

A Novel Approach Based on Genetic Algorithms and Region Growing for Magnetic Resonance Image (MRI) Segmentation

Elnomery A. Zanyat^{1,2} and Ahmed S. Ghiduk^{1,3}

¹College of Computers and IT, Taif University,
Taif, Saudi Arabia
{n.allam, asaghiduk}@tu.edu.sa

²Department of Mathematics, Faculty of Science,
Sohag University, Egypt

³Department of Mathematics, Faculty of Science,
Beni-Suef University, Egypt

Abstract. This paper presents a new segmentation approach based on hybridization of the genetic algorithms (GAs) and seed region growing to produce accurate medical image segmentation, and to overcome the oversegmentation problem. A new fitness function is presented for generating global minima of the objective function, and a chromosome representation suitable for the process of segmentation is proposed. The proposed approach starts by selecting a set of data randomly distributed all over the image as initial population. Each chromosome contains three parts: control genes, gray-levels genes, and position genes. Each gene associates the intensity values by their positions. The region growing algorithm uses these values as an initial seeds to find accurate regions for each control gene. The proposed fitness function is used to evolve the population to find the best region for each control gene. Chromosomes are updated by applying the operators of GAs to evolve segmentation results. Applying the proposed approach to real MRI datasets, better results were achieved compared with the clustering-based fuzzy method.

Keywords: Image segmentation, genetic algorithms, region growing method, fuzzy c-means.

1. Introduction

Medical Image segmentation is an important tool in analyzing magnetic resonance (MR) images and solving a wide range of problems in medical imaging. Accurate segmentation of MR images represents a key problem in this area [1-3]. Genetic algorithms (GAs) are powerful methods in medical image segmentation [4-7, 15, 17, 18]. Many techniques that successfully apply genetic algorithms to image segmentation have been developed in [8-18]. Farmer and Shugars [8] integrated segmentation and classification in a

manner analogous to wrapper methods in feature selection. They initially performed low-level segmentation to label the image as a set of non-overlapping blobs and then use the wrapper framework to select the blobs that comprise the final segmentation based on the classification performance of the wrapper. Yoshimura and Oe [9] combined GA and Kohonen's self-organizing map for clustering of textured images. Andrey [10] suggested an original approach as no objective fitness function is needed to evaluate segmentation results. Li and Chiao [11] proposed a genetic algorithm dedicated to texture images where the fitness function is based on texture features similarity. Melkemi et al. [12] used genetic algorithms to combine different segmentation results obtained by different agents. Lai and Chang [13] presented hierarchical evolutionary algorithms (HEA) for medical image segmentation that can classify the image into appropriate classes and avoid the difficulty of searching for the proper number of classes. Karteeka et al. [14] studied medical image segmentation and attempted to extract the shape of the tissues in medical images automatically using automatic clustering using differential evolution. A method based on hybrid GA and active contour was presented to solve some of active contour problems for accurate medical ultrasound image segmentation [15]. Wang et al. [16] combined GA and fuzzy clustering in which the genetic algorithm is adopted to optimize the initial cluster center and then the fuzzy clustering is used for image segmentation. Wang et al. [17] combined GA and fuzzy clustering in which the genetic algorithm is adopted to optimize the initial cluster center and then the fuzzy clustering is used for image segmentation. Maulik [18] reviews the major applications of GAs to the domain of medical image segmentation.

Although the hybridization between GA and fuzzy clustering in [17] or active contour in [16] was presented to avoid the drawbacks of the fuzzy clustering and active contour respectively, these methods suffer oversegmentation problem. The main disadvantage of these methods is the difficulty of searching for the proper number of classes.

To avoid the drawbacks of the previous work, this paper introduces a new combination between GA and seed region growing to overcome the oversegmentation problem, improve the quality of image segmentation, and accelerate the search for finding the optima. The region growing is used to extract a complete region according to chromosome information to avoid the oversegmentation. The proposed approach consists of four steps: finding the initial population, performing seed region growing, evaluating fitness function for each chromosome, and evolving the chromosomes. The initial population is randomly generated by uniform discrete image sampling. The proposed GA attempts to find out the optimal centroid for each region for fine segmentation. The chromosome representation includes control genes, gray-levels genes, and x and y -axis values of the gray level. The gray-level genes which have control-genes equal to one and located at (x, y) are centroids of the clusters. Then, the initial population is passed to the seed region growing with initial seed (with location (x, y)). The fitness function is improved by considering the covered and uncovered data for quantitative measure of segmentation result.

The proposed method is experimented using different MRI images with weak boundaries to prove its efficiency and applicability. The experimental results show that the proposed technique produces more accurate and stable results compared with other segmentation techniques such as fuzzy c-means (FCM) [19] and hybrid GA and fuzzy clustering (GAFCM) [17] methods.

From the above discussion, the main contributions of this paper are:

- Introducing a new hybrid method based on genetic algorithms and seed region growing for medical image segmentation.
- Establishing a new fitness function for evaluating the generated segments of the image.
- Presenting a new chromosome representation to appropriate the image segmentation.
- Introducing an experimental study to evaluate the proposed method using different MRI images and comparing these results to other segmentation methods.

The rest of the paper is organized as follows. The MRI segmentation problem is discussed in section 2. Section 3 presents region growing technique. The proposed method is described in section 4. The experimental and comparative results are discussed in Section 5. Section 6 gives the related work. The conclusion is presented in section 7.

2. The MRI segmentation problem

The basic idea of image segmentation can be described as follows. Suppose that $X = \{x_1, x_2, \dots, x_n\}$ is a given set of data and P is a uniformity set of predicates. We aim to obtain a partition of the data into disjoint nonempty groups $X = \{v_1, v_2, \dots, v_k\}$ subject to the following conditions:

$$\bigcup_{i=1}^k v_i = X$$

$$v_i \cap v_j = \emptyset, i \neq j$$

$$P(v_i) = \text{TRUE}, i = 1, 2, \dots, k$$

$$P(v_i \cup v_j) = \text{FALSE}, i = j$$

The first condition ensures that every data value must be assigned to a group, while the second condition ensures that a data value can be assigned to only one group. The third and fourth conditions imply that every data value in one group must satisfy the uniformity predicate while data values from two different groups must fail the uniformity criterion.

Our study is related to 3D-model from MRI and simulated brain database of McGill University [20]. MRI has several advantages over other imaging techniques enabling it to provide 3-dimensional data with high contrast between soft tissues. However, the amount of data is far too much for manual analysis/interpretation, and this has been one of the biggest obstacles in the

effective use of MRI. Segmentation of MR images into different tissue classes, especially gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF), is an important task.

MR image segmentation involves the separation of image pixels into regions comprising different tissue types. All MR images are affected by random noise. The noise comes from the stray current in the detector coil due to the fluctuating magnetic fields arising from random ionic currents in the body, or the thermal fluctuations in the detector coil itself, more discussion can be seen [21]. When the level of noise is significant in an MR image, tissues that are similar in contrast could not be delineated effectively, causing error in tissue segmentation. Then more sophisticated techniques would be needed to reconstruct the 3D image from incomplete information [22-25], where a 3D image can be obtained from many consecutive 2D slices.

3. Seed region growing

The basis of the method is to segment an image of N pixels into regions with respect to a set of seeds [26] using only the initial seed pixels. The initial seed pixel is selected from a pixel with mask 3×3 . These seeds are grown by merging neighboring pixels whose properties are most similar to the premerged region. Typically, the homogeneity criterion is defined as the difference between the intensity of the candidate pixel and the average intensity of the premerged region. If the homogeneity criterion (threshold T) is satisfied, the candidate pixel will be merged to the premerged region. The procedure is iterative: at each step, a pixel is merged according to the homogeneity criterion (under threshold T). This process is repeated until no more pixels are assigned to the region [27]. Since we only perform the seed growing on edge pixels, the amount of data needed to be processed is much reduced, resulting in increased speed. The algorithm complexity is proposed to be $O(N \cdot h \cdot \log(c \cdot N))$, where h is the average number of distances to be recalculated on each step and c is the image connectivity.

4. The proposed genetic algorithm

GAs [19, 28, 29] are efficient, adaptive, and robust search and optimization techniques guided by the principles of evolution and natural genetics, and have implicit parallelism. The essential components of GAs are the following: 1) a representation strategy called chromosomes; 2) a population of chromosomes encode candidate solutions to the optimization problem; 3) a mechanism for evaluating each chromosome (fitness function); 4) selection/reproduction procedure; and 5) genetic operators (crossover, mutation, and elitism). In the proposed algorithm, chromosomes are represented in binary strings of 0s and 1s and evolved toward better solutions. A chromosome consists of four parts: control genes, gray-levels

genes, and position genes to serve our problem. The evolution usually starts from a population of randomly generated individuals. The region growing algorithm is used to extract accurate regions for each control gene value to reduce the search space in the whole gray levels image. In each generation, a fitness function is used to evaluate individuals, and reproductive success varies with fitness. The goal of a fitness function is to provide a meaningful, measurable, and comparable value given a set of genes. If the fitness test takes a long time to perform, then the GA may take a long time to execute. Here, the fitness function is improved in order to reduce the repeated tests.

4.1. Fitness function

In general, an image can be described by a two-dimensional function $f(x, y)$, where (x, y) denotes the spatial coordinates, and the intensity value at (x, y) is $f(x, y) \in [0, 255]$.

In a GA, a population of individuals, described by some chromosomes, is iteratively updated by applying operators of selection, mutation and crossover to solve the problem. Each individual is evaluated by a fitness function that controls the population evolution in order to optimize it. The most important components of our proposed method concern both the modeling of the problem with GA and the definition of the fitness function. Here, we present the most fitness functions that give almost reliable decision.

In Andrey [9], the image was segmented using selectionist relaxation (SR). The image site s is mapped in the feature space to a point $V_s = \{v_1, v_2, \dots, v_n\}$ which is the concatenated set of gray level values in the $I = k \times k$ image window-centered on s . A unit is a real-coded chromosome of length p that represents a point in the feature space. A population of units is built by assigning a unit to each site of the image. At site s , the fitness fit of the associated unit $U_s = \{u_1, u_2, \dots, u_p\}$ is computed as the opposite of the city-block distance between U_s and V_s , so that a unit whose coordinates (x, y) are close to the actual local pixels values will have a high fitness:

$$fit_s = - \sum_{i=1}^I |u_i - v_i| \quad (1)$$

In a hierarchical method that was presented in [12], the image $f(x, y)$ was subdivided into $\{v_1, v_2, \dots, v_k\}$ with each v_i being a connected region of $f(x, y)$, such that:

$$fit = n \sum_{i=1}^n \sum_{j=1}^k |Re(p(v_i)) - c_j| \quad (2)$$

where $Re(p(v_i))$ denotes the representative gray level of some region v_i and K is the number of regions.

The above fitness function fails to find the optima, because it evaluates the distances between the region centre and points without considering the uncovered data.

In order to solve the shortcomings of previous fitness function, we present a new fitness function that works under constraints for cover and uncover data. The fitness function includes two terms: covers and uncovers penalty. The point x called covered point if $x \in S_j, S_j$ is a region that contains connected points around the centre c_j , and x called uncovered if $x \notin \cup_j S_j$.

$$fitness = \alpha \sum_{i=1}^n \sum_{j=1}^K \|C_j - R_j(x_i, y_i)\|^2 + NCR \quad (3)$$

where NCR is a penalty for uncover points.

The Euclidian distance term represents the shortest distances between the centroid $c_j, j=1,2,...,k$ and all pixels $p_i, i=1,2,...,N$ of a region R_j . A pixel is selected as a member of region if Euclidian distance of this point is smaller than a prescribed value. Thus, some pixels are still uncovered by regions. NCR is used to represent these pixels. The minimum of the first term is obtained when all pixels fall in regions with the centre c_j . Otherwise the centre of uncovered pixels NCR is obtained by computing the median value of pixels. This value is considered as the centre of uncover pixels. NCR can be written as:

$$NCR = \alpha \sum_{i=1}^m \|Median - R(x_i, y_i)\|^2 \quad (4)$$

where m is the number of uncovered points. We provide optimal solutions of fitness function in a limited time by minimizing the value of NCR term.

In general, a GA contains a fixed-size population of potential solutions over the search space. These potential solutions of the search space are encoded as binary or floating-point strings, and called individuals or chromosomes.

This formulation can capture the intuition of segmenting an image; it is difficult to solve due to two reasons: first, the proper number of regions is not known beforehand. According to Felzenszwalb and Huttenlocher [30], the segmented regions are neither too fine nor too coarse. The second difficulty is that the decision of the representative gray level implicitly requires considering all the possible partitions. The fuzzy methods are most popular in this direction. The final number of clusters of these methods is still always sensitive to one or two user-selected parameters that define the threshold criterion for merging. Therefore, these drawbacks will reduce the clustering performance in real applications. In other hand, the seed region growing algorithm, one of the most widely used methods, can be applied to the image segmentation problem. However, it suffers from a couple of drawbacks, the seed region growing algorithm is very sensitive to the image structure, as well as the initial seeds selection. To apply this method, the initial seeds in the given data set should be known in advance. In order to tackle these problems,

we combine a GA algorithm and seed region growing algorithm, for automatically searching the number of regions as well as properly finding the representative gray level for each region.

4.2. The proposed algorithm

Fig. (1) gives the overall diagram of the proposed algorithm. From this figure, we can see that the input of our algorithm is only the target image. The outputs of the algorithm are the set of centroids and the set of regions. There is a set of parameters which are tuned by the user. These parameters are the population size (i.e., number of chromosomes (popSize)), the maximum of generations (maxGen), the chromosome length (chromLen), the mutation rate (Pm) and the crossover rate (Px) and the radius of the region T.

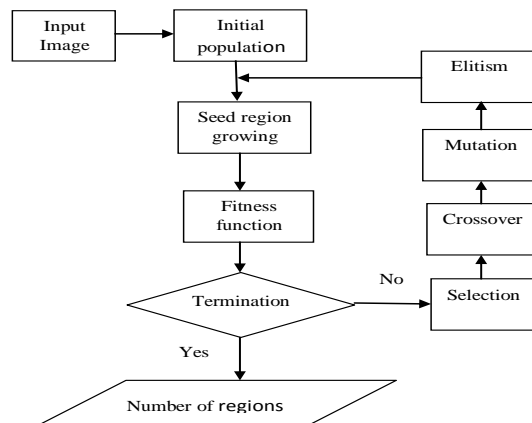


Fig. 1. The flowchart of the proposed algorithm

The different steps of our proposed algorithm which are shown in Fig. (1) can be stated as follows:

Algorithm: image segmentation.

Input: image to be segmented.

Tune: the parameters popSize, maxGen, chromLen, Pm and Px.

Outputs: the set of centroids and the set of regions.

Begin:

Step 1: Read the image.

Step 2: Find the initial population

for i = 1 to popsize

Generate randomly chromLen of binary digits to be Control Genes of the current chromosome.

Select randomly chromLen of pixels to be Gray-levels Genes of the current chromosome.

Get the X-Axis values and Y-Axis values and add them to the current chromosome.

end for

Step 3: Evaluate the initial population

for i = 1 to popsize

Find the Control Genes of the current chromosome with value "1" and the corresponding gray level value to be the centroids the regions.

For each centroid apply the seed region growing to get the region.

Find the uncovered pixels.

Apply the fitness function to evaluate the current chromosome.

end for

Let current population is the initial population.

Step 4: Repeat the following steps while number of generation is less than maxGen

Keep the best chromosome of the current population and its fitness.

Select the mating pool (the parents of the next population) using Tournament algorithm.

Crossover the mating pool to generate the offsprings.

Mutation the offsprings to generate the newpopulation.

Evaluate the new population (offsrings) as in step 3.

Keep the best chromosome of the new population and its fitness.

Apply the Elitism.

Let current population is the new population.

Step 5: Output the following:

The best chromosome of the current population and its fitness.

The set of centroids.

The set of regions.

End of the algorithm.

Chromosome Representation. In our approach, we divide the chromosome into four parts. The first part consists of series of binary digits (the total number of "1" implicitly represents the number of regions) [13]. The second part consists of integer numbers (representing the pixels value of discrete image). The third part contains the x-axis values of the pixels. Finally, the forth part contains the y-axis values of the gray level (i.e., the third and forth parts represent the position of the pixels in the image). The number of control genes is decided by soft estimating the upper bound of the number of regions. We explain the different components of GAs by using an illustrative example in Table 1. We consider a discrete image of 6x12 containing 72 different integer values as shown in Table 1. The chromosome structure in our approach is illustrated in Table 2. In this chromosome, there are three control genes having the value "1". Therefore, there are three regions in this image. The centroids of these regions are the gray level 1, 4, and 8 which are located at (5, 3), (3, 4), and (5, 2), respectively.

Table 1. Gray image as an example.

1	2	6	5	6	10	1	2	6	5	6	10
4	6	6	6	4	7	4	6	6	6	4	7
3	6	7	4	4	1	3	6	7	4	4	1
20	10	9	8	8	20	20	10	9	8	8	20
5	8	1	5	6	20	5	8	1	5	6	20
20	2	3	7	6	8	20	2	3	7	6	8

Table 2. Extended Hierarchical Chromosome.

Control Genes	Gray-levels Genes	X-Axis values	Y-Axis values
1, 0, 1, 0, 1	1, 6, 4, 20, 8	5, 1, 3, 5, 5	3, 5, 4, 6, 2

Initial Population. A GA requires a population of potential solutions to be initialized at the beginning of the GA process. In our approach, we randomly select a set of gray levels from the image as the initial parametric genes and their x and y axis values. As for the control genes, they are generated randomly from the set (0, 1). For example, Table 3 shows an initial population of four individuals.

Table 3. Initial Population.

#	Control Genes	Gray-levels Genes	X-Axis values	Y-Axis values
1	1, 0, 1, 0, 1	1, 6, 4, 20, 8	5, 1, 3, 5, 5	3, 5, 4, 6, 2
2	0, 1, 1, 0, 1	6, 5, 2, 6, 6	1, 5, 1, 2, 2	3, 1, 2, 4, 2
3	0, 1, 0, 1, 0	1, 10, 20, 6, 4	3, 4, 5, 1, 3	6, 2, 6, 5, 5
4	1, 1, 0, 0, 0,	4, 6, 6, 1, 7	2, 2, 1, 3, 6	5, 4, 5, 6, 4

According to this initial population, the first chromosome consists of three centers 1, 4, and 8, and three centers 5, 2, and 6 for the second chromosome. The third chromosome contains two centers 10 and 6 and two centers 4 and 6 for the fourth.

Evaluation Technique. To prove the efficiency of our proposed fitness function, we apply it on the discrete image in Table 1 with $\alpha = 1$ as follow.

We first apply the seed region growing to identify a region for each center in the chromosome, after finding the region of uncovered pixels. Table 4 shows two regions with centers 1 and 4 respectively. Table 5 shows the third region with center 8 and the region of uncovered pixels.

Table 4. Two regions for centers 1 and 4 in the first chromosome.

1	2	6	5	6	10	1	2	6	5	6	10
4	6	6	6	4	7	4	6	6	6	4	7
3	6	7	4	4	1	3	6	7	4	4	1
20	10	9	8	8	20	20	10	9	8	8	20
5	8	1	5	6	20	5	8	1	5	6	20
20	2	3	7	6	8	20	2	3	7	6	8
Segment 1						Segment 2					

Table 5. The region for center 8 and the uncovered region.

1	2	6	5	6	10	1	2	6	5	6	10
4	6	6	6	4	7	4	6	6	6	4	7
3	6	7	4	4	1	3	6	7	4	4	1
20	10	9	8	8	20	20	10	9	8	8	20
5	8	1	5	6	20	5	8	1	5	6	20
20	2	3	7	6	8	20	2	3	7	6	8
Segment 3						Uncovered segment					

According to the proposed fitness function without the penalty term, the values of the fitness function for the four chromosomes are 142, 152, 82, and 126 of the first, second, third, and fourth chromosomes respectively. In this situation, the third chromosome is the best one.

By applying the proposed fitness function with the penalty term; we noted that the values of the fitness function of the four chromosomes are 142, 714, 1025, and 882 of the first, second, third, and fourth chromosomes respectively as shown in Table 6. We noted that the first region is the best region according to the value 142 of the proposed fitness function. But according to the existing fitness function, third region is the best. We noted that the computation of the fitness values is enhanced and identified in more accurately the best chromosome.

Table 6. The fitness values of all chromosomes.

Chromosome#	1	2	3	4
Fitness values before penalty	142	152	82	126
Fitness values after penalty	142	714	1025	882

Selection. The selection/reproduction process copies individual strings into a tentative new population, the mating pool, for genetic operations. The number

of copies individually received in the next generation is usually taken to be directly proportional to its fitness value; thereby mimicking the natural selection procedure. This scheme is commonly called the proportional selection scheme. Roulette wheel parent selection, stochastic universal selection, and binary tournament selection [19, 28] are some of the most frequently used selection procedures. In the commonly used elitist model of GAs, the best chromosome seen up to the last generation is retained either in the population, or in a location outside it. In our approach, we adopt the tournament selection method [31] because the time complexity of it is low. The basic concept of the tournament method is as follows: randomly select a positive number of chromosomes from the population and copy the best fitted item from them into an intermediate population. The process is repeated P times, and here P is the population size. Table 7 shows the selected chromosome according to tournament method. This set will be the parents of the next generation. The crossover operation will apply on this set of chromosomes. The algorithm of tournament selection is shown as follows:

Algorithm: Tournament selection

Input: Population P (size of P is $popSize$), tournament size N_{tour} (a positive number)

Output: Population after selection P_0 (size of P_0 is also $popSize$)

begin

 for $i = 1$ to $popSize$ do

P_0 best fitted item among N_{tour} elements randomly selected from P ;

 return P_0

end

Table 7. Selected Parents.

#	Control Genes	Gray-levels Genes	X-Axis values	Y-Axis values
4	1, 1, 0, 0, 0,	4, 6, 6, 1, 7	2, 2, 1, 3, 6	5, 4, 5, 6, 4
2	0, 1, 1, 0, 1	6, 5, 2, 6, 6	1, 5, 1, 2, 2	3, 1, 2, 4, 2
2	0, 1, 1, 0, 1	6, 5, 2, 6, 6	1, 5, 1, 2, 2	3, 1, 2, 4, 2
2	0, 1, 1, 0, 1	6, 5, 2, 6, 6	1, 5, 1, 2, 2	3, 1, 2, 4, 2

Crossover. The main purpose of crossover is to exchange information between randomly selected parent chromosomes by recombining parts of their genetic information. The common crossover techniques are two-point crossover, multiple-point crossover, shuffle-exchange crossover, and uniform crossover [32]. The successful operation of GAs depends a lot on the coding technique used to represent the problem variables. The building block hypothesis indicates that GAs work by identifying good building blocks, and by eventually combining them to get larger building blocks [19]. Unless good building blocks are coded tightly, the crossover operation cannot combine them together. Thus coding–crossover interaction is important for the successful operation of GAs.

The crossover operator randomly pairs chromosomes and swaps parts of their genetic information to produce new chromosomes. For that, we use the uniform crossover [31]. The uniform crossover is applied to the control genes as well as the parametric genes, simultaneously. Two chromosomes are randomly selected as parents from the current population (in Table 7). The crossover creates the offspring chromosome on a bitwise basis, copying each allele from each parent with a probability p_i . The p_i is a random real number uniformly distributed in the interval $[0, 1]$. Let P_1 and P_2 be two parents, and C_1 and C_2 are offspring chromosomes; the i^{th} allele in each offspring is defined as:

$$\begin{aligned} C_1(i) &= P_1(i) \text{ and } C_2(i) = P_2(i) \text{ if } p_i \geq 0.5; \\ C_1(i) &= P_2(i) \text{ and } C_2(i) = P_1(i) \text{ if } p_i < 0.5 \end{aligned}$$

The illustrative example is processed using this operator and the result is presented in Table 8.

Table 8. Selected Parents.

#	Control Genes	Gray-levels Genes	X-Axis values	Y-Axis values
P_1	1, 1, 0, 0, 0,	4, 6, 6, 1, 7	2, 2, 1, 3, 6	5, 4, 5, 6, 4
P_2	0, 1, 1, 0, 1	6, 5, 2, 6, 6	1, 5, 1, 2, 2	3, 1, 2, 4, 2
P_i		0.3, 0.5, 0.25, 0.8, 0.3		
C_1	0, 1, 1, 0, 1	6, 6, 2, 1, 6	1, 2, 1, 3, 2	3, 4, 2, 6, 2
C_2	1, 1, 0, 0, 0	4, 5, 6, 6, 7	2, 5, 1, 2, 6	5, 1, 5, 4, 4

Mutation. Mutation is the process by which a random alteration in the genetic structure of a chromosome takes place. The main objective is to introduce genetic diversity into the population. Mutating a binary gene involves simple negation of the bit, while that for real coded genes is defined in a variety of ways [28].

The mutation operator is needed to explore new areas of the search space and helps the search procedure avoid sticking in local optima. Here, we apply bit mutation to the control genes. These results in some bits in control genes of the children being reversed: “1” is changed to “0” and “0” is changed to “1”. Either of these cases will change the number of regions. In the former, the associated parametric genes are disabled, while in the latter, the associated parametric genes are activated and the gene values are modified by randomly selecting a new gray level of the image.

Elitism. This step keeps the best chromosome from destroying. In this step, if the best chromosome of the previous population is fitter than the best chromosome of the current population then we exchange them. Then we replace the worst chromosome of the current population with the best chromosome of the previous population. If the best chromosome of the previous population is not fitter than the best chromosome of the current

population then we replace the worst chromosome of the current population with the best chromosome of the previous population.

5. Experimental and comparative results

The experiments were performed on several data sets, original image with slice thickness of 1mm, no intensity inhomogeneities, and corrupted by 6% salt and pepper noise. The original images size are 129×129 pixels, as shown in Figs. 2(a,b,c) obtained from the classical simulated brain database of McGill University (Brain Web [20]). The quality of the segmentation algorithm is of vital importance to the segmentation process. The comparison score S for each algorithm as proposed in [2] is defined as follows:

$$S = \frac{|A \cap A_{ref}|}{|A \cup A_{ref}|} \quad (5)$$

where A represents the set of pixels belonging to a class as found by a particular method and A_{ref} represents the reference cluster pixels.

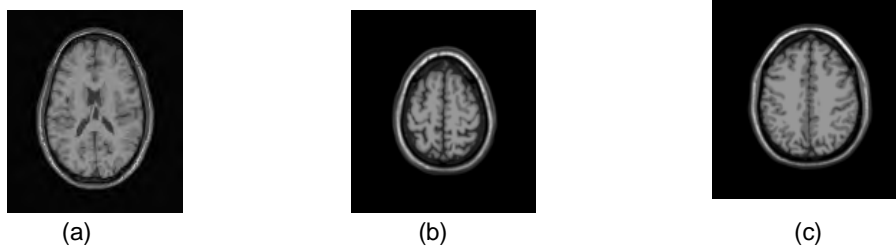
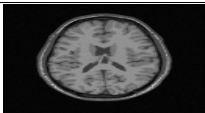
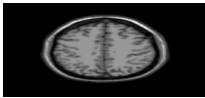



Fig. 2. Test images: (a), (b), and (c) two original slices from the 3D simulated data [24] (slices 91, 64, 65 respectively).

There are several parameters in GAs that have to be tuned by the user. Some among these are the population size, probabilities of performing crossover and mutation, and the termination criteria. Most of such parameters in GAs are problem dependent, and no guidelines for their choice exist in the literature. Therefore, several researchers have also kept some of the GA parameters variable and/or adaptive.

In our experiments, we adopted the mutation rate (P_1) and the crossover rate (P_2) to be 0.05, and 0.60, respectively. In addition, we select the varied parameters of the GA for each run to estimate the effect of these parameters on the segmentation accuracy. Table 9 shows the setup of the parameters of our suggested algorithm in each run.

Table 9. Parameters setup.

Image#	Run #	The parameters of our approach			T
		population size(PS)	maximum of generations(MG)	chromosome length(CL)	
	1	20	50	20	25
	3	15	25	20	20
	4	10	25	15	20
	5	20	30	20	35
	6	10	20	10	20
	7	15	50	15	25
	8	20	30	20	35
	9	10	30	20	35
	10	20	30	20	25

5.1. Experiments on MRI images

The proposed algorithm is tested by different values of parameters as shown in Table 9. These tests show that few parameters always give good results while others lead to unstable results. Stable results are obtained using the three set of parameters (T=35, CL =20, MG=30, PS=20), (T=20, CL =10, MG=20, PS=10), and (T=25, CL =15, MG=50, PS=15) while other parameters give unstable results. The output of these runs includes *desirable* and *undesirable* regions. The desirable region is a complete region and the undesirable regions may be correct but incomplete or incorrect. For example, Fig. (3a) shows the output of the first run. It contains five desirable regions as shown in Fig. (3a) and five regions are undesirable as in Fig. (3b). In these experiments, Figs. 3(a,b), 4(a,b), 5(a,b), 6(a,b), 7(a,b), 8(a,b), 9(a,b), 10(a,b), and 11(a,b) show the output of the first, second, third, fourth, fifth, sixth, seventh, eighth, and ninth runs, respectively. We only calculate the accuracy of desirable regions that are shown in Figs. (4a, 5a, 6a, 7a, 8a, 9a, 10a, and 11a) and ignore all the undesirable regions that are shown in Figs. (4b, 5b, 6b, 7b, 8b, 9b, 10b, and 11b). Fig. (12) presents the ratio between the number of desirable regions and the true number of regions (according to reference regions) of each run corresponding to the three set of parameters. It is noted that the first set of parameter (T=35, CL =20, MG=30, PS=20) gives the best results of all other parameters. The percentage number of the true regions equals 84%, 100%, and 100% regions for image1, image2, and image3 respectively as shown in Fig.(12). However, the accuracy for each region is estimated using Eq.(5) and the mean accuracy is presented in the last column

of Table 10. The mean accuracy is found to be 0.876, 0.965, and 0.976 for image1, image2, and image3 respectively. In other hand, we calculate the ratio between the number of desired regions and the total number of resultant regions when the proposed method is applied to the three test images using the three different set of parameters as shown in Fig. (13). This figure shows that the proposed method almost give good results when the set of parameters is chosen to be (T=35, CL =20, MG=30, PS=20) as shown in run1, run4, and run7.

The experimental results show that the proposed method gives more accurate results and more stable in complex medical image structures even with varying chromosome parameters (PS, MG, and CL). It is interesting to note that the average of the runs for GA produces keys that are at a very similar level of correctness, yet the GA with the proposed fitness function is scoring significantly higher fitness levels. The proposed approach has a major improvement of the original GA in three aspects:

(1) It does not require the a priori number of image regions, however it can effectively and adaptively controls the segmentation quality; (2) the chromosome structure is revised from the original and condition of fitness is convinced; (3) three revised genetic operations with region growing method are presented to make the algorithm computation-efficient.

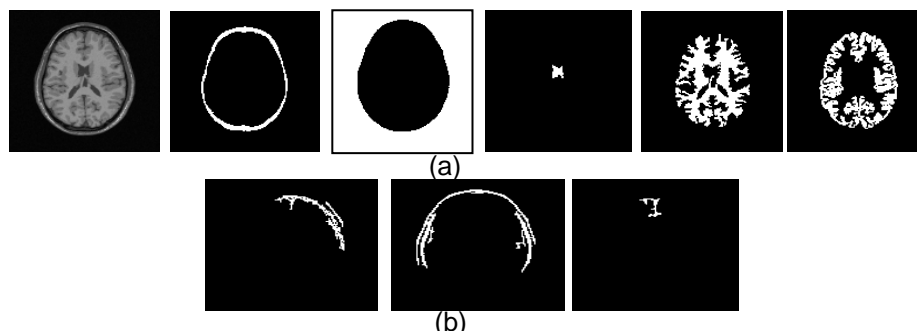


Fig. 3. The results of the first run when T = 25

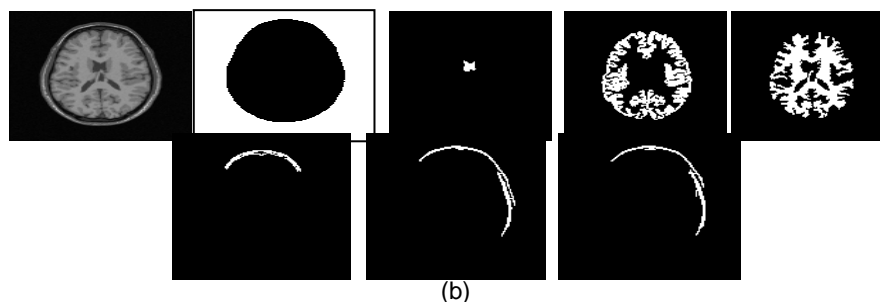


Fig. 4. The results when T = 20.

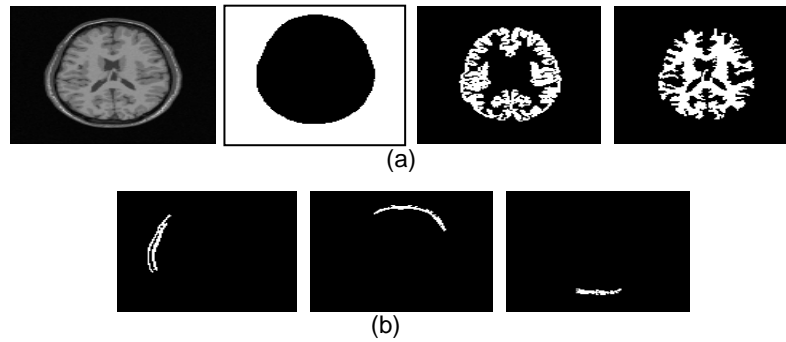


Fig. 5. The results when $T = 20$.

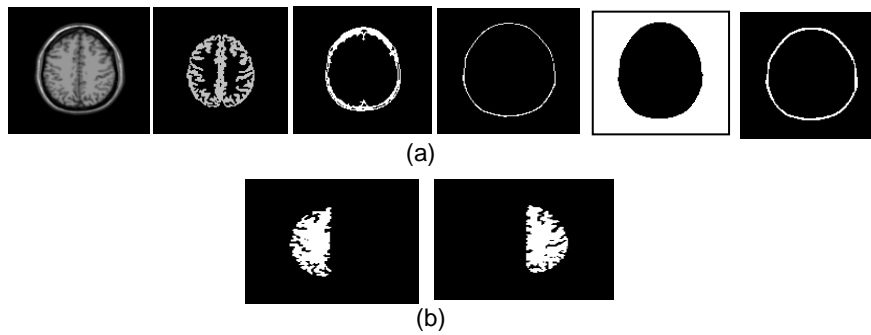


Fig. 6. The results when $T = 35$.

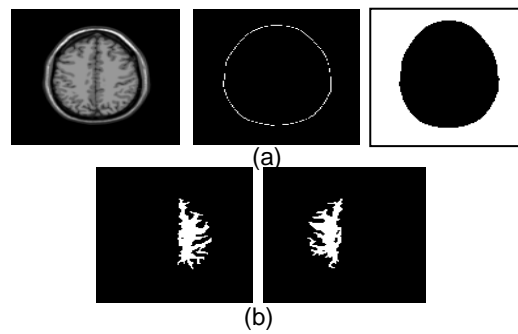


Fig. 7. The results when $T = 20$.

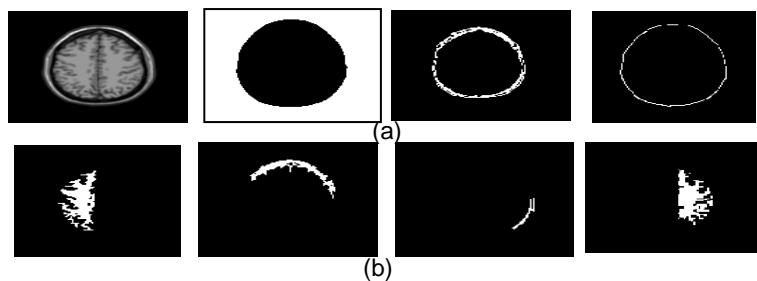


Fig. 8. The results when $T = 25$.

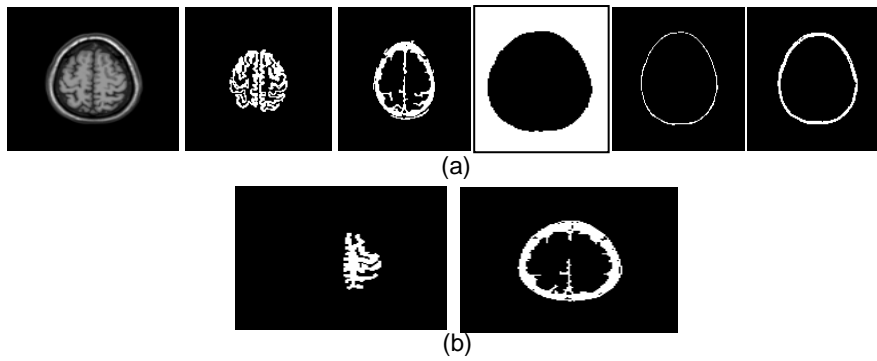


Fig. 9. The results when $T = 35$.

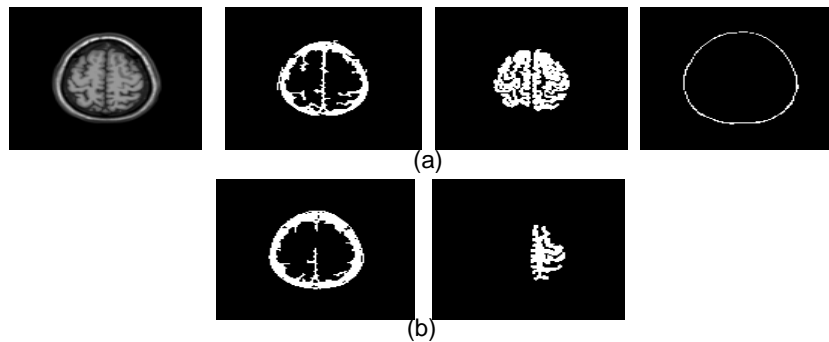


Fig. 10. The results when $T = 35$.

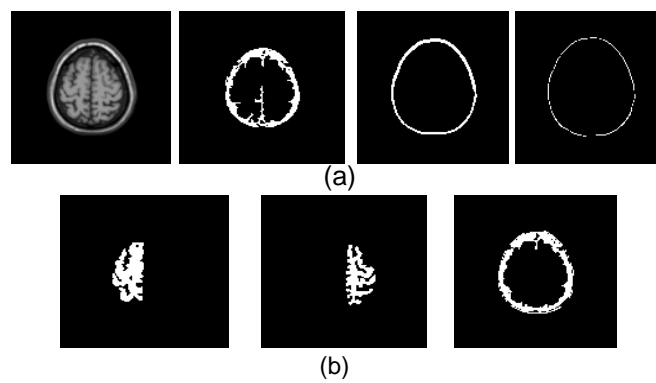


Fig.11. The results when $T = 25$.

Table 10. Comparison between the fitness functions

Image#	Run #	Total number of generated regions	Number of Correct Regions	Efficiency = number of regions/ total number of regions in the image%	Mean accuracy (%)
1	1	8	5	5/6 = 84%	0.876
	2	7	4	4/6=67%	0.765
	3	6	3	3/6=50%	0.45
2	4	7	5	5/5=100%	0.965
	5	4	2	2/5=40%	0.232
	6	7	3	3/5=60%	0.553
3	7	7	5	5/5=100%	0.976
	8	5	3	3/5=60%	0.643
	9	6	3	3/5=60%	0.615

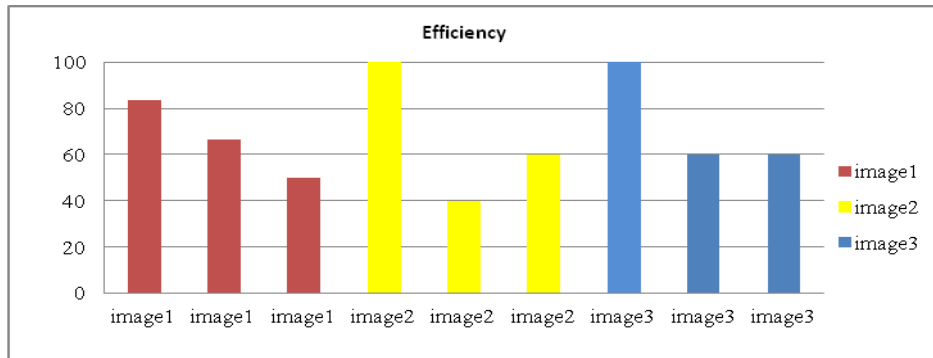


Fig. 12. The relation between number of desired regions and the resultant number when the proposed method applied to three test images using three different sets of parameters($T=35$, $CL=20$, $MG=30$, $PS=20$), ($T=20$, $CL=10$, $MG=20$, $PS=10$), and ($T=25$, $CL=15$, $MG=50$, $PS=15$).

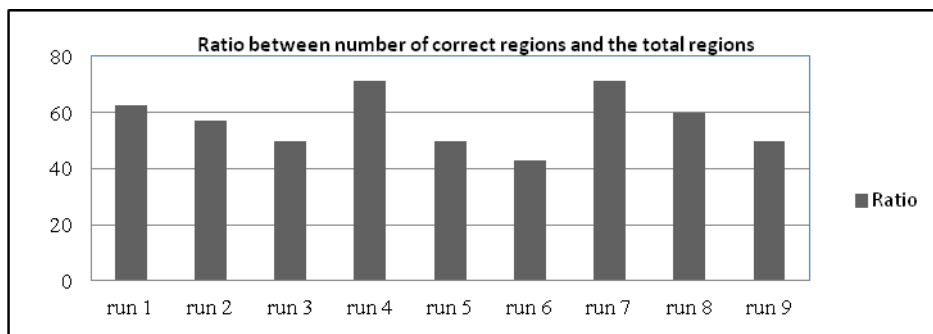


Fig. 13. The ratio between number of desired regions and the total number of resultant regions for nine runs: runs 1, 2, 3 on the first image, runs 4, 5, and 6 on the second image and runs 7, 8, and 9 on the third image.

5.2. Comparative results

In this section, the proposed method is compared to the fuzzy c-means (FCM) [19] and the recent hybrid GA and fuzzy c-means (GAFCM) algorithms [17] to prove its efficiency. These methods are experimented on three T1-weighted image with noise level 0% and 6% respectively as shown in Fig. 2(a,b,c). Table 11 presents the segmentation accuracy (Eq.(5)) for the proposed method and the FCM and GAFCM algorithms. The objective function J_m of fuzzy clustering methods is minimized:

$$\sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \|x_j - c_i\|^2 \quad (6)$$

$$\sum_{i=1}^C u_{ij} = 1$$

where u_{ij} is the membership, N is the number of data, C is the number of clusters, c_i is the cluster centre of fuzzy group i , and the parameter m is a weighting exponent on each fuzzy membership. More discussion can be discussed in [19].

Through fuzzy implementation, we set the following parameters: $m = 2$, a 2×2 window centered at each pixel, and $C=6, 5$, and 5 for test images 1, 2, and 3 respectively. The mean accuracy (%) of all resultant clusters is presented in Table 11. After obtaining the result from the proposed method, a post process was used to smooth and merge the similarity regions into one segment. It was noted that the proposed method achieves more accurate results and is stable in all images. The most important thing is that our method gave the best accuracy without any prior knowledge about image structure, while fuzzy methods must know the cluster number C and the exponential weight m , if we take into account the data shape in the objective function [17]. Moreover, the proposed method achieves better results than GAFCM by factors of 10 (for 0% noise level) and 9% (for 6% noise level) and better results than FCM by factor 13% (0% noise) and 20% (6% noise). This proves that the proposed method is more stable and efficient even for large factor noise. Also, we noted that the hybridization of GA with FCM (GAFCM) is always gives better results than FCM.

Table 11. Comparison between the accuracy of the proposed method and FKM and PFCM.

Image#	Accuracy					
	0%			6%		
	Proposed method	GAFCM	FCM	Proposed method	GAFCM	FCM
1	0.914	0.813	0.785	0.906	0.810	0.701
2	0.967	0.814	0.772	0.955	0.840	0.746
3	0.970	0.854	0.732	0.946	0.82	0.765

6. Related work

The hybridization of GA and other methods became necessary to overcome the limitations for applying each one alone in medical image segmentation [15-17]. For example, the combination of GA with contour method have solved the problem of sensitivity to the initial position of contour and the entrapment within local minima which are the problems inflicted on the active contour model used for image segmentation [15]. This problem is magnified when one tries to apply the model to the segmentation of low quality images like an ultrasound images. The genetic active contour is used for examining the tissue with two closed contours (two centroid circle with two radiuses). In this situation, the specific region under study can be considered as concentric circles, where each of these circles acts like a contour. Each of these contours can contain points from the tissue's edge. Therefore, by separating these points from each contour and connecting them together, one can reveal the tissue's edge. Through the use of this algorithm, the problems of determining the contour's initial position and contour's entrapment within local minima no longer exist and the only thing needed for specifying the contours' initial positions is the center of the circles which could be found by determining the image's center of gravity.

Halder et al. [33] described a GA-based approach for gray-scale image segmentation that segments an image into its constituent parts automatically. They use fuzzy c-means clustering to help in generating the population of GA which there by automatically segments the image. In this method, the FCM is run t times for generating these t chromosomes; each chromosome is of size k . Each chromosome of the population is a potential solution by FCM algorithm with no. of clusters $C=k$. The Fuzzy C-Means (FCM) algorithm assigns pixels to each category by using fuzzy memberships. The fitness computation is accomplished in two steps. First, the pixel dataset is clustered according to the centers encoded in the chromosome under consideration, such that each intensity value x_i , $i = 1, 2, \dots, m \times n$ is assigned to cluster with center z_j , $j = 1, 2, \dots, k$. The next step involves adjusting the values of the cluster centers encoded in the chromosome, replacing them by the mean points of the respective clusters. Subsequently, the clustering metric is computed as the sum of Euclidean distances of each point from their respective cluster centers. A low value of intra-cluster spread is a characteristic of efficient clustering. Hence the objective is to minimize the clustering metric.

Wang et al. [17] combined GA with clustering fuzzy c-means by improving the following aspects:(1) the parameters in the genetic algorithm were adjusted adaptively according to the value and the varying velocity of individual fitness to increase the genetic algorithm's adaptability and the accuracy of results;(2) the constraint based on the second order derivative of histogram was introduced into genetic algorithm to reduce the searching scope and increase calculating efficiency. Consequently, a novel fuzzy clustering image segmentation algorithm based on improved genetic algorithm was proposed.

The main disadvantage of the hybridization methods [15-17, 33] is the difficulty of searching for the proper number of classes in case of fuzzy c-means which lacks number of clusters. For that the user should feed these algorithms by number of clusters manually which always undefined for real images. Moreover, these methods suffer to oversegmentation.

7. Conclusion

In this paper, a novel approach for automatic segmentation has been presented that could improve medical images segmentation. The proposed method combines the GA algorithms process and seed region growing to overcome their limitations such as: the accuracy of image segmentation and the consumed time for the search space.

The method starts by optimising the parameters of existing image segmentation algorithms to find: a best way to represent the chromosomes, a fitness function that is a good measure of the quality of image segmentation, and the best way to reduce the consumed time. These improvements are included in our algorithm to present automatic, accurate, and stable medical image segmentation.

Without needing any details or information about the target medical image, the approach randomly selects an initial set of chromosomes that are distributed through the image. Each chromosome is represented as a vector of values which includes control genes, gray-levels genes, and position genes. Each control gene in the chromosome is associated with a gray-level. These values are passed to the region growing algorithm which uses them as initial seeds to find accurate regions for each control gene value. Each chromosome is evaluated by the proposed fitness function. Then, each chromosome is updated by applying the operators of genetic algorithms to evolve segmentation results.

Despite the available guidelines to set mutation and crossover parameters, the proposed approach is not a parameter-free algorithm since the number of generations and the area of pixel windows used as feature vectors must be specified externally. However, their tuning is facilitated by the fact that the former is indexed on the size of the image and only a few choices have to be considered for the latter. Moreover, we redefined all genetic operations and extended the chromosome size to include three different types of genes. In addition, we proposed a new function which enhanced the efficiency of the proposed genetic algorithm.

The proposed algorithm has been tested on real MRI images with noise levels up to 6%. The superiority of the proposed algorithm is demonstrated by comparing its performance with the existing GAFCM and FKM algorithms. We noted that the segmentation accuracy of the proposed method is increased over the existing GAFCM by factors of 10 (for 0% noise level) and 9% (for 6% noise level) and over than FCM by factor 13% (0% noise level) and 20% (6% noise level). Finally, we suggest to the readers that the application of the

proposed genetic method to other abnormal braincases will be the topic for further research.

References

1. D. W. Shattuck, S. R. Sandor-Leahy, K. A. Schaper, D. A. Rottenberg, R. M. Leahy, "Magnetic resonance image tissue classification using a partial volume model", *NeuroImage*, vol. 13, pp. 856-876, 2001.
2. E. A. Zanaty, S. Aljahdali, N. Debnath, "A kernelized fuzzy c-means algorithm for automatic magnetic resonance image segmentation", *Journal of Computational Methods in Science and Engineering (JCMSE)*, pp. 123-136, 2009.
3. K. Nakamura, E. Fisher, "Segmentation of brain magnetic resonance images for measurement of gray matter atrophy in multiple sclerosis patients", *NeuroImage* vol. 44, pp. 769-776, 2009.
4. D. N. Chun, H. S. Yang, "Robust image segmentation using genetic algorithm with a fuzzy measure", *Pattern Recognition*, vol. 29, no. 7, pp. 1195-1211, 1996.
5. Y. J. Zhang, "A survey on evaluation methods for image segmentation", *Pattern Recognition* vol. 29, no. 8, pp. 1335-1346, 1996.
6. Y. J. Zhang, J. J. Gerbrands, "Objective and quantitative segmentation evaluation and comparison", *Signal Processing*, vol. 39, pp. 43-54, 1994.
7. D. E. Goldberg, "Genetic algorithms in search, optimization and machine learning" New York: Addison-Wesley, 1989.
8. M.E. Farmer and D. Shugars, "Application of genetic algorithms for wrapper-based image segmentation and classification", *IEEE Congress on Evolutionary Computation*, pp. 1300-1307, 2006.
9. M. Yoshimura, S. Oe, "Evolutionary segmentation of texture using genetic algorithms towards automatic decision of optimum number of segmentation areas", *Pattern Recognition*, vol. 32, pp. 2041-2054, 1999.
10. P. Andrey, "Selectionist relaxation: genetic algorithms applied to image segmentation", *Image and Vision Computing*, vol. 17, pp.175-187, 1999.
11. C. Li, R. Chiao, "Multiresolution genetic clustering algorithm for texture segmentation", *Image and Vision Computing*, vol. 21, pp. 955-966, 2003.
12. K. E. Melkemi, M. Batouche, S. Fofou, "A multiagen system approach for image segmentation using genetic algorithms and extremal optimization heuristics", *Pattern Recognition Letters*, vol. 27, pp.1230-1238, 2006.
13. C. C. Lai, C. Y. Chang, "A hierarchical evolutionary algorithm for automatic medical image segmentation", *Expert Systems with Applications*, vol. 36. no. 1, pp. 248-259, 2007.
14. K. K.Pavan, V. S.Srinivas, A. SriKrishna, B. E. Reddy "Automatic tissue segmentation in medical images using differential evolution", *Journal of Applied Sciences*, vol. 12, pp. 587-592, 2012.
15. M. Karman, S. Lee, "Advanced classification using genetic algorithm and image segmentation for Improved FD", *Second International Conference on Computer Research and Development*, pp.364-368, 2010.
16. M. Talebi, A. Ayatollahi, A. Kermani, "Medical ultrasound image segmentation genetic active counter", *Journal of Biomedical Science and Engineering* vol. 4, pp.105-109, 2011.
17. H. Wang, B.J. Zhang, X.Z. Liu, D.Z. Luo, S.B. Zhong, "Image segmentation method based on improved genetic algorithm and fuzzy clustering", *Advanced Materials Research* vol. 379, pp. 143-144, 2011.

18. U. Maulik, "Medical image segmentation using genetic algorithms", IEEE Transactions on Information Technology In Biomedicine, vol. 13, no.2, pp. 166-173, 2009.
19. A. W. C. Liew, H. Yan, "An adaptive spatial fuzzy clustering algorithm for 3-D MR image segmentation", IEEE Transactions on Medical Imaging, vol. 22, no. 9, pp. 1063–1075, 2003.
20. B. Web, "Simulated brain database", McConnell Brain Imaging Centre, Montreal Neurological Institute, McGill, <http://brainweb.bic.mni.mcgill.ca/brainweb/>.
21. F. Hsieh, C. Han, N. Wu, T. C. Chuang, K. Fana, "A novel approach to the detection of small objects with low contrast", Signal Processing, vol. 86, pp. 71–83, 2006.
22. D. W. Chakeres, P. Schmalbrock, "Fundamentals of magnetic resonance imaging", Williams and Wilkins, Baltimore, 1992.
23. R. B. Buxton "Introduction to functional magnetic resonance imaging-principles and techniques", Cambridge University Press, 2002.
24. A. W. C. Liew, H. Yan, "Current methods in the automatic tissue segmentation of 3D magnetic resonance brain images", Medical Imaging Reviews, vol. 2, pp. 91-103, 2006.
25. H. Yan, J. C. Gore, "An efficient algorithm for MR image reconstruction without low spatial frequencies", IEEE Transactions on Medical Imaging, TMI-9, pp.179-184, 1990.
26. R. Adams, L. Bischof, "Seeded region growing", IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 16, pp. 641-647, 1994.
27. J. Fan, G. Zeng, M. Body, M. Hacid, "Seeded region growing: an extensive and comparative study", Pattern Recognition Letters, vol. 26, pp. 1139-1156, 2005.
28. Z. Michalewicz, "Genetic algorithms + data structures = evolution programs", New York: Springer-Verlag, 1992.
29. S. Bandyopadhyay, S. K. Pal, "Classification and learning using genetic algorithms: application in bioinformatics and web intelligence", Berlin, Germany: Springer, 2007.
30. P. Felzenszwalb, D. Huttenlocher, "Efficient graph-based image segmentation", international Journal of Computer Vision, vol. 59, no. 2, pp. 167-181, 2004.
31. K. Deb, H. Beyer, "Self-adaptive genetic algorithms with simulated binary crossover", Evolutionary Computation Journal, vol. 9, no. 2, pp. 197-221, 2001.
32. L. Davis, "Handbook of genetic algorithms", New York: Van Nostrand, 1991.
33. A. Halder, S. Pramanik, A. Karm, "Dynamic Image Segmentation using Fuzzy C-Means based Genetic Algorithm", International Journal of Computer Applications vol. 28, no.6, pp. 15-20, 2011.

E. A. Zanyat is an Associate Professor at the College of Computers and Information Technology, Taif University, Saudi Arabia. Before, he was an Associate Professor of Computer Science at Sohag University, Egypt. He received his MSC and PhD studies at TU Chemnitz, Germany. His research interests are data reduction, image segmentation, medical image processing, and image reconstruction. In these areas he has published several technical papers in refereed international journals and conference proceedings.

Elnomery A. Zanaty and Ahmed S. Ghiduk

Ahmed Ghiduk is an assistant professor at Beni-Suef University, Egypt. He received the BSc degree from Cairo University, Egypt, in 1994, the MSc degree from Minia University, Egypt, in 2001, and a Ph.D. from Beni-Suef University, Egypt. His research interests include software engineering especially search-based software testing, genetic algorithms, and ant colony, and image processing. Currently, Ahmed S. Ghiduk is an assistant professor at College of Computers and Information Systems, Taif University, Saudi Arabia.

Received: June 4, 2012; Accepted: December 6, 2012