.388
		2	46, 73, 100, 127, 149, 169, 193, 212	90707.0511
		3	46, 72, 99, 122, 140, 157, 179, 208	47419.664
		4	**46, 73, 99, 122, 140, 158, 180, 208**	**46906.552**
	Proposed	1	42, 58, 83, 105, 130, 152, 176, 207	33531.256
		2	42, 59, 84, 107, 129, 152, 175, 207	32630.511
		3	**42, 58, 83, 105, 131, 152, 176, 207**	**32496.323**
		4	46, 72, 99, 122, 140, 157, 179, 208	47429.683

Table 1.3 Different algorithm-based means and standard deviations using different types of fitness functions and mean of time taken by different algorithms for the Lena image

Fitness function	# Segments	Algorithm	Mean ± standard deviation	Mean time
ρ	6	FCM	0.9311 ± 0.02033	31 sec
		GA-based FCM	0.9623 ± 0.00003	23 sec
		Proposed	**0.9824 ± 0.00002**	**20 sec**
	8	FCM	0.9506 ± 0.01814	41 sec
		GA-based FCM	0.9817 ± 0.00021	39 sec
		Proposed	**0.9898 ± 2.34E-16**	**37 sec**
Q	6	FCM	13119.427 ± 7046.408	39 sec
		GA-based FCM	12018.495 ± 2149.968	25 sec
		Proposed	**8966.715 ± 2635.109**	**22 sec**
	8	FCM	47943.902 ± 29441.925	49 sec
		GA-based FCM	47417.776 ± 16557.246	47 sec
		Proposed	**43233.689 ± 7179.615**	**43 sec**

Table 1.4 Class boundaries and evaluated segmentation quality measures ρ by different algorithms for different classes of the peppers image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	**1**	**28, 76, 108, 140, 178, 201**	**0.9291**
		2	26, 75, 103, 144, 169, 191	0.9312
		3	23, 69, 98, 148, 171, 201	0.7860
		4	25, 71, 107, 139, 170, 204	0.8916
	GA-based FCM	1	**29, 76, 108, 144, 173, 201**	**0.9829**
		2	28, 75, 108, 144, 170, 201	0.9828
		3	29, 76, 108, 145, 173, 203	0.9828
		4	30, 76, 110, 144, 173, 201	0.9827
	Proposed	1	29, 78, 108, 144, 173, 203	0.9830
		2	28, 76, 110, 144, 172, 203	0.9830
		3	29, 78, 110, 146, 173, 202	0.9830
		4	**28, 75, 108, 144, 172, 201**	**0.9831**
8	FCM	1	26, 69, 94, 111, 136, 160, 181, 203	0.9268
		2	17, 46, 79, 106, 128, 156, 178, 203	0.9317
		3	**40, 79, 104, 121, 143, 160, 176, 199**	**0.9585**
		4	17, 46, 79, 106, 127, 155, 178, 202	0.9309
	GA-based FCM	1	27, 71, 98, 115, 146, 170, 193, 213	0.9887
		2	**17, 45, 79, 106, 128, 156, 178, 203**	**0.9890**
		3	27, 71, 98, 114, 146, 170, 193, 213	0.9887
		4	18, 47, 81, 108, 140, 191, 167, 211	0.9889
	Proposed	1	18, 47, 81, 108, 140, 167, 191, 211	0.9890
		2	27, 71, 98, 114, 146, 170, 193, 213	0.9890
		3	17, 46, 79, 106, 128, 156, 179, 203	0.9891
		4	**17, 46, 78, 106, 129, 156, 179, 203**	**0.9892**

deviation of both fitness values (ρ and Q) are best for the proposed algorithm compared to the same reported by the other two algorithms. It is also observed that the proposed method has taken less time than the other two approaches.

The same experiment is repeated for the peppers image. The ρ-based and Q-based class boundaries and the fitness values are reported in Table 1.4 and Table 1.5, respectively. The better results are highlighted in **bold**. From these tables, it is detected that the fitness values derived by the proposed algorithm are better than the same results obtained by the other two algorithms. This observation will remain unchanged for the peppers image whether it is segmented in six segments or eight segments.

The peppers image is segmented by the proposed approach 50 times for each fitness function. The mean and standard deviation of those different function-based fitness values are reported in Table 1.6. The mean of the time taken by the proposed algorithm is also tabulated in this table. The same results are also saved for the other two algorithms. The mean and standard deviation of the fitness values and the mean time taken by the other two algorithms are also reported in Table 1.6. The MfGA-based FCM algorithm is much better than the other two algorithms if anyone considers the accounted results in

Table 1.5 Class boundaries and evaluated segmentation quality measures *Q* by different algorithms for different classes of the peppers image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	1	28, 76, 108, 145, 173, 201	42696.252
		2	29, 76, 110, 144, 175, 202	37371.362
		3	27, 75, 105, 143, 170, 201	31326.604
		4	**26, 71, 108, 144, 173, 201**	**9813.211**
	GA-based FCM	1	28, 78, 106, 144, 169, 203	17130.923
		2	**29, 76, 108, 140, 173, 201**	**9130.923**
		3	26, 75, 109, 144, 174, 202	15065.869
		4	27, 76, 108, 144, 171, 203	14531.759
	Proposed	1	29, 76, 108, 144, 173, 201	5693.583
		2	29, 78, 110, 156, 174, 194	4907.482
		3	29, 76, 107, 144, 172, 201	5693.583
		4	**29, 76, 108, 145, 173, 201**	**4892.024**
8	FCM	1	18, 47, 81, 108, 140, 167, 191, 211	181883.131
		2	**27, 71, 98, 115, 146, 170, 193, 213**	**102247.881**
		3	27, 71, 98, 115, 146, 172, 198, 211	198840.268
		4	17, 46, 79, 106, 128, 156, 179, 203	283016.901
	GA-based FCM	1	27, 71, 98, 115, 146, 170, 193, 213	53784.935
		2	27, 76, 97, 118, 140, 167, 191, 211	44311.485
		3	28, 71, 98, 115, 146, 170, 193, 213	53908.007
		4	**27, 71, 98, 115, 147, 175, 193, 212**	**27230.946**
	Proposed	1	18, 47, 81, 108, 125, 142, 170, 199	20251.096
		2	17, 46, 79, 106, 128, 156, 178, 202	21406.658
		3	27, 71, 99, 115, 147, 171, 199, 213	24926.826
		4	**17, 46, 79, 106, 128, 156, 178, 202**	**19632.432**

Table 1.6 Different algorithm-based mean and standard deviation using different types of fitness functions and mean of time taken by different algorithms for the peppers image

Fitness function	# Segments	Algorithm	Mean ± standard deviation	Mean time
ρ	6	FCM	0.8890 ± 0.06096	36 sec
		GA-based FCM	0.9727 ± 2.686E-06	28 sec
		Proposed	**0.9831 ± 3.789E-07**	**24 sec**
	8	FCM	0.9472 ± 0.01815	58 sec
		GA-based FCM	0.9824 ± 0.00021	56 sec
		Proposed	**0.9890 ± 0.00016**	**54 sec**
Q	6	FCM	14739.441 ± 15846.203	30 sec
		GA-based FCM	14681.419 ± 2967.542	26 sec
		Proposed	**5534.818 ± 334.728**	**24 sec**
	8	FCM	111303.905 ± 91628.397	59 sec
		GA-based FCM	40479.148 ± 15736.887	56 sec
		Proposed	**32567.705 ± 15566.113**	**55 sec**

Table 1.7 Class boundaries and evaluated segmentation quality measures ρ by different algorithms for different classes of the baboon image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	**1**	**37, 70, 95, 121, 140, 173**	**0.9204**
		2	37, 69, 95, 118, 144, 175	0.9154
		3	37, 69, 89, 116, 142, 174	0.9099
		4	35, 68, 98, 118, 145, 170	0.9188
	GA-based FCM	**1**	**37, 69, 96, 119, 146, 173**	**0.9785**
		2	36, 69, 95, 118, 144, 175	0.9783
		3	37, 69, 95, 120, 144, 173	0.9784
		4	37, 68, 94, 118, 145, 175	0.9783
	Proposed	1	38, 70, 96, 120, 154, 173	0.9784
		2	43, 79, 110, 133, 158, 174	0.9785
		3	**37, 69, 98, 120, 145, 173**	**0.9786**
		4	38, 68, 96, 118, 144, 174	0.9785
8	FCM	**1**	**36, 66, 82, 101, 125, 141, 159, 180**	**0.9514**
		2	31, 60, 86, 96, 118, 130, 159, 179	0.9183
		3	29, 58, 81, 100, 116, 128, 161, 181	0.9204
		4	30, 59, 79, 99, 119, 138, 159, 178	0.9498
	GA-based FCM	1	32, 61, 81, 101, 118, 137, 160, 178	0.9869
		2	**32, 59, 81, 100, 119, 137, 159, 177**	**0.9870**
		3	33, 59, 82, 99, 118, 135, 158, 178	0.9869
		4	31, 60, 81, 101, 120, 137, 158, 177	0.9861
	Proposed	1	33, 59, 81, 99, 121, 137, 159, 178	0.9869
		2	32, 59, 81, 101, 117, 138, 159, 178	0.9870
		3	**33, 59, 81, 100, 118, 137, 158, 178**	**0.9871**
		4	32, 59, 81, 101, 118, 137, 159, 178	0.9870

Table 1.6. The fitness values and the execution time of the proposed algorithm are best compared to the same for the other two algorithms.

The class boundaries and the ρ-based fitness values for segmenting the baboon image are reported in Table 1.7. In Table 1.8, the class levels for segmenting the same image and the Q-based fitness values are presented. The best results derived by the individual algorithm are highlighted in **bold**.

The proposed approach is applied 50 times for segmenting the baboon image on the basis of two fitness functions separately. The mean and standard deviation of those different function-based fitness values are tabulated in Table 1.9. The mean of the time taken by the proposed algorithm is also reported in this table. The same thing is also done for the other two algorithms. The mean and standard deviation of the fitness values and the mean time taken by the other two algorithms are also accounted in Table 1.9. If anyone goes through the reported results in Table 1.9, they can observe that the other two algorithm-based results are far behind the results derived by the MfGA-based FCM algorithm.

The segmented grayscale output images obtained for the $K = \{6, 8\}$ classes, with the proposed approach vis-à-vis with the FCM [5] and the GA-based FCM [28] algorithms,

Table 1.8 Class boundaries and evaluated segmentation quality measures *Q* by different algorithms for different classes of the baboon image

# Segments	Algorithm	#	Class levels	Fitness value
6	FCM	1	38, 71, 95, 116, 140, 173	16413.570
		2	**37, 69, 95, 114, 134, 165**	**13422.376**
		3	36, 69, 92, 114, 141, 170	18223.109
		4	37, 68, 95, 112, 143, 176	16808.168
	GA-based FCM	1	37, 69, 98, 120, 144, 173	7933.554
		2	36, 69, 95, 119, 146, 173	15368.872
		3	**38, 71, 98, 118, 144, 174**	**5532.344**
		4	37, 73, 95, 118, 145, 173	15368.872
	Proposed	**1**	**38, 69, 95, 120, 146, 173**	**4376.556**
		2	38, 70, 94, 118, 144, 173	4571.008
		3	43, 79, 110, 133, 158, 174	5053.403
		4	37, 68, 94, 118, 144, 173	4985.565
8	FCM	1	32, 49, 80, 99, 118, 131, 159, 178	95888.688
		2	33, 62, 81, 101, 118, 137, 152, 178	73697.780
		3	32, 60, 81, 100, 113, 137, 159, 179	102621.172
		4	**29, 59, 81, 97, 118, 137, 159, 178**	**70320.873**
	GA-based FCM	**1**	**32, 59, 82, 101, 118, 137, 159, 178**	**27848.183**
		2	33, 60, 82, 102, 119, 138, 160, 179	31323.896
		3	34, 60, 82, 101, 118, 138, 159, 178	32329.115
		4	33, 59, 81, 101, 118, 137, 161, 178	28488.563
	Proposed	1	33, 61, 84, 104, 121, 147, 164, 175	21661.820
		2	32, 59, 80, 99, 117, 135, 156, 177	24137.965
		3	**34, 61, 84, 104, 121, 143, 160, 172**	**21198.127**
		4	33, 60, 81, 101, 118, 137, 158, 178	26805.747

Table 1.9 Different algorithm-based mean and standard deviation using different types of fitness functions and mean of time taken by different algorithms for the baboon image

Fitness function	# Segments	Algorithm	Mean ± standard deviation	Mean time
ρ	6	FCM	0.9211 ± 0.01114	25 sec
		GA-based FCM	0.9680± 8.198E-07	22 sec
		Proposed	**0.9788 ± 7.971E-07**	**14 sec**
	8	FCM	0.9489 ± 0.01751	59 sec
		GA-based FCM	0.9762 ± 2.135E-05	56 sec
		Proposed	**0.9870 ± 2.519E-05**	**54 sec**
Q	6	FCM	9777.519 ± 5695.687	42 sec
		GA-based FCM	9671.766 ± 4961.474	40 sec
		Proposed	**4894.895 ± 628.569**	**32 sec**
	8	FCM	52408.164 ± 30991.141	53 sec
		GA-based FCM	28137.835 ± 2594.486	51 sec
		Proposed	**24343.203 ± 8091.355**	**43 sec**

are demonstrated afterward. The segmented grayscale test images with $K = 8$ segments are presented in this chapter. The segmented multilevel grayscale test images derived by the FCM algorithm and the GA-based FCM algorithm are depicted in the first and second rows of each figure. In the third row of each figure, the segmented multilevel grayscale test images by the proposed MfGA-based FCM algorithm are shown. For easy recognition in each figure, the GA-based FCM and the MfGA-based FCM algorithms are noted as GA FCM and MfGA FCM, respectively. The segmented images by the FCM algorithm are presented in (a–d), the segmented images by the GA-based FCM algorithm are shown in (e–h), and the MfGA-based FCM-based multilevel segmented images are pictured in (i–l) of each figure, respectively. In Figure 1.2, the class levels (K=8) of Table 1.1, obtained by the proposed approach and other two algorithms, are employed to get the segmented output image of the Lena image. The fitness function, ρ, is applied in this case. Figure 1.3 is generated with the class levels obtained by each algorithm. These images are segmented into eight segments, and Q is applied as the evaluation function in this case.

It is observed from Figures 1.2 and 1.3 that the proposed approach gives better segmented outputs than the same derived by the other two approaches.

The multilevel segmented outputs of the peppers image are shown in Figures 1.4 and 1.5. The four class levels (K=8) of Table 1.4, obtained by the proposed algorithm as

Figure 1.2 8-class segmented 256×256 grayscale Lena image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.1, with ρ as the quality measure.

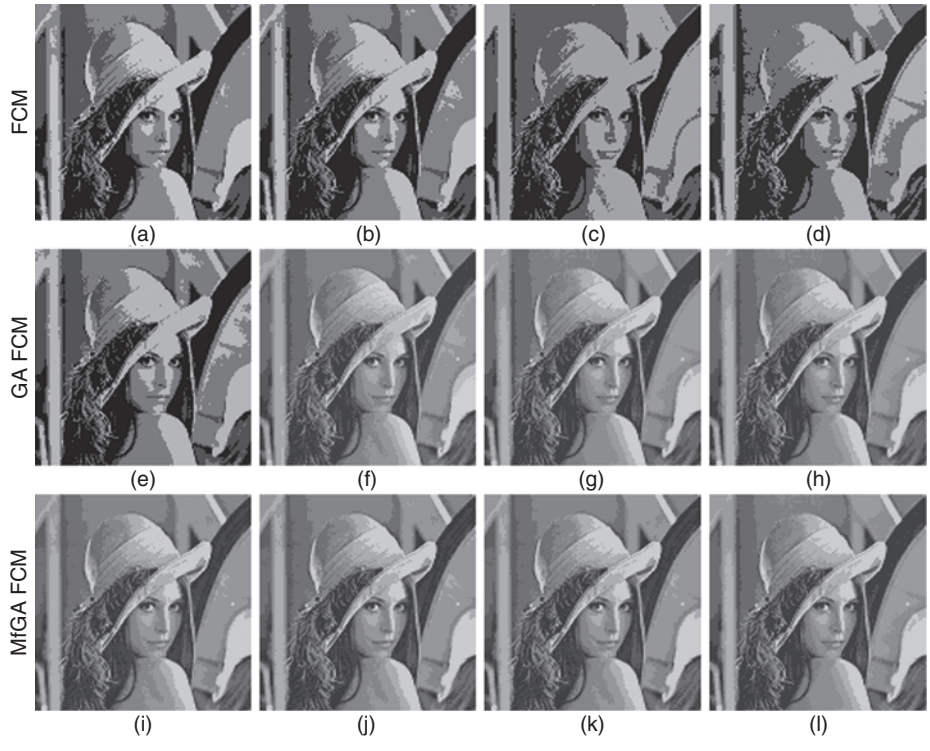

Figure 1.3 8-class segmented 256×256 grayscale Lena image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.2, with Q as the quality measure.

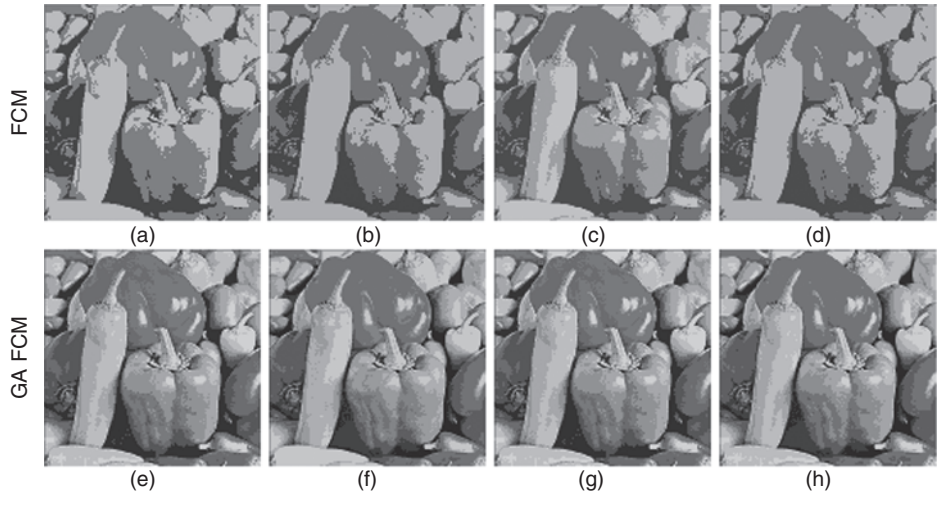

Figure 1.4 8-class segmented 256×256 grayscale peppers image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.4, with ρ as the quality measure.

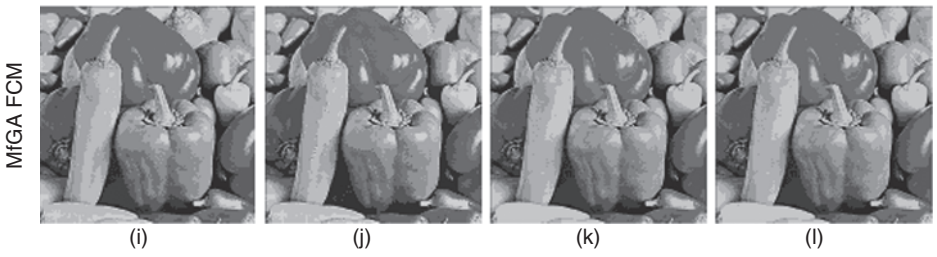

Figure 1.4 (*Continued*)

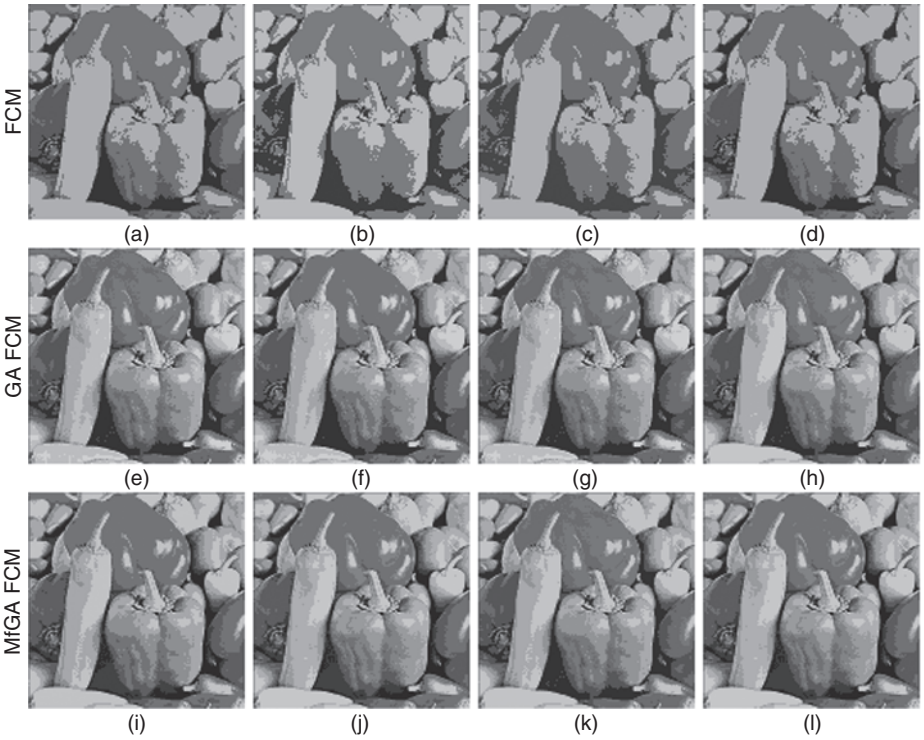

Figure 1.5 8-class segmented 256×256 grayscale peppers image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.5, with *Q* as the quality measure.

well as other two approaches, are employed to derive the segmented output images of the peppers image in Figure 1.4. In this case, ρ is employed as the evaluation function. In Figure 1.5, the multilevel segmented outputs of the peppers image are yielded using the Q fitness function based on four results from Table 1.5 with K=8 class levels. The multilevel segmented peppers images by the proposed MfGA-based FCM algorithm are segmented in a better way than the segmented images deduced by the other two approaches, and this is clear from Figures 1.4 and 1.5.

The ρ is applied as the fitness function to generate the segmented baboon image, which is shown in Figure 1.6 using K=8 class levels of Table 1.7. The class levels (K=8) of

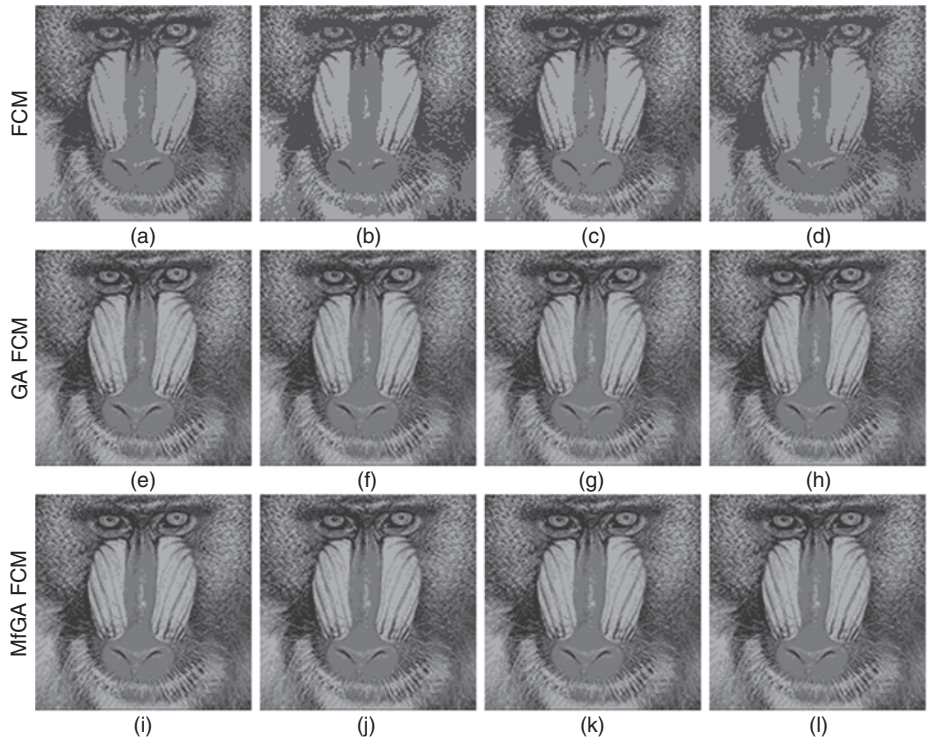

Figure 1.6 8-class segmented 256×256 grayscale baboon image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.7, with ρ as the quality measure.

Table 1.8 are employed to generate the segmented baboon image that is depicted in Figure 1.7. In this case, the empirical measure Q is applied as the quality measure.

From Figures 1.6 and 1.7, it can be said that the segmented multilevel baboon images are better segmented by the proposed algorithm than by the FCM and GA-based FCM algorithms.

At the end, it can be concluded that the proposed MfGA algorithm overwhelms the FCM [5] and GA-based FCM [28] algorithms quantitatively and qualitatively.

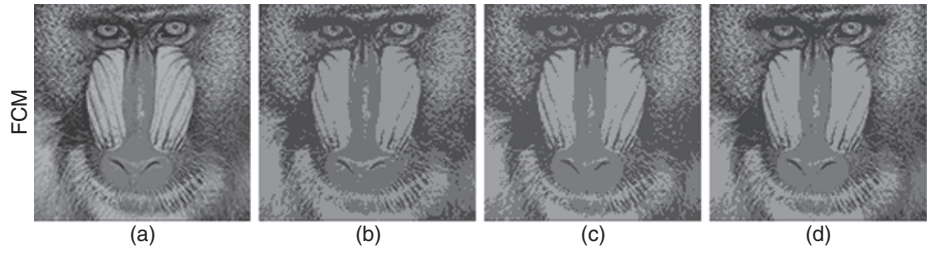

Figure 1.7 8-class segmented 256×256 grayscale baboon image with the class levels obtained by (a–d) FCM, (e–h) GA-based FCM, and (i–l) MfGA-based FCM algorithms of four results of Table 1.8, with Q as the quality measure.

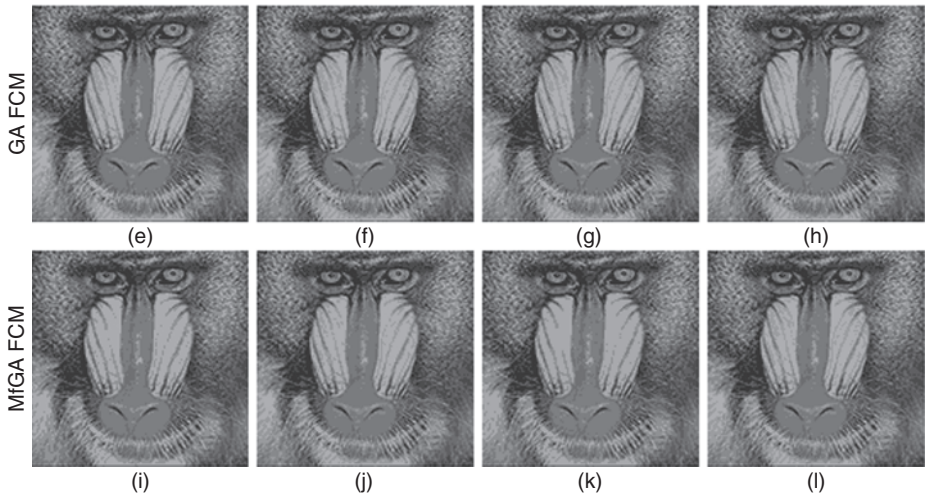

Figure 1.7 (*Continued*)

1.7 Conclusion

In this chapter, different types of multilevel grayscale image segmentation techniques are considered. In this regard, a modified version of the genetic algorithm (MfGA)-based FCM algorithm is proposed to segment the multilevel grayscale images. The FCM and the GA-based FCM algorithms are also recounted briefly, and they are also used to segment the same multilevel grayscale images. The drawback of the original FCM algorithm is also pointed out in this chapter quite efficiently. The way to get rid of the drawback of the FCM algorithm is also discussed elaborately by proposing the proposed algorithm. The solutions derived by the MfGA-based FCM algorithm are globally optimized solutions. To derive the optimized class levels in this procedure, different image segmentation quality measures are used. The performance of the proposed MfGA-based FCM algorithm for real-life multilevel grayscale image segmentation is superior in most of the cases as compared to the other two segmentation algorithms.

References

1 Zaitoun, N.M. and Aqel, M.J. (2015) Survey on image segmentation techniques, in *Proceedings of International Conference on Communications, management, and Information technology (ICCMIT'2015)*, vol. 65, Procedia Computer Science, vol. 65, pp. 797–806.

2 Khan, W. (2013) Image segmentation techniques: a survey. *Journal of Image and Graphics*, **1** (4), 166–170.

3 Chauhan, A.S., Silakari, S., and Dixit, M. (2014) Image segmentation methods: a survey approach, in *Proceedings of 2014 Fourth International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 929–933.

4 MacQueen, J. (1967) Some methods for classification and analysis of multivariate observations, in *Fifth Berkeley Symposium on Mathematics, Statistics and Probability*, pp. 281–297.

5 Bezdek, J. (1981) *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York.

6 Luo, M., Ma, Y.F., and Zhang, H.J. (2003) A spatial constrained K-means approach to image segmentation, in *Proceedings of the 2003 Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia*, vol. 2, pp. 738–742.

7 Khan, S.S. and Ahamed, A. (2004) Cluster center initialization algorithm for K-means clustering. *Pattern Recognition Letters*, **25** (11), 1293–1302.

8 Barakbah, A.R. and Kiyoki, Y. (2009) A new approach for image segmentation using Pillar-kmeans algorithm. *World Academy of Science, Engineering and Technology*, **59**, 23–28.

9 Ng, H.P., Ong, S.H., Foong, K.W.C., Goh, P.S., and Nowinski, W.L. (2006) Medical image segmentation using k-means clustering and improved watershed algorithm, in *2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 61–65.

10 Ahmed, M.N., Yamany, S.M., Mohamed, N., Farag, A.A., and Moriarty, T. (2002) A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data. *IEEE Transactions on Medical Imaging*, **21** (3), 193–199.

11 De, S., Bhattacharyya, S., and Dutta, P. (2010) Efficient grey-level image segmentation using an optimised MUSIG (OptiMUSIG) activation function. *International Journal of Parallel, Emergent and Distributed Systems*, **26** (1), 1–39.

12 Tripathy, B.K., Basu, A., and Govel, S. (2014) Image segmentation using spatial intuitionistic fuzzy C means clustering, in *2014 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–5.

13 Balafar, M.A., Ramli, A.R., Saripan, M.I., Mahmud, R., Mashohor, S., and Balafar, M. (2008) New multi-scale medical image segmentation based on fuzzy c-mean (FCM), in *2008 IEEE Conference on Innovative Technologies in Intelligent Systems and Industrial Applications*, pp. 66–70.

14 Noordam, J.C., van den Broek, W.H.A.M., and Buydens, L.M.C. (2000) Geometrically guided fuzzy c-means clustering for multivariate image segmentation, in *Proceedings of 15th International Conference on Pattern Recognition*, pp. 462–465.

15 Pham, D.L. and Prince, J.L. (1999) Adaptive fuzzy segmentation of magnetic resonance images. *IEEE Transastions on Medical Imaging*, **18** (9), 737–752.

16 Chen, L., Cui, B., Han, Y., Guan, Y., and Luo, Y. (2015) Two-dimensional fuzzy clustering algorithm (2DFCM) for metallographic image segmentation based on spatial information, in *2nd International Conference on Information Science and Control Engineering*, pp. 519–521.

17 Hruschka, E.R., Campello, R.J.G.B., Freitas, A.A., and Leon, P. (2009) A survey of evolutionary algorithms for clustering. *IEEE Transactions on System Man Cybernatics, C: Applcations and Reviews*, **39**, 133–155.

18 Ripon, K.S.N., Tsang, C.H., and Kwong, S. (2006) Multi-objective data clustering using variable-length real jumping genes genetic algorithm and local search method, in *IEEE International Joint Conference on Neural Networks*, pp. 3609–3616.

19 Goldberg, D.E. (1989) *Genetic Algorithms: Search, Optimization and Machine Learning*, Addison-Wesley, New York.

20 Hammouche, K., Diaf, M., and Siarry, P. (2008) A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation. *Computer Vision and Image Understanding*, **109** (2), 163–175.

21 Fu, Z., He, J.F., Cui, R., Xiang, Y., Yi, S.L., Cao, S.J., Bao, Y.Y., Du, K.K., Zhang, H., and Ren, J.X. (2015) Image segmentation with multilevel threshold of gray-level & gradient-magnitude entropy based on genetic algorithm, in *International Conference on Artificial Intelligence and Industrial Engineering*, pp. 539–542.

22 Hauschild, M., Bhatia, S., and Pelikan, M. (2012) Image segmentation using a genetic algorithm and hierarchical local search, in *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation*, pp. 633–640.

23 De, S., Bhattacharyya, S., and Dutta, P. (2009) Multilevel image segmentation using OptiMUSIG activation function with fixed and variable thresholding: a comparative study, in *Applications of Soft Computing: From Theory to Praxis, Advances in Intelligent and Soft Computing* (J. Mehnen, M. Koppen, A. Saad, and A. Tiwari), Springer-Verlag, Berlin, pp. 53–62.

24 Min, W. and Siqing, Y. (2010) Improved k-means clustering based on genetic algorithm, in *International Conference on Computer Application and System Modeling (ICCASM)*, pp. 636–639.

25 Biju, V.G. and Mythili, P. (2012) A genetic algorithm based fuzzy c-mean clustering model for segmenting microarray images. *International Journal of Computer Applications*, **52** (11), 42–48.

26 Wei, C., Tingjin, L., Jizheng, W., and Yanqing, Z. (2010) An improved genetic FCM clustering algorithm, in *2nd International Conference on Future Computer and Communication*, vol. 1, pp. 45–48.

27 Halder, A., Pramanik, S., and Kar, A. (2011) Dynamic image segmentation using fuzzy c-means based genetic algorithm. *International Journal of Computer Applications*, **28** (6), 15–20.

28 Jansi, S. and Subashini, P. (2014) Modified FCM using genetic algorithm for segmentation of MRI brain images, in *IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–5.

29 Borsotti, M., Campadelli, P., and Schettini, R. (1998) Quantitative evaluation of color image segmentation results. *Pattern Recognition Letters*, **19** (8), 741–747.

30 Ruspini, E. (1969) A new approach to clustering. *Information and Control*, **15**, 22–32.

31 Zhang, H., Fritts, J., and Goldman, S. (2004) An entropy-based objective evaluation method for image segmentation, in *Proceedings of SPIE Storage and Retrieval Methods and Applications for Multimedia*, pp. 38–49.