A Robust Dynamic Niching Genetic Clustering Approach for Image Segmentation

Dongxia Chang
Institute of Information Science,
Beijing Jiaotong University,
Beijing Key Laboratory of Advanced Information Science and Network Technology,
Beijing, China, 100044
dxchang@bjtu.edu.cn

Yao Zhao
Institute of Information Science,
Beijing Jiaotong University,
Beijing Key Laboratory of Advanced Information Science and Network Technology,
Beijing, China, 100044
yzhao@bjtu.edu.cn

Yanhui Xiao
Institute of Information Science,
Beijing Jiaotong University,
Beijing Key Laboratory of Advanced Information Science and Network Technology,
Beijing, China, 100044
xiaoyanhui@gmail.com

ABSTRACT

In this paper, a novel genetic clustering algorithm based on dynamic niching (DNGA) for image segmentation is proposed. It is an effective and robust approach to image segmentation on the basis of a total similarity function relating to the approximate density shape estimation. In the new algorithm, a dynamic identification of the niches is performed at each generation to automatically evolve the proper number of clusters and appropriate cluster centers of the data set. Moreover, a local search method is embeded in the evolutionary process which makes the dynamic niching method insensitive to the radius of the niche. Compared to existing methods, DNGA algorithm does not need to pre-specify the number of segmentation. Several images are used to demonstrate its superiority. The experimental results show that DNGA algorithm has high performance, effectiveness and flexibility.

Categories and Subject Descriptors: I.4.6 Segmentation: pixel classification

General Terms: Algorithms.

Keywords: genetic algorithm, clustering, niche, image segmentation.

1. INTRODUCTION

Image segmentation is a first and key step for image analysis and pattern recognition [1]. It is is a process of partitioning an image into different regions that are homogeneous or "similar" in some image characteristics. These regions may roughly correspond to objects, parts of objects, or groups of objects in the scene represented by that image. It can also be viewed as the process of identifying edges that correspond to boundaries between objects, and regions that correspond to surfaces of objects in the image. This task is often the preliminary step in many video and computer vision applications, such as object localization or recognition, data compression, tracking, image retrieval, or understanding. In recent years, many segmentation algorithms have been proposed in the literature [2, 3]. Generally, they may be broadly divided into three categories: edge-based [4, 5], region-based [6, 7], and clustering-based [8, 9].

In computer vision and pattern recognition, clustering algorithm has been used extensively to image segmentation due to its clustering validity and simplicity of implementation. It is a pixels clustering process of dividing pixels into clusters so that pixels in the same cluster are as similar as possible and those in different clusters are as dissimilar as possible. This accords with segmentation application sice different regions should be visually as different as possible. However, the implementations of the traditional clustering algorithms often encounter two unavoidable initialization difficulties of deciding the cluster number and obtaining the initial cluster centers that are properly distributed. In order to overcome these problems, stochastic clustering algorithms based on simple Genetic Algorithm (GA) [10, 11] or its variants have been proposed [12, 13, 14]. In fact, all these algorithms consider the clustering problem as an unimodal problem. Each chromosome is described by a sequence of the cluster centers. When all the cluster centers are contained in the chromosome, then the validity function reaches its global optimum. However, a simpler way is to consider the clustering problem as a multimodal problem and each cluster center corresponds to a local optimum of the validity function. In this circumstance, each chromosome represents a cluster center and all the local optima of the validity function should be found. Algorithms allow for the formation and the maintenance of different solutions and thus prevent the GA from being trapped in local optima, which can be considered to solve this problem.

In order to preserve the population diversity which prevents GAs being trapped by a single local optimum, several methods have been developed. In Ref. [15], the fitness sharing (FS) is introduced. The fitness of an individual is reduced if there are many other individuals near it, therefore the GA is forced to maintain diversity in the population. This method requires to define a similarity metric on the search space and an appropriate niche radius, where the radius represents the maximal distance among individuals to

GECCO'11, July 12–16, 2011, Dublin, Ireland.

be considered similar. It has been proved that when the number of individuals within the population is large enough and the niche radius is properly set, FS provides as many species in the population as the number of peaks in the fitness landscape [16]. In most circumstances, it is difficult to give an effective value for the niche radius without any a priori knowledge. In Ref. [17], the implicit fitness sharing is proposed. In this algorithm, sharing is accomplished by inducing competition for limited and explicit resources. The difficulty of appropriately choosing the niche radius is avoided. But, there are some other parameters needed to be set, such as the size of the sample of individuals to compete, the number of competition cycles and the definition of a matching procedure. In order to overcome the drawbacks of the FS methods, Deb and Goldberg proposed a criterion for estimating the niche radius in Ref. [18]. But the heights of the peaks and their distances should be known a priori. In Ref. [19, 20], two dynamic fitness sharing algorithms have been proposed. Only the individuals belonging to the same niche share the resources of the niche. The performance of these algorithms is also dependent on the niche radius. Species conserving genetic algorithm (SCGA) has been proposed in Ref. [21] which does not consider any sharing mechanism. Once a new species is discovered, its fittest individual is retained in the next generations until a fitter individual for that species is generated. Therefore, each species populating a region of the fitness landscape survives during the entire evolution, whether or not it corresponds to an actual niche. Moreover, this algorithm does not find all the niches perfectly when the peaks have different shapes.

Motivated by these observations, a novel clustering algorithm based on dynamic niching (DNGA) for image segmentation is presented in this paper. Within the DNGA algorithm, a dynamic niching with local search is developed to preserve the diversity of the population. A simpler representation is adopted, whereby each individual represents a single cluster center. All the niches presented in the population at each generation are automatically and explicitly identified. Then, the application of FS is limited to individuals belonging to the same niche. In order to overcome the dependence on the niche radius, a local search method is considered. This makes the algorithm work properly and independently of the niche radius even if the clusters are not equally spaced and have different volumes.

The rest of this paper is organized as follows. Section 2 provides the objective function of the segmentation problem used in the algorithm. The dynamic niching method is presented in Section 3. Section 4 describes the evolutionary clustering algorithm. Experimental results are given in Section 5. Experimental results demonstrate the efficiency of the DNGA clustering algorithm. Finally, conclusions are drawn in Section 6.

2. THE OBJECTIVE FUNCTION

Let $\mathbf{X} = \{ \mathbf{x}_1, \ \mathbf{x}_2, \ \cdots, \ \mathbf{x}_n \}$ be a finite subset of a N-dimensional vector space, K be the number of clusters and $S(\mathbf{x}_j, \mathbf{c}_i)$ be the similarity measure between $\mathbf{x}_j$ and the i-th cluster center $\mathbf{c}_i$. Our clustering goal is to find $\mathbf{c}_i$ to maximize the total similarity measure $J_s(\mathbf{c})$ with

$$J_s(\mathbf{c}) = \sum_{i=1}^{K} \sum_{j=1}^{n} \left(\exp\left(-\frac{\|\mathbf{x}_j - \mathbf{c}_i\|^2}{\beta} \right) \right)^{\gamma} \qquad (1)$$

where $\mathbf{c} = (\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_K)$ and β can be defined by

$$\beta = \frac{\sum_{j=1}^{n} \|\mathbf{x}_j - \bar{\mathbf{x}}\|^2}{n}, \text{ where } \bar{\mathbf{x}} = \frac{\sum_{j=1}^{n} \mathbf{x}_j}{n}. \qquad (2)$$

According to the analysis of γ in ref. [22], we know that γ can determine the location of peaks in the objective function $J_s(\mathbf{c})$. And the value of β is no longer sensitive to the peak. Let $\tilde{J}_s(\mathbf{x}_k)$ be the total similarity of the data point $\mathbf{x}_k$ to all data points with

$$\tilde{J}_s(\mathbf{x}_k) = \sum_{j=1}^{n} \left(\exp -\frac{\|\mathbf{x}_j - \mathbf{x}_k\|^2}{\beta} \right)^{\gamma}, \quad k = 1, 2, \cdots, n. \qquad (3)$$

This function can be seen closely related to the density shape of the data points in the neighborhood of $\mathbf{x}_k$. A large value for $\tilde{J}_s(\mathbf{x}_k)$ means that the data point $\mathbf{x}_k$ is close to some cluster centers and has many data points around it. A good estimation of γ can give a good estimation of the peak of $\tilde{J}_s(\mathbf{x}_k)$. Here, we use the data set shown in Fig. 1 (a) to see the influence of γ on Equation(3) and more detailed explanation can be found in Ref. [22]. Note that the "•" in Fig. 1 means the value of $\tilde{J}_s(\mathbf{x}_k)$ with respect to the data point $\mathbf{x}_k$, $k = 1, 2, \cdots, n$. According to Fig. 1 (b), only two peaks will be found when $\gamma = 1$ and all the five peaks will be separated when γ increases to 5 and 10 as shown in Fig. 1 (c) and Fig. 1 (d).

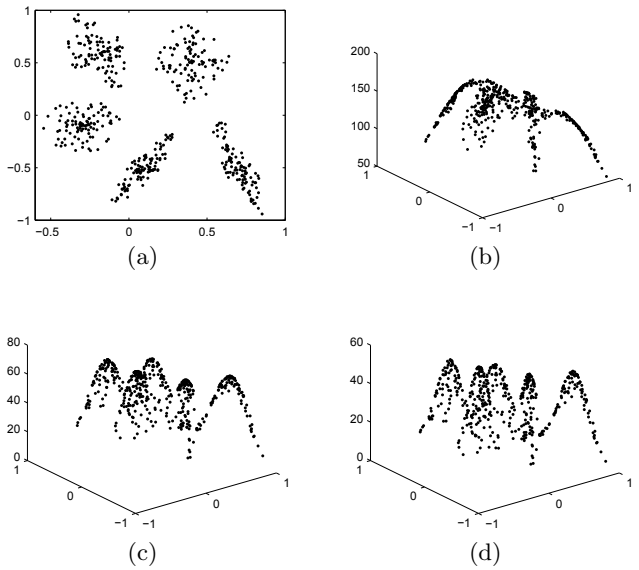


Figure 1: (a) Five-clusters data set.(b), (c) and (d) are plots of (3) (the approximate density shapes) with $\gamma = 1$, $\gamma = 5$ and $\gamma = 10$, respectively.

Here, the CCA algorithm [22] is used to estimate γ. For convenience, it is presented in the following:

1. Set $m = 1$ and $\varepsilon_1 = 0.97$.
2. Calculate the correlation of the value of $\tilde{J}_s(\mathbf{x}_k)_{\gamma_m}$ and $\tilde{J}_s(\mathbf{x}_k)_{\gamma_{(m+1)}}$.
3. If the correlation is greater than or equal to the specified ε_1, then choose γ_m to be the estimate of γ, else $m = m + 1$ and goto step 2.

After getting the estimation of γ, the function $\tilde{J}_s(\mathbf{x}_k)$ becomes a multimodal function, and the number of peaks is equal to the number of clusters. Therefore, the clustering problem can be transformed into a multimodal problem through this objective function. In the following, our new algorithm will be used to estimate all the local optima of $\tilde{J}_s(\mathbf{x}_k)$. The number of the local optima is the same as the number of clusters, and the local optima are the cluster centers. Moreover, the objective function $\tilde{J}_s$ is used as the fitness function in the evolutionary clustering algorithm.

3. THE DYNAMIC NICHING METHOD

In the evolutionary systems, a population of individuals evolve according to the transition operators, i.e., selection, crossover and mutation. At each generation, the individuals are selected according to their fitness and create offspring. At the end of the evolution process of the simple GA, the population consists of a single fittest individual, representing the best solution found by the algorithm. There are many cases, however, when the desired solution is not necessarily the best one, but rather a collection of best. In order to deal with this class of problem, niching has been suggested as a viable mean to simultaneously evolve subpopulations exploiting different niches by some kind of sharing.

In order to ensure that subpopulations are steadily formed and maintained, only the individuals belonging to the same niche should share the resources of the niche. But for the FS method, each individual in the population shares its fitness with all the individuals located at a distance smaller than the niche radius, no matter for the actual peak, i.e., for the niche, to which they belong. As a consequence, individuals belonging to different peaks may share their fitness, and this makes the perfect discrimination hypothesis is not satisfied. Therefore, it is not guaranteed that FS would provide as many subpopulations as the number of peaks. Moreover the number of peaks identified by the FS methods is dependent on the niche radius. In order to overcome these drawbacks, a dynamic niching method is proposed. In this method, each individual in the population belongs to one and only one niche.

From Ref. [19, 20, 21], we can see that the radius of the niches plays a crucial role in the identification of the niches and niche masters. If the radius chosen is too small, many niches may be found in every generation. On the other hand, a large value of the radius will make many solutions indistinguishable. This means that too few niches will be conserved. If the radius is so large that only one niche master is found, the algorithm will degenerate into a simple genetic algorithm and only find one optimum with the largest fitness value. In this section, a local search method is embeded in the dynamic niching algorithm which makes the algorithm insensitive to the niche radius.

Various types of local search methods are developed to find the optimum point in a bounded region of a continuous or discrete landscape. Among the local search algorithms, the gradient method is one of the widely used algorithms. Here, it is used in the dynamic niching algorithm. Our clustering objective is equivalent to the following optimization problem

$$\mathbf{c} = \arg\max_{\mathbf{c}} J_s(\mathbf{c}) = \arg\min_{\mathbf{c}}(-J_s(\mathbf{c})) \tag{4}$$

We differentiate $-J_s(\mathbf{c})$ with respect to $\mathbf{c}$ and get the gradient $\bigtriangledown_{\mathbf{c}}(-J_s(\mathbf{c}))$

$$\begin{aligned}\nabla_{\mathbf{c}}(-J_s(\mathbf{c})) &= \frac{\partial(-J_s(\mathbf{c}))}{\partial \mathbf{c}} \\ &= -\sum_{j=1}^{n} 2\frac{\gamma}{\beta}(\mathbf{x}_j - \mathbf{c})(\exp(-\frac{\|\mathbf{x}_j - \mathbf{c}\|^2}{\beta}))\end{aligned} \tag{5}$$

Then a one-step local search based on the gradient is applied to each individual and the individual moves along the local gradient, which leads to the following formulation of the steepest-descent method:

$$\begin{aligned}\mathbf{c} &= \mathbf{c} - \mu \bigtriangledown_{\mathbf{c}} (-J_s(\mathbf{c})) \\ &= \mathbf{c} + \mu \sum_{j=1}^{n} 2\frac{\gamma}{\beta}(\mathbf{x}_j - \mathbf{c}) \left(\exp\left(-\frac{\|\mathbf{x}_j - \mathbf{c}\|^2}{\beta}\right)\right)^{\gamma}\end{aligned} \tag{6}$$

Here, μ is the learning factor and $\mu \in (0, 1)$. This one-step local search procedure makes the individual move toward the local optimum. Therefore, the individuals belonging to the same peak will become closer. This enables the dynamic niching algorithm to lessen its dependency on the radius to some extent. The skeleton of the dynamic niching algorithm with local search is presented in Table 1. Here, Pop_t denotes the population of individuals at generation t.

Table 1: The dynamic niching algorithm

Input: Pop_t the population at generation t P population size σ the niche radius.
Phase I: The local search applied to each individual.
All individuals are updated according to Equation(6). Compute the fitness of each chromosome.
Phase II: The niche master candidates identification.
Sort the current population according to the raw fitness $v(t) = 0$ (the number of niches at generation t) $u(t) = 0$ (the number of niche master candidates)

For $i = 1$ to P do
If the ith individual is not marked then
$u(t) = u(t) + 1$
$N(u(t)) = 1$ (number of individuals in the $u(t)$th niche candidate)
For $j = i + 1$ to P do
If $(d(i, j) < \sigma)$ and ($u(t)$th individual is not marked)
insert the jth individual into the $u(t)$th niche masters candidate
$N(u(t)) = N(u(t)) + 1$
End If
End For
If $(N(u(t)) > 1)$ then
$v(t) = v(t) + 1$
mark the ith individual as the niche master of the $v(t)$th niche
End If
End If
End For

After the dynamic identification of the niche master candi-

dates of the population Pop_t at generation t, the individuals belonging to the same master candidate can be defined as a subset $S_t^i \neq \emptyset$ in the population Pop_t which have a distance from the master candidate less than the niche radius and do not belong to other niches. If the number of the individuals in S_t^i is larger than 1, then this subset is assumed as an actual niche; otherwise, the single individual in the subset is considered as an isolated individual and all the isolated individuals form the subset S_t^*. Then, the population Pop_t at the generation t is partitioned into $v(t)$ groups, say $S_t^1, \quad S_t^2, \quad \cdots, \quad S_t^{v(t)}$, and a number of isolated individuals

$$Pop_t = \left(\bigcup_{i \in \{1,2,\cdots,v(t)\}} S_t^i \right) \cup S_t^* \tag{7}$$

where S_t^* represents the set of all the isolated individuals.

The output provided by the dynamic niching algorithm is then used to implement the fitness sharing mechanism. The sharing fitness of each individual is calculated according to

$$f_{sh,t}(i) = \frac{f_t(i)}{m_t(i)} \tag{8}$$

Here, the shared fitness value for an individual within a niche (identified by the dynamic niching algorithm) is its raw fitness value divided by the niche count. Otherwise, the individual belongs to the isolated category, and its fitness is not modified. The niche count is calculated as

$$m_t(i) = \sum_{p_j \in S_t^i} sh(d_{ij}), \tag{9}$$

where

$$sh(d_{ij}) = \begin{cases} 1 - (\frac{d_{ij}}{\sigma_{sh}})^{\alpha_{sh}} & \text{if } \ d_{ij} < \sigma_{sh} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

and σ_{sh} is the niche radius, d_{ij} is the distance between individuals i and j, and α_{sh} is a constant parameter which regulates the shape of the sharing function. The value of α_{sh} is commonly set to 1, yielding to a triangular form for the sharing function.

After all the niches have been found, the new population is constructed by applying the usual genetic operators. Since some niche masters may not survive during the evolution, the species elitist strategy is implemented to enable the niche masters to survive. Here, only the actual masters are conserved.

4. THE DNGA CLUSTERING ALGORITHM

In this section, the DNGA algorithm is proposed, which can be used to optimize the objective function to automatically evolve the proper number of clusters and the appropriate segmentation of the image.

4.1 Chromosome representation and initialization

For any GA, a chromosome representation is needed to describe each individual in the population. The representation method determines how the problem is structured in the algorithm and the genetic operators that are used. Each chromosome is made up of a sequence of genes from certain alphabet. An alphabet can consist of binary digits (0 and 1), floating-point numbers, integers, symbols (i.e., A, B, C, D), etc. In early GAs, the binary digit was used. In our method, real-valued representation is used, e.g., a chromosome corresponds to a cluster center. Each chromosome is described by a sequence of N real-valued numbers where N is the dimension of the feature space. That is to say, the chromosome of the algorithm is written as

$$\mathbf{c} = [\ c_1, \quad c_2, \quad \cdots, \quad c_N \ ]. \tag{11}$$

An initial population of size P for DNGA algorithm is usually chosen at random. In this paper, P points are randomly chosen from the data set but on the condition that there are no identical points to initialize the P chromosome.

4.2 Fitness function

The fitness function is used to define a fitness value to each candidate solution. Here, the fitness function of the chromosome, f, is defined as the objective function introduced in Section 2

$$f(\mathbf{c}) = \tilde{J}_s(\mathbf{c}) = \sum_{j=1}^{n} \left(\exp - \frac{\|\mathbf{x}_j - \mathbf{c}\|^2}{\beta} \right)^{\gamma}, \quad j = 1, 2, \cdots, n \tag{12}$$

where $\mathbf{x}_j$, $j = 1, 2, \cdots, n$ are all data points in the data set to be clustered.

4.3 Genetic Operators

Any combination of standard selection, crossover and mutation operators can be employed by our algorithm. Here intermediate recombination and uniform neighborhood mutation are used.

For two randomly chosen parents $\mathbf{c}_1$ and $\mathbf{c}_2$, the offspring of the intermediate recombination crossover (with probability p_c) is

$$\mathbf{c} = \mathbf{c}_1 + r(\mathbf{c}_1 - \mathbf{c}_2) \tag{13}$$

where r is a uniformly distributed random number over $[0, 1]$.

Each chromosome undergoes mutation with a probability p_m. Let the minimum and maximum values of the data set along the ith dimension be $c_{\min}^i$ and $c_{\max}^i$, respectively. If the position to be mutated is the ith dimension of a cluster center with value c^i, then after uniform neighborhood mutation the value becomes

$$c^i = c^i + r_m R(c_{\max}^i - c_{\min}^i) \tag{14}$$

where R is a uniformly distributed random number over $[-1, 1]$ and $r_m \in (0, 1)$.

4.4 Description of the algorithm

In our DNGA algorithm, each chromosome represents one cluster center and is evaluated by using the fitness function described in Section 4.2. The niches are identified by the dynamic niching algorithm at each generation and the fitness sharing is computed in every niche. The evolutionary operators, selected on the basis of probability, can be crossover or mutation, where the former transforms two individuals (parents) into two offspring by combining parts from each parent, and the latter develops on a single individual and creates an offspring by mutating that individual. The elitist strategy [11] is implemented by replacing the worst chromosome of every niche with the niche masters found at each

generation. The process terminates after some number of generations, fixed either by the user or determined dynamically by the program itself, and the niche masters obtained are taken to be the solutions.

The DNGA algorithm is described as follows:

1. Initialize a group of cluster centers with size of P.
2. Evaluate each chromosome.
3. Apply the dynamic niching algorithm and apply the fitness sharing among the individuals belonging to the same niche. Copy the niche masters in a separate location.
4. If the termination condition is not reached, go to Step 5. Otherwise, select the niche masters from the population as the final cluster centers.
5. Apply the selection operator.
6. Apply crossover operator to the selected individuals based on the crossover probability.
7. Apply mutation operator to the selected individuals based on the mutation probability.
8. Evaluate the newly generated candidates.
9. Apply the elitist strategy.
11. Go back to Step 3.

5. EXPERIMENTS RESULTS

In order to validate the proposed algorithm, we have performed a set of experiments. In the experiment, the number of population is set to 500 and the maximum generation 50. The crossover and mutation probabilities used by DNGA algorithm is $p_c = 0.8$ and $p_m = 0.005$.

As mentioned earlier, the performance of the DNGA algorithm is insensitive to the niche radius. To prove this claim, we conduct an experiment for a data set shown in Fig. 2, in which we vary the value of the niche radius σ and count the number of niches found. The performances of the DNS [19], DFS [20], SCGA [21] and DNGA are compared through this experiment. The result is averaged over 20 runs for each value of σ. The results obtained by the four niching algorithms are shown in Fig. 3.

From Fig. 3, we can see that the performance of the DNGA algorithm is superior to the other three algorithms. As expected, as the niche radius is increased, the number of niches found by DNGA remains the same to the number of clusters. But for the other three algorithms, the number of clusters are identified correctly only within a small range of the niche radius.

In the following, the DNGA algorithm is used to segmentation several images taken from the Berkeley segmentation dataset [23] (shown in Fig. 4) and the segmentation result obtained through the grouping of the pixels. For the purpose of comparison, we have also executed two popular partitioning techniques Fuzzy C-means [24] and Expectation Maximization [25] on the test images with K is set as the actual number of clusters present in the image. The manually segmentations of the images are shown in Fig. 5. The clustering results for the four images are shown in Fig. 6. Here, we do not show the clustering results for all the contestant algorithms in order to save space.

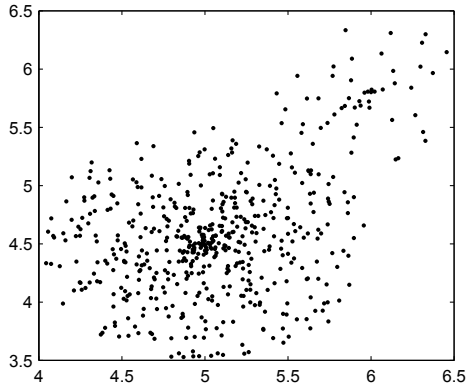


Figure 2: The data set used in the experiment.

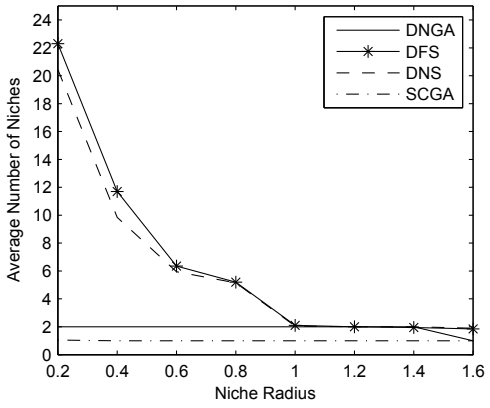
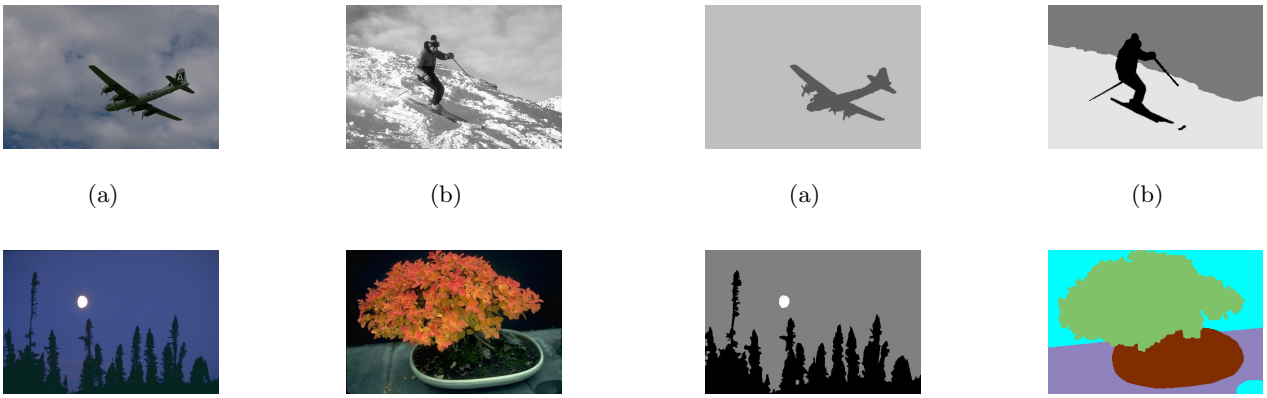


Figure 3: Variation in the average number of niches found with niche radius for the data set shown in Fig. 2. The population size $P = 100$.

Since test images are too heterogenous to allow for visual validation, some statistical tools have to be used to evaluate the segmentation results on a quantitative basis. However, a ground truth representation of the image is necessary to compute the statistical scores. Fortunately, the manually segmented images are available for the images used here. In order to compare the algorithms more carefully, three statistical measures (Overall accuracy, Kappa index and Adjusted rand index) are employed for the quantitative evolution of the final clustering results using the notations inspired from [26] are provided. Let n be the total number of pixels, and n_{ij} denote the number of pixels classified into class i as produced by the algorithm which also are in cluster j in the true cluster structure. Then $n_{i\cdot} = \sum_j n_{ij}$ be the number of pixels classified into cluster i in the image under experiment, and $n_{\cdot j} = \sum_i n_{ij}$ be the number of samples classified into class j in the ground truth image.

(1) Overall accuracy

This is the most widely used statistical score function for



(a) (b) (c) (d)

Figure 4: (a) image 1, (b)image 2, (c)image 3, (d)image 4.

(a) (b) (c) (d)

Figure 5: (a)Manually segmented image for image 1 (2 classes), (b)Manually segmented image for image 2 (3 classes), (c)Manually segmented image for image 3 (3 classes), (d)Manually segmented image for image 4 (4 classes).

the validation. The overall accuracy is the percentage of correctly classified pixels in the image. It is given by:

$$p_o = \frac{\sum_i n_{ii}}{n} \tag{15}$$

(2) Kappa index

This measure is very similar to the overall accuracy, but it introduces a chance agreement. A zero value would indicate that the classification agrees with the reference as bad as an arbitrary classification. The Kappa index is computed by:

$$Ka = \frac{n\sum_i n_{ii} - \sum_i n_{i+}n_{+i}}{n^2 - \sum_i n_{i+}n_{+i}} \tag{16}$$

(3) Adjusted Rand Index

The Rand Index is a clustering quality measure that measures the agreement of the clustering result with the true cluster structure. The Adjusted Rand Index additionally introduces a statistically induced normalization in order to yield values close to 0 for random experiments. The Adjusted Rand index is defined as [27]:

$$ARI = \frac{\sum_{ij} C_2^{n_{ij}} - \left[\sum_i C_2^{n_{i+}} \cdot \sum_j C_2^{n_{+j}}\right]/C_2^n}{\left[\sum_i C_2^{n_{i+}} + \sum_j C_2^{n_{+j}}\right]/2 - \left[\sum_i C_2^{n_{i+}} \cdot \sum_j C_2^{n_{+j}}\right]/C_2^n} \tag{17}$$

The Adjusted Rand Index return values in the interval $[0, 1]$ and the optimum score is 1, with higher scores being "better".

The three statistical measures described above are used to measure the segmentation performance of the algorithms. Table 2 shows the mean and standard deviation of the overall accuracy (in %),the kappa index (in %) and the adjusted rand index, all of which are calculated over 20 runs of the three clustering algorithms. Results show that these measures corresponding to the partitioning provided by DNGA algorithm is the best among all the partitions. This implies the superior performance of DNGA algorithm for automatically segment the proper partitioning from the images.

6. CONCLUSIONS

In this paper, a robust clustering algorithm based on dynamic niching genetic clustering algorithm (DNGA) has been developed for image segmentation with unknown cluster number. The DNGA algorithm can find the optimal number of segmentation and the cluster centers automatically. As the number of segmentation is not known a priori in most practical circumstance, DNGA algorithm can be used more widely. In the DNGA algorithm, each chromosome is encoded a center of a cluster by a sequence of real-valued numbers. This is more natural than the presentation used by other clustering algorithms based on simple GA. The dynamic niching is accomplished without assuming any a priori knowledge on the number of niches. The introduction of the one-step local search makes the DNGA algorithm insensitive to the niche radius. The superiority of the DNGA algorithm over FCM and EM algorithm for image segmentation has been demonstrated by the experiments on four images. Most data sets with different cluster volumes and also with noisy points can be successfully processed using DNGA clustering algorithm. All the experiment results have shown that our algorithm is effective, because it can provide all the actual cluster centers.

7. ACKNOWLEDGMENTS

This paper was supported by the Fundamental Research Funds for the Central Universities of China (2011JBM026, 2011JBM029); the China Postdoctoral Science Foundation (00480190); the Beijing Municipal Natural Science Foundation (4113075); the the National Natural Science Foundation of China (60803073, 60802045).

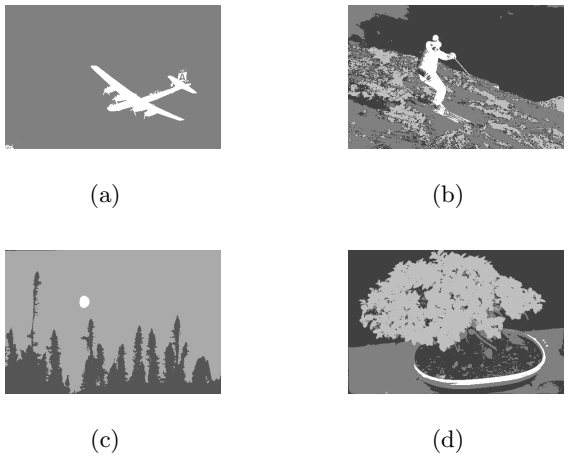
8. REFERENCES

[1] D. Ballard and C. Brown. Computer Vision, Englewood Cliffs, NJ: Prentice-Hall, 1982.

[2] K. S. Fu and J. K. Mei. A survey on image segmentation, Pattern Recognifion, 13:3-16, 1981.

Table 2: The mean and variance of the three statistical validity measures calculated on the final clustering results produced by FCM, EM and DNGA algorithms over 20 runs.

Data set	Validity measures	FCM	EM	DNGA
image 1	Overall accuracy(%)	98.38(1.4791e-32)	98.42(3.5655e-10)	99.40(2.3651e-8)
	Kappa index(%)	85.58(0)	85.62(2.2740e-8)	95.03(1.0297e-6)
	Adjusted Rand Index	0.8303(9.6977e-4)	0.8411(2.7739e-8)	0.9438(1.3826e-6)
image 2	Overall accuracy(%)	62.48(7.3994e-7)	82.53(1.5771e-5)	81.86(4.4517e-6)
	Kappa index(%)	41.64(4.9185e-4)	67.75(6.3220e-5)	66.79(1.4590e-5)
	Adjusted Rand Index	0.3093(3.4239e-33)	0.4737(1.1739e-4)	0.4562(2.8772e-5)
image 3	Overall accuracy(%)	79.17(0.0049)	95.30(4.9817e-9)	96.45(3.3000e-6)
	Kappa index(%)	59.92(0.0158)	88.36(3.4527e-8)	91.39(2.1568e-5)
	Adjusted Rand Index	0.6606(6.3514e-9)	0.8147(7.3163e-8)	0.8679(4.6910e-5)
image 4	Overall accuracy(%)	53.31(0.0263)	71.90(5.2436e-4)	90.43(1.4439e-5)
	Kappa index(%)	37.55(0.0537)	61.34(0.0010)	86.75(2.6649e-5)
	Adjusted Rand Index	0.3794(2.8987e-10)	0.5042(1.5722e-4)	0.8078(8.3737e-5)



(a) (b) (c) (d)

Figure 6: (a)Segmentation with DNGA algorithm for image 1(2 classes), (b)Segmentation with DNGA algorithm for image 2 (2 classes), (c)Segmentation with DNGA algorithm for image 3 (3 classes), (d)Segmentation with DNGA algorithm for image 4 (4 classes).

[3] R. Pal and S. K. Pal. A review in image segmentation techniques, Pattern Recognition, 26:1277-1294, 1993.

[4] P. K. Sahoo, S. Soltani, and A. K. C. Wong. A survey of thresholding technique, CVGIP 41:233-260, 1988.

[5] J. D. Helterbrand. One-pixel-wide closed boundary identification, IEEE Trans. on Image Processing, 5(5):780-783, 1996.

[6] R. Adams and L. Bischof. Seeded region growing, IEEE Trans. on Pattern Analysis and Machine Intelligence, 16(6):641-647, 1994.

[7] F. Calderero and F. Marques. Region merging techniques using information theory statistical measures, IEEE Trans. on Image Processing, 19(6):1567-1586, 2010.

[8] T. N. Pappas. An adaptive clustering algorithm for image segmentation, IEEE Trans. on Signal Processing, 40(4):901-914, 1992.

[9] K. S. Tan and N. S. M. Isa. Color image segmentation using histogram thresholding-Fuzzy C-means hybrid apprach, Pattern Recognition, 44(1):1-15, 2011.

[10] Z. Michalewicz. Genetic Algorithms + Data Structures = Evolution Programs, AI Series, Springer-Verlag, New York, 1994.

[11] K.A. De Jong. An analysis of the behavior of a class of genetic adaptive systems, Doctoral dissertation, University of Michigan, Ann Arbor, Michigan, 1975.

[12] S. Bandyopdhyay and S. Saha. GAPS: A clustering method using a new point symmetry-based distance measure, Pattern Recogn. 40:3430-3451, 2007.

[13] M. Laszlo and S. Mukherjee. A genetic algorithm that exchanges neighboring centers for k-means clustering, Pattern Recognition Letters, 28 (2007) 2359-2366.

[14] S. Saha and S. Bandyopadhyay. A new point symmetry based fuzzy genetic clustering technique for automatic evolution of clusters, Inform. Sci., 179(19):3230-3246, 2009.

[15] D. E. Goldberg and J. Richardson. Genetic Algorithms with sharing for multimodal function optimization, Genetic Algorithms and Their Applications, J.J. Grefenstette, Ed. Hillsdale, NJ: Lawrence Erlbaum, 41-49, 1987.

[16] S. W. Mahfoud. Population size and genetic drift in fitness sharing, Proc. Foundations Genetic Algorithms, L. D. Whitley and M. D. Vose, Eds., 185-223, 1995.

[17] R. E. Smith, S. Forrest and A. S. Perelson. Searching for diverse, cooperative populations with genetic algorithms, Evol. Comput., 1(2):127-149, 1992.

[18] K. Deb and D. E. Goldberg. An investigation of niche and species-for-mation in genetic function optimization, Proc. 3rd Int. Conf. Genetic Algorithms, J. D. Schaffer, Ed., San Mateo, CA, 42-50, 1989.

[19] B. L. Miller and M. J. Shaw. Genetic algorithms with dynamic niche sharing for multimodal function optimization, Proc. 1996 IEEE Trans. Evol. Comput., 786-791, 1996.

[20] C. D. Antonio, S. D. Claudio and M. Angelo. Where are the niches? Dynamic fitness sharing, IEEE Trans. Evol. Comput., 11(4):453-465, 2007.

[21] J. P. Li, M. E. Balazs, G. T. Parks and P. J. Clarkson. A species conserving genetic algorithm for multimodal function optimization, Evol. Comput., 10(3):207-234, 2002.

[22] M. S. Yang and K. L. Wu. A similarity-based robust clustering method, IEEE Trans. Pattern Anal. Machine Intell., 26(4):434-448, 2004.

[23] D. Martin, C. Fowlkes, D. Tal and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, Proc. 8th Int. Conf. Comput. Vis., 2:416-423, 2001.

[24] J. C. Bezdek. Fuzzy Mathematics in Pattern Classificaiton, Ph.D. Thesis, Cornell University, Ithaca, NY, 1973.

[25] A. K. Jain, M. N. Murthy and P. J. Flynn. Data clustering: a review, ACM Computing Reviews, 1999.

[26] R. G. Congalton, K. Green, Assessing the Accuracy of Remotely Sensed Data: Principles and Practices, Lewis, 45-50, 1999.

[27] W. M. Rand, Objective criteria for the evaluation of clustering methods, J. Am. Stat. Assoc, 66:846-850, 1971.