Will be presented in IEEE international conference on Signal and Image Processing, Dec 7-9 2006, BCET, Hubli, Karnataka.
Archived in Dspace@nitr, dspace@nitrkl.ac.in
pkanungo@rediffmail.com

CLASSIFICATION OF OBJECTS AND BACKGROUND USING PARALLEL GENETIC ALGORITHM BASED CLUSTERING

P. Kanungo[1], P. K. Nanda[2], A.Ghosh[3] and U. C. Samal[4]
[1,2,4]National Institute of Technology, Rourkela, Orissa, India 769008
[3]Indian Statistical Institute, Kolkota, India 700108
[1]p.kanungo@yahoo.co.in, [2]pknanda_d13@yahoo.co.in, [3]ash@isical.ac.in, [4]umesh.samal@rediffmail.com

ABSTRACT
In this paper, a novel strategy based on the notion of threshold is proposed to accomplish segmentation of objects and background in a scene. Optimal threshold for two class and three classes problems are determined from the histogram of featured pixel values as opposed to the original normalized histogram. Genetic algorithm (GA) and Parallel Genetic Algorithm (PGA) based clustering algorithms are proposed to determine the optimal thresholds for two as well as three class problems. The optimal threshold could segment the noisy images. Our results, for two class problems, could be comparable with that of Otsu's approach. Our approach yielded satisfactory results even for histograms having overlapping class distributions.

KEY WORDS
Segmentation, Genetic Algorithm, Parallel Genetic Algorithm, Clustering, Thresholding

1. Introduction

It is important in picture processing to select an adequate threshold of gray level for extracting object from the background. Image thresholding is a necessary step in many image analysis applications [1-4]. In its simplest form, thresholding means to classify the pixels of a given image into two groups (e.g. objects and background). One including those pixels with their gray values above a certain threshold, and the other including those with grey values equal to and below the threshold. This is called bi-level thresholding. Generally, one can select more than one threshold, and use them to divide the whole range of gray values in to several sub ranges. This process is called multilevel thresholding. Most thresholding techniques [5-9] utilize shape information of the histogram of given image while selecting thresholds.

In an ideal case, for images having two classes, the histogram has a deep and sharp valley between two peaks representing objects and back ground respectively. Thus the threshold can be chosen at the bottom of this valley [2]. However, for most real pictures, it is often difficult to detect the valley precisely, because (i) valley could be flat and broad and (ii) the two peaks could be extremely unequal in height, often producing no traceable valley. Rosenfeld et al [3] proposed the valley sharpening techniques which restricts the histogram to the pixels with large absolute values of derivatives, S. Watanable et al.[11] proposed the difference histogram method, which selects threshold at the gray level with the maximal amount of difference. These utilize information concerning neighbouring pixels or edges in the original picture to modify the histogram so as to make it useful for thresholding. Another class of methods deal directly with the gray level histogram by parametric techniques. The histogram is approximated in the least square sense by a sum of Gaussian distributions, and statistical decision procedures are applied [11]. However, such methods are tedious and computationally involved.

In this paper optimal threshold is determined based on histogram of featured pixels. The average value of the pixels of a window over a given pixel is the feature of the pixels taken into consideration. Thus, a featured image is generated based on the feature value of pixels. Optimal threshold is determined based on the histogram corresponding to featured image. Normalized discrete histogram is used to determine the optimal threshold. We have proposed GA and PGA based crowding algorithms to determine the peaks of modified histogram by maintaining stable subpopulations at the niches. Subsequently, the valleys between the successive peaks are determined based on GA based strategy. The valleys, thus determined, correspond to the optimal threshold. The histograms corresponding to many images, specifically noisy images, exhibit overlapping class distributions. But the histogram corresponding to the featured images reduces appreciably the overlapping of the distributions of classes in the histogram. Thus, the optimal threshold could be determined for non overlapping class distributions of the modified histogram.

2. Problem Statement

Often the problem in many computer vision applications is to detect objects from background. In a typical scenario of a object and background, it is necessary to obtain the proper segmentation followed by object recognition. These types of pictures or scenes exhibit bimodality in the histogram. Analogous situation is also present in case of two objects and a background. The histogram distribution of such scenes exhibit tri-modal features. Segmentation of such scenes requires determining optimal thresholds for proper segmentation. In the past, many thresholding techniques have been suggested [1-8] to obtain proper segmentation. The inherent bottleneck is the errors incurred due to the overlapping classes reflected in the histogram. In this work, the histogram of the featured pixels is considered as opposed the original histogram. A window around a pixel is considered and the first moment i.e. the average value of the pixels in the window is considered. This is defined as follows.

If $|x_{ij}-x_{ijav}| < threshold$, then $x_{ij} = x_{ijav}$ (1)

Where x_{ij} is the gray value corresponding to the $(i,j)^{th}$ pixel and x_{ijav} is the average gray value of the window centred at $(i,j)^{th}$ pixel. Thus another image, consisting of featured pixels is generated. Fig. 1 shows the histogram of a two class image as shown in Fig. 4 (c). It is seen from the Fig. 1 that there are distinct overlapping of the class distributions. The histogram of the featured image generated using (1) is shown in Fig. 2 that exhibits clearly bimodality with minimum overlapping. Thus optimal thresholds can be determined using Fig. 2.

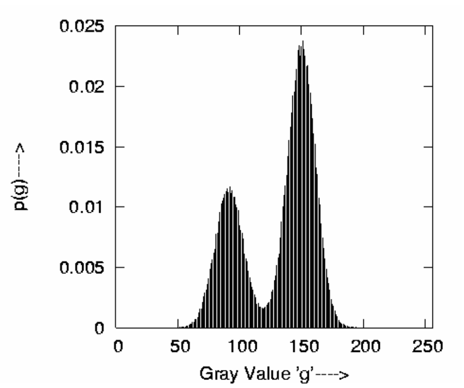

Fig. 1. Normalized Histogram of the original image

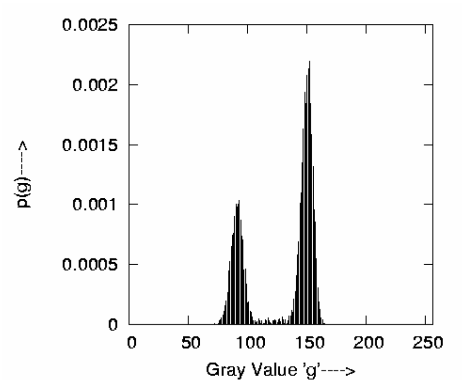

Fig. 2. The Featured based Normalized Histogram of the image

Hence, we propose GA based clustering to determine the optimal threshold using this discrete histogram. Usual search algorithm might determine threshold as suboptimal solution because of the discrete nature of the histogram. Hence a GA and PGA based algorithms are proposed to determine niches (peaks) followed by determination of the valley. This valley corresponds to the optimal threshold.

3. Segmentation Using Threshold Operators

As mentioned in Section 2, optimal threshold may be used to segment images. Optimal threshold is determined as follows. Given a histogram with multimodal features, the two peaks corresponding to two classes are determined using the proposed GA based crowding schemes. This algorithm determines the two peaks by maintaining stable sub population at the respective peaks of the multimodal histogram. Once the peaks are determined, the valley or the minima between these peaks are determined by GA. The two steps combinedly thus determines the valleys and hence the thresholds for segmentation.

3.1 GA Class Models

Usually GA are used for function optimization and hence determining the global optimal solution. In case of nonlinear multimodal function optimization, the problem of determining the global optimal solution as well as local optimal solution reduces to determining the niches in the multimodal functions. Thus the problem boils down to clustering the population elements around the given niches. Some effort has been directed in this direction for last couple of years where new strategies and algorithms are proposed [12].

3.2 Crowding method

In the deterministic crowding, sampling occurs without replacement [12-13]. We will assume that an element in a given class is closer to an element of its own class than to elements of the other classes. A crossover operation between two elements of same class yield two elements of that class, and the crossover operation between two elements of different class will yield either: (i) one element from both the classes, (ii) one element from two hybrid classes. For example, for a four class problem, the crossover operation between two elements of class AA and BB may results in elements either belonging to the set of classes AA, BB, or AB, BA. Hence the class AB offspring will compete against the class AB parents, the class BA offspring will compete with class BA parents. Analogously for a two class problem, if two elements of class A randomly paired, the offspring will also be of class A, and the resulting tournament will advance two class A elements to the next generation. The random pairing of two class B elements will similarly result in no net change to the distribution in the next generation. If an element of class A gets paired with an element of class B, one offspring will be from class A, and the other from class B. The class A offspring will compete against class A parent, the class B offspring against the class B parent. The end results will be that one element of the both classes advances to the next generation no net change.

3.3 GA for optimal threshold

Once the stable subpopulation is maintained at the peaks of the multimodal function, the problem is to determine the valley between the two successive peaks. The two peaks correspond to two gray levels and in between these two gray values the minimum of the function is found out. The fitness function here is the discrete histogram itself

Fitness function $f(g) = p(g)$ (2)

Where $p(g)$ denotes the featured histogram distribution. We have employed basic GA to determine the minimum of the objective function which is the fitness function itself.

3.4 GA Based Algorithm

Our proposed algorithm consists of determining all the peaks of the histogram distribution and in the sequel to find out the minima in between each pair of peaks. The salient steps of the algorithm are:

(i) Initialize randomly a population space of size N (each element corresponds to a gray value between 0 and 255) and their classes are determined.

(ii) Choose two parents randomly for crossover and mutation operation with crossover probability P_c and mutation probability P_m. Compute the fitness of parents and off-springs. The fitness function is the normalized featured histogram function p(g).

(iii) The offspring generated complete with the parents based on the concept of tournament selection strategy.

(iv) After selection the selected elements are put in their respective classes.

(v) Step (ii), (iii) and (iv) are repeated for all elements in the population.

(vi) Steps (v) is repeated till the convergence is met i.e. the elements of respective classes are equally fit.

(vii) The peaks will be determined from the converged classes of step (vi)

(viii) Initialize randomly a population space of size N_v between the two peaks (i.e. between the two corresponding gray values).

(ix) Choose two parents randomly for crossover and mutation operation with crossover probability P_c and mutation probability P_m. Compute the fitness of parents and off-springs. The fitness function is the featured normalized histogram function p(g).

(x) The fittest two elements between the parents and offspring are selected for the next generation in the selection strategy.

(xi)Step (ix), (x) are repeated for all elements in the population.

(xii) Step (xi) is repeated till the convergence is met.

(xiii) The converged value is the gray value corresponding to the valley between the two peaks. The image is then segmented using this value as threshold.

3.5 PGA for Optimal Threshold

The Objective of designing parallel GA is two fold: (i) reducing the computational burden and (ii) improving the quality of the solutions. The design of PGA involves choice of multiple populations where the size of the population must be decided judiciously. These populations may remain isolated or they may communicate exchanging individuals. This process of dividing the entire population into sub-populations and then providing the mechanism of interaction between them is known as coarse grained parallelism. The process of communications between individual demes is known as migration. The coarse grained PGA is broadly based on the island model and stepping stone model. In an island model the population is partitioned in to small subpopulations by geographic isolation and individuals can migrate to any other subpopulation. In this parallel scheme, the population is divided into demes and the demes evolve for convergence. After some generations migration is carried out to achieve convergence. We have adopted the following Interconnection Island model for the PGA algorithm. We have adopted the good-bad based migration policy. In our problem we considered four demes D1, D2, D3 and D4 and the interaction network model is shown in fig 3.

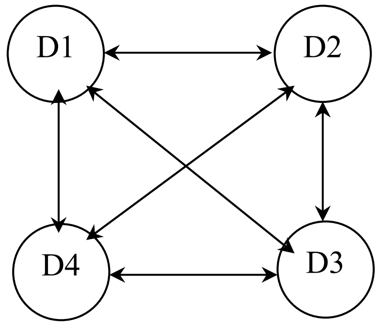

Fig. 3 Interconnection of demes

Tournament selection mechanism is applied to all demes. We have employed a new crossover operator known as *Generalized Crossover (GC)* operator [9].

3.6 PGA Based Algorithm

The steps of the parallelized crowding scheme are the following.

1. Initialize randomly a population space of size N (each element corresponds to a gray value between 0 and 255) and their classes are determined.
2. Divide the population space into fixed number of sub-populations and determine the class of individuals in each sub-population.
3. i. In the given sub-population, choose two elements at random for Generalized Crossover (GC) and Mutation operation with crossover probability P_c and mutation probability P_m.

   ii. Evaluate fitness of each parents and offspring. The fitness function is the featured normalized histogram function p(g).

   iii. The tournament selection mechanism is a binary tournament selection among the two parents and offsprings, the set which contains the individual having highest fitness among the four elements is selected to the set of parents for the next generation.

   iv. Repeat steps i, ii & iii for all the elements in the sub population.

v. Repeat steps i, ii, iii & iv for a fixed number of generations

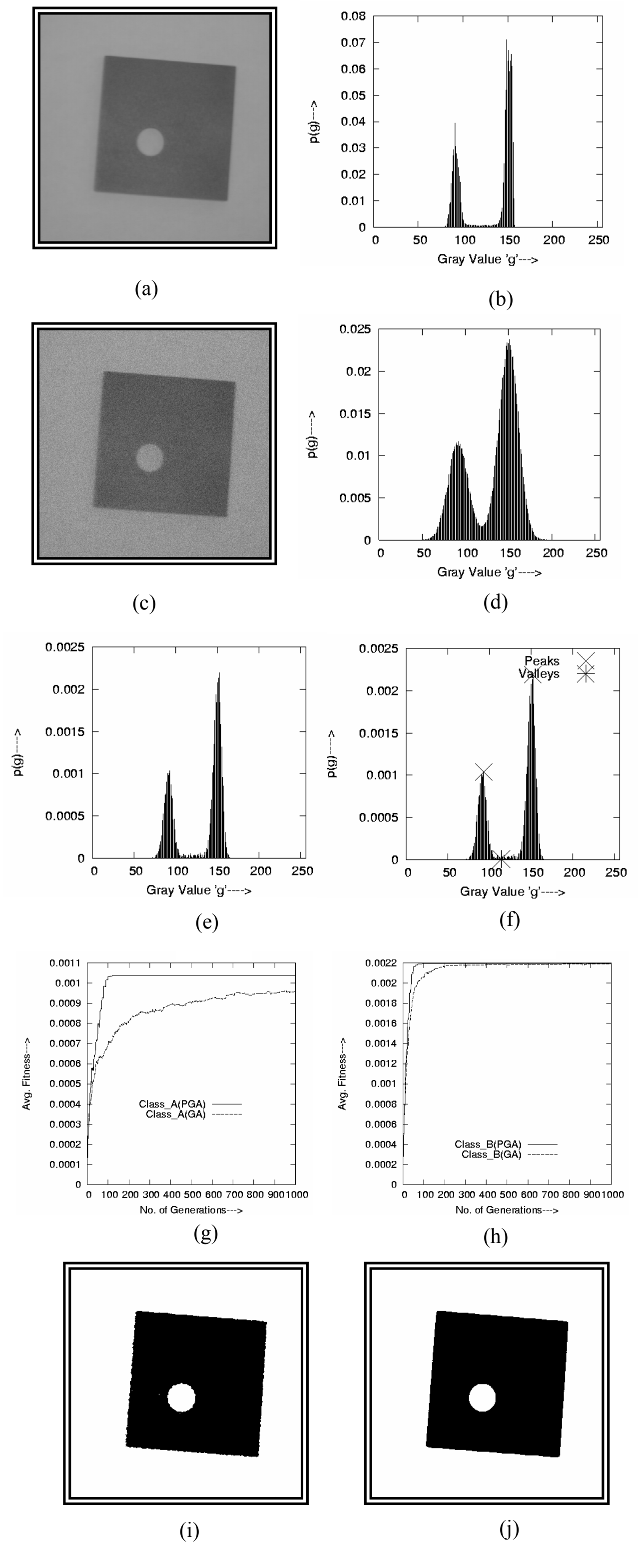

Fig. 4: (a) Original image. (b) Normalized Histogram. (c) Noisy Version of Original Image having SNR 22 dB. (d) Normalized Histogram of Image "c". (e) Featured Histogram of the Image "c". (f) Detected Peaks and Valley. (g) Average Fitness vs Generations of Class A for both PGA and GA. (h) Average Fitness vs Generations of Class B for both PGA and GA. (i) Segmented Image using the Proposed Method. (j) Segmented Image using Otsu's Method.

4. Step 3 is repeated for each sub-population.
5. Migration is allowed from each deme to every other deme. The individuals are migrated based on the selected migration policy [10-11]. Numbers of elements to migrate are determined from the selected rate of migration. The elements migrate with migration probability $P_{mig.}$
6. Repeat steps 3, 4 & 5 till convergence is achieved. The algorithm stops when the average fitness of the total population is above pre selected threshold.
7. The peaks will be determined from the converged classes of step 6.
8. Initialize randomly a population space of size N_v between the two peaks (i.e. between the two corresponding gray values of the peaks).
9. Employ PGA for determining the Valley between these two peaks. The image is then segmented using this value as threshold.

4. Results and discussion

In simulation, we have considered images having both bimodal and tri-modal features. Bimodal histogram corresponds to two classes where as the tri-modal histogram corresponds to three classes. Fig 4(a) shows the original image of size (374x374) while histogram is shown in Fig 4(b). Fig. 4(c) show the noisy image of SNR 22dB and the corresponding histogram of Fig 4(d) exhibit distinct overlapping. The noisy image is generated by adding Gaussian noise to the image of Fig. 4 (a). We define SNR as

$$SNR_{dB} = 10\log_{10}\frac{\sum_{ij} x_{ij}^2}{\sum_{ij} n_{ij}^2} \tag{3}$$

Where x_{ij} denotes the gray value of the pixels and n_{ij} denotes the samples from Gaussian distribution. The histogram in Fig. 4(e) corresponds to the image of featured pixels. It is observed from Fig 4(e) that the two modes corresponding to two classes are clearly separated. GA and PGA based algorithms determine the peak and valleys as shown in Fig. 4(f). The "×" symbol indicates the cluster of population in the peaks. The parameters used for GA are: Generation=1000, Probability of Crossover P_c=0.8, Probability of Mutation P_m=0.001, population size (N) = 400. The parameters used for PGA are: Generation=1000, Migration period is after 10 generation, Number of demes is 4, Probability of Crossover P_c=0.8, Probability of Mutation P_m=0.001, population size (N) = 400, Probability of migration P_{mig}=0.8, Migration rate is (R_{mig}) = 4%. The peaks are detected at gray values of 93 and 152. The valley corresponding to the threshold is found to be 114. The rate of convergence of GA and PGA based algorithms are shown in Fig. 4(g) and 4(h) for class A and class B respectively. It is observed from

Fig. 4(g), that PGA converges at around 100 generations while GA converges at around 1000 generations.

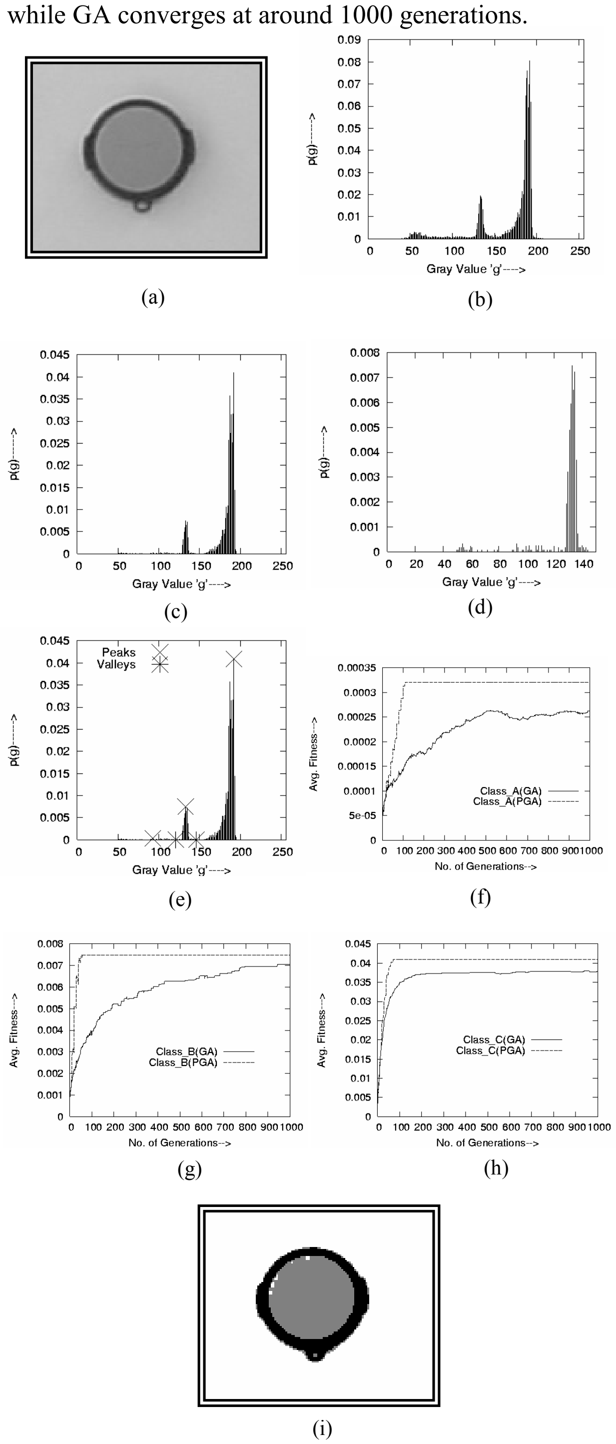

Fig. 5: (a) Original image. (b) Normalized Histogram. (c) Featured Histogram of the image "a". (d) Part of the Featured Histogram showing the 1st and 2nd Peak (e) Detected Peaks and Valley. (f) Average fitness vs Generations of Class A for both PGA and GA. (g) Average Fitness vs Generations of Class B for both PGA and GA. (h) Average Fitness vs Generations of Class C for both PGA and GA. (i) Segmented Image using the Proposed Method.

For class B, PGA converges around 50 iteration while GA converges after 200 generations. As observed in both the cases, PGA is much faster than that of GA. In a Pentium 4, 512 MB SDRAM & 3 GHz machine, GA takes around 8seconds while the serial implementation of PGA takes around 6 seconds and its parallel implementation would take 1.5seconds because of 4 demes. Fig. 4(i) is the segmented image obtained by our approach while Fig. 4(j) is the result obtained by Otsu's approach. We define segmentation accuracy as follows.

$$SA = (1-(N_{mc}/N_t)) \times 100 \qquad (4)$$

Where N_{mc} is the number of misclassified pixels and N_t represents the total number of pixels. Segmentation accuracy for our proposed algorithm for Fig. 4(i) is 99.15 and for Otsu's method 99.

In simulation, we have also considered images having three classes. One of the images and the corresponding histogram are shown in Fig 5(a) and Fig 5(b) respectively. It is observed from Fig 5(b) that there are three distinct modes with a pseudo mode towards the low values of gray levels. The corresponding featured image is generated and the corresponding histogram is shown in Fig. 5(c) while the lower part of the histogram is shown in Fig. 5(d). Although from Fig. 5(c) it is appears to have two modes the third mode is visible from Fig. 5(d). The three peaks and the thresholds detected by the algorithms are shown in Fig. 5(e). The peaks are at gray values of 93, 133 and 192 where as the valleys are at 121 and 146. The convergence of GA and PGA are shown in Fig. 5(f), (g) and (h) and it is observed in all the cases that PGA converges much faster than that of GA. The segmented result in Fig. 5(i) shows that three distinct classes could be obtained. The parameters of the algorithms remain unchanged. Another three class image used in our simulation is shown in Fig. 6(a) and the corresponding histogram in Fig. 6(b). It is seen from Fig. 6(b) that three distinct modes are observed with appreciable overlapping. The histogram of the featured image is shown in Fig. 6(c) where it is observed that the overlapping has been reduced. The peaks are found at 39, 123 and 211 and valleys are at 97 and 185 respectively. The clustering of the subpopulation and the determination valleys are shown in Fig. 6(d). The convergence of GA and PGA for three classes are shown in Fig.6(e), (f) and (g) and it is observed that PGA converges much faster than GA. The segmented image is shown in Fig. 6(h) where three distinct classes could be observed. Thus two and three class images could be segmented properly.

5. Conclusions

This work attempts to segment images having objects and background. The images could be either two or three class images. We have proposed a feature based strategy to determine the optimal thresholds. Thresholds are determined from the modified histograms. GA and PGA based strategies are proposed to determine clusters at the peaks and subsequently the valleys are determined. It is observed that GA takes appreciable amount of time and hence PGA is adapted to reduce the computational time. The results presented correspond to the serial implementation of the parallel algorithm.

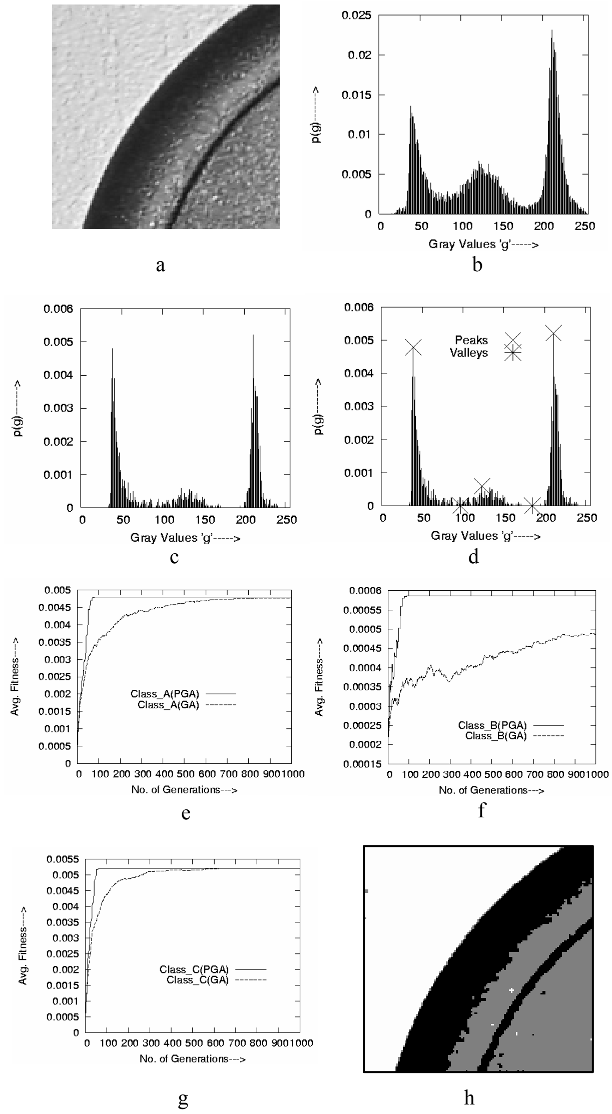

Fig. 6: (a) Original Image. (b) Normalized Histogram. (c) Featured Histogram of the image "a". (d) Detected Peaks and Valley. (e) Average Fitness vs Generations of Class A for both PGA and GA. (f) Average Fitness vs Generations of Class B for both PGA and GA. (g) Average Fitness vs Generations of Class C for both PGA and GA. (h) Segmented Image using the Proposed Method.

The proposed algorithm could segment satisfactory for two class images and the results were also very much comparable with that of Otsu's approach. We have also validated our approach for images having three classes. In case of images having three classes and the histogram exhibited appreciable overlapping among the classes. Hence, the segmented image contained few misclassified pixels which could be over come by post processing the image by median filtering. Thus, the misclassification due to the overlapping of the classes could be alleviated and proper segmentation could be obtained. Since, our algorithm takes very few generations and around 2 Seconds to determine the threshold, current attempts are to implement them in real time using Texas Instrument's TMS 320C 6711 based *Image Developer Kit* (IDK).

Acknowledgements

This research work has been supported by the Ministry of Human Resource and Development (MHRD), India project on "Real-Time Image Processing using soft computing Approach" and the collaborative research project with Centre for Soft Computing, ISI, Kolkata.

References

[1] K. S. Fu and J. K. Mui, A Survey on Image segmentation, *Pattern Recognition*, *13*, 1981, 3-16.

[2] M. Sezgin and B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, *Journal of Electronic Imaging*, *13*(1), 2004, 146-165.

[3] N. R. Pal and S. K. Pal, A Review on Image Segmentation Techniques, *Pattern Recognition*, *26*(9), 1993, 1277-1294.

[4] P. K. Sahoo, S. Soltani and A. K. C. Wong, A Survey of Thresholding Techniques, *Computer Vision, Graphics, and Image Processing*, *41*, 1988, 133-260.

[5] J. Kittler and J. Illingworth, Minimum Error Thresholding, *Pattern Recognition*, *19*(1), 1986, 41-47.

[6] J. M. S. Prewitt and M. L. Mendelsohn, The analysis of cell images, *Ann. N. Y. Acad. Sci.*, *128*, 1966, 1035-1053.

[7] J. S. Weszka, R. N. Nagel, and A. Rosenfeld, A threshold selection technique, *IEEE Trans. Comput.*, *C-23*, 1974, 1322-1326.

[8] N. Otsu, A Threshold Selection Method from Gray-Level Histograms, *IEEE Trans. Syst., Man, Cybern.*, *SMC-9* (1), 1979, 62-66.

[9] S. Watanable and CYBEST Group, An automated apparatus for cancer processing: CYBEST, *Comp. Graph. Image processing, 3*, 1974, 350-358.

[10] P. K. Nanda & P. Kanungo, Parallelized Crowding Scheme Using a New Interconnection Model, *Proceedings of 2002 AFSS, International conference on Fuzzy Systems*, Calcutta, India, February, 2002, *LNAI 2275*, Springer-Verlag, 2002, 436-443.

[11] P. K. Nanda & P. Kanungo, Parallel Genetic Algorithm based Crowding Scheme Using Neighboring Net Topology, *Proceedings of sixth international conference on Information Technology*, Bhubaneswar, India, December 2003, 583-585.

[12] S. W. Mahfoud, Simple Analytical Models of Genetic Algorithms for Multimodal Function Optimization, *Technical Report, Illinois Genetic Algorithm Laboratory*, *IlliGAL report No. 93001*, Feb. 1993.

[13] S. W. Mahfoud, Cross over Interaction among Niches, *Proceedings of the First IEEE Conference on Evolutionary Computation, World Congress on Computation Intelligence,1994,* 188-193.