International Journal of the Physical Sciences Vol. 6(31), pp. 7203 - 7211, 30 November, 2011
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.1248
ISSN 1992 - 1950 ©2011 Academic Journals

Full Length Research Paper

Image thresholding based on evolutionary algorithms

N. Razmjooy[1], B. Somayeh Mousavi[2]*, P. Sargolzaei[3] and F. Soleymani[4]

[1]Department of Electrical Engineering, Majlesi Branch, Islamic Azad University, Isfahan, Iran.
[2]Department of Computer, Zahedan Branch, Islamic Azad University, Zahedan, Iran.
[3]Department of Mathematics, University of Sistan and Baluchestan, Zahedan, Iran.
[4]Department of Mathematics, Zahedan Branch, Islamic Azad University, Zahedan, Iran.

Accepted 26 September, 2011

The objective of this paper is to propose an adaptive-evolutionary method for thresholding which is used as an artificial intelligent algorithm for image segmentation especially for object segmentation. This method employs resistant versus mixed histograms because of its suitable fitness function selection that consists of the histogram details. As things develop in the paper, three evolutionary methods known as genetic algorithm (GA), imperial competitive algorithm (ICA) and adaptive particle swarm optimization are used to minimize the error function. Finally, a powerful algorithm for image thresholding is found. The comparisons and experimental results show that this system is better than other methods particularly Otsu's, GA and even new algorithms like ICA.

Key words: Segmentation, adaptive particle swarm optimization (APSO), genetic algorithm (GA), imperialist competitive algorithm (ICA), threshold, fitness function.

INTRODUCTION

Finding a suitable thresholding value to segment an arbitrary object from its background is an important step in image processing. The main step herein is maybe defect detections or any other operations. In more applications, image segmentation is included by the main step (Fu et al., 1981; Pal et al., 1993). For more information on related matters (Mirzaei et al., 2011; Sojodishijani et al., 2010; Logeswari et al., 2010). Explicitly, thresholding is to classify the given images into two portions (that is, object and background) in which one sector consists of pixels more than or equal to the proposed gray level and the other consists of pixels less than the proposed gray level; this type of thresholding is called bi-level threshold.

Ideally, for a bi-level image, the histogram consists of two peaks; in such condition, lowest point between these two peaks (valley) denotes the thresholding point. Figure 1 shows the histogram, its valley and the peaks in it.

The optimum point belongs to the lowest one. In this situation, the pixels with a minor value than the presented point will be returned to zero and the major ones that are equal or bigger than it will return to one (Fernandez et al., 2009).

*Corresponding author. E-mail: bbmoosavi@gmail.com.

Several image segmentation methods are defined formerly (Guo et al., 1998; Shaoo et al., 1998; Snyder et al., 1990). Common approaches for colour image segmentation are clustering algorithms such as K-means or mixture of principal components. However, these algorithms do not take spatial information into account. Some progress has been made on this issue; still, much experimentation needs to be done (Sahoo et al., 1988; Sezgin et al., 2004). In general, colour segmentation methods could be classified as:

1. Histogram thresholding (mode method) and color space clustering: Histogram thresholding is one of the widely used techniques for monochrome image segmentation. It assumes that images are composed of regions with different grey level ranges, the histogram of an image can be separated into a number of peaks (modes), each corresponding to one region, and there exists a threshold value corresponding to valley between the two adjacent peaks. As for color images, the situation is different from monochrome image, because of multi-features. Multiple histogram-based thresholding divided the color space by thresholding each component histogram. There are some limitations when dividing multiple dimensions, because thresholding is a technique for gray scale images. For example, the shape of the cluster is rectangle.

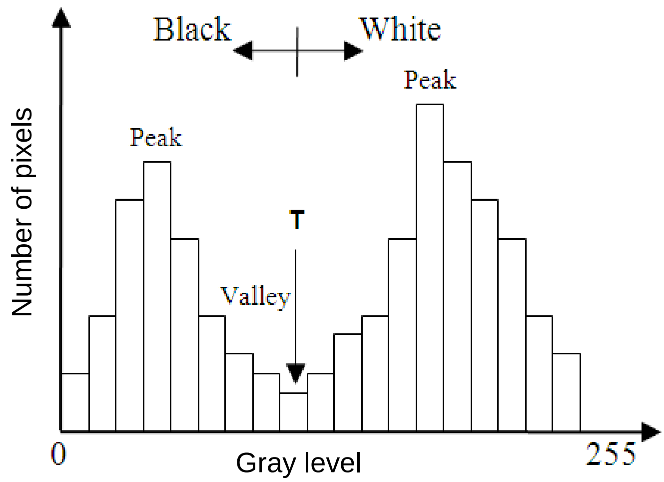

Figure 1. Histogram of a gray level image.

2. Region based approaches: Region based approaches, including region growing, region splitting, region merging and their combination, attempt to group pixels into homogeneous regions. The region based approach is widely used in color image segmentation because it considers the color information and spatial details at the same time.
3. Edge detection: In a monochrome image, edge is defined as a discontinuity in the gray level, and can be detected only when there is a difference of brightness between two regions. However, in color images, the information about edge is much richer than that in monochrome case. For example, edges between two objects with the same brightness but different hue can be detected in color images. Accordingly, in a color image, an edge should be defined by a discontinuity in a three-dimensional color space. It should be emphasized here that edge detection cannot segment an image by itself. It can only provide useful information about the region boundaries for the higher level systems, or it can be combined with other approaches, e.g. region based approaches, to complete the segmentation tasks.
4. Fuzzy techniques: The segmentation approaches mentioned earlier take crisp decisions about regions. Nevertheless, the regions in an image are not always crisply defined, and uncertainty can arise within each level of image analysis and pattern recognition. Fuzzy set theory provides a mechanism to represent and manipulate uncertainty and ambiguity. Fuzzy operators, properties, mathematics and inference rules (IF-THEN rules) have found considerable applications in image segmentation.
5. Neural networks approaches: Artificial neural networks (ANN) are widely applied for pattern recognition. Their extended parallel processing capability and nonlinear characteristics are used for classification and clustering. ANN explore many competing hypotheses simultaneously through parallel nets instead of performing a program of instructions sequentially, hence ANN can be feasible for parallel processing. Neural networks composed of many computational elements connected by links with variable weights. The complete network, therefore, represents a very complex set of interdependencies which may incorporate any degree of nonlinearity, allowing very general function to be modeled. Training time is usually very long, but after training, the classification using ANN is rapid.

Some of these methods are subjected into two important errors:

1. They have no standard image segmentation method whenever they object to a more class images.
2. Having a more class image is a reality that presents some tenacious problems.

Different methods were applied to have an optimal threshold. In this article, one algorithm is contributed consisting of genetic algorithm (GA), adaptive particle swarm optimization (APSO) and a new evolutionary that is called imperialist competitive algorithm (ICA) to acquire an optimize threshold to a bi-level histogram image; after getting the fitness function, it is essential to have a method for minimizing.

Typically, experiments show that evolutionary methods are good and include low cost for these purposes (Lai et al., 2000; Tseng et al., 1999).

Among the evolutionary methods, because of its population-based property, APSO minor variables and speedy rate computation is a proper algorithm for optimizing targets (Zhang et al., 2010). The structure of this algorithm is different from those of other algorithms, such as GA and ICA; it is a social method instead of competition, because, while the particles are weak being alone, they become powerful when they are together. The rest of paper is organized as follows. Subsequently, we present the proposed algorithm, and then the fitness function is applied, after which we commented on GA, ICA and adaptive PSO. This is followed by a characterization of the threshold point. Afterwards, the experimental and discussions are given. Finally, the conclusions are summarized.

PROPOSED ALGORITHM

Noise reduction

In image processing, it is frequently desirable to be able to achieve some kind of noise reduction on an image (Otsu, 1979). Noise reduction is basically carried out by temporal or spatial averaging techniques. Such noise reduction is a typical pre-processing step to improve the results of later processing (in our work, image thresholding). Gaussian low pass filtering (GLPF) is very vastly used in digital image processing, because under certain conditions, it preserves edges while removing noise (Arce, 2005). Mathematically, a Gaussian filter improves the input signal by convolution with a Gaussian function; which is also known as the Weierstrass transform.

The transfer function of a Gaussian low pass filter is given by:

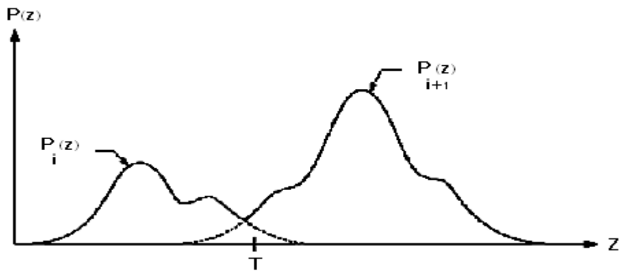

Figure 2. An unfit histogram of a gray level image.

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2} \quad (1)$$

$$D = \sqrt{u^2 + v^2}$$

where σ represents standard deviation. When applied in two dimensions, this formula produces a surface whose contours are concentric circles with a Gaussian distribution from the center point. Values from this distribution are used to build a convolution matrix which is applied to the original image. Each pixel's new value is set to a weighted average of that pixel's neighborhood. The original pixel's value receives the heaviest weight (having the highest Gaussian value) and neighboring pixels receive smaller weights as their distance to the original pixel increases (Shapiro et al., 2001)

Fitness function

Typically, manual thresholding (Kennedy et al., 1995) in this condition can make an unfit image that is not a proposed point. Even Otsu's method sometimes causes an improper thresholding as will be shown as the work progresses. Figure 2 shows an unfitted histogram of a gray level image.

whereas a gray-level distribution $[0,1,\cdots,L-$ is known for any vicinal of image. After trials, Fitness function is defined as follow:

$$Fitness = \sum_{i=1}^{L} \left| x - \frac{LENGTH(image)}{MEAN(Hist(image))} \right| \quad (2)$$

where *x* is the chromosome (in GA), *Hist(image)* is the histogram of each pixels of image from 1 to *L* in gray level and *LENGTH* is the length of image. It is essential to use an iterative method to solve this function (equation) and even to achieve a minimum point via gradient method and meeting initial conditions is needed.

EVOLUTIONARY ALGORITHMS

After finding a proper function for thresholding, it must be minimized by evolutionary algorithms; the applied algorithms are: GA, ICA and APSO, respectively which are explained as follows:

Genetic algorithm

Our genetic algorithm is an ad-hoc one based on a classical structure. The main steps are as follows (Tseng et al., 1999):

1. Create a random number as the first generation of *pop* chromosomes with *f* genes each one.
2. Resume the next steps while the number of generations *gen* is not reached:

a. Fitness: Measure the population, computing the fitness of every chromosome.
b. Selection: Select *pop*/2 chromosomes from the better adjusted chromosomes.
c. Crossover: Cross those *pop*/2 selected chromosomes in pairs, producing two new chromosomes by pair.
d. Turnover the *pop*/2 non-crossed chromosomes by the new ones produced by the crossover operation.
e. Mutation: Randomly mutate the population.
3. When *gen* is reached, return the best chromosome (the one with the best fitness throughout all generations).

Imperialist competitive algorithm

Imperialist competitive algorithm (ICA) is a new evolutionary optimization method which is derived from imperialistic competition. Like other evolutionary algorithms, ICA commences with an initial population which is known as the country; this country consists of two species of colonies and imperialists which together form empires (Gargari et al., 2007). Indeed, imperialist countries try to overcome other countries and turning them to their colonies. Also, imperialist countries compete strongly with each other for taking occupancy of other countries; imperialistic competition among these empires forms the proposed evolutionary algorithm. During this competition the weakest empire collapse and stronger ones will get more potency.

Imperialistic competition converges to a situation in which there exists only one empire and the colonies have the same fitness function and power value as the imperialist, as a piece of colony can be replaced by the imperialist.

The pseudo code of imperialist competitive algorithm is introduced as (Otsu, 1979):

1. Select some random points on the function and initialize the empires.
2. Move the colonies toward their relevant imperialist (Assimilation).
3. Randomly change the position of some colonies (Revolution).
4. If there is a colony in an empire which has lower cost than the imperialist, exchange the positions of that colony and the imperialist.
5. Unite the similar empires.
6. Compute the total cost of all empires.
7. Pick the weakest colony (colonies) from the weakest empires and give it (them) to one of the empires (imperialistic competition).
8. Eliminate the powerless empires.
9. If stop conditions is satisfied, stop, if not go to 2.

In ICA, the epithet country indeed is an array of variables that should be optimized by finding the final goal. In such condition, country is a $1 \times N$ array which is used to optimize an N dimensional problem. The floating point numbers is applied to the variable values of the country.

From the aforementioned, algorithm starts with some initial countries which are randomly distributed in search space. N number of stronger countries (countries with lower cost) is selected to be the imperialists and the others are divided among them based on their power. The initial number of colonies of each imperialist should be directly commensurate to its normalized power.

After dividing all colonies among imperialists and creating the initial empires, these colonies commence moving into their relevant imperialist territory which is based on assimilation policy (Arias-

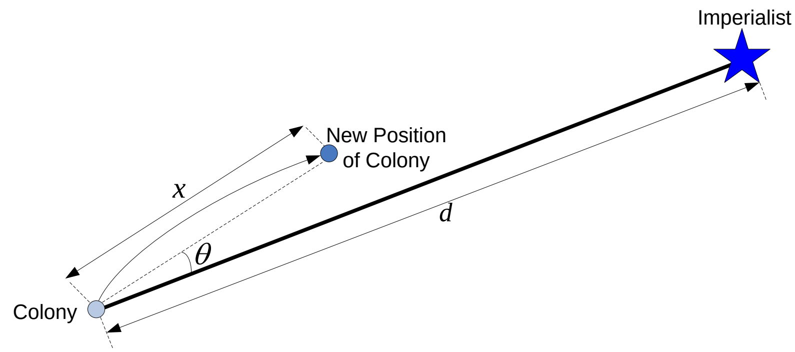

Figure 3. Colonies movement into their relevant imperialist.

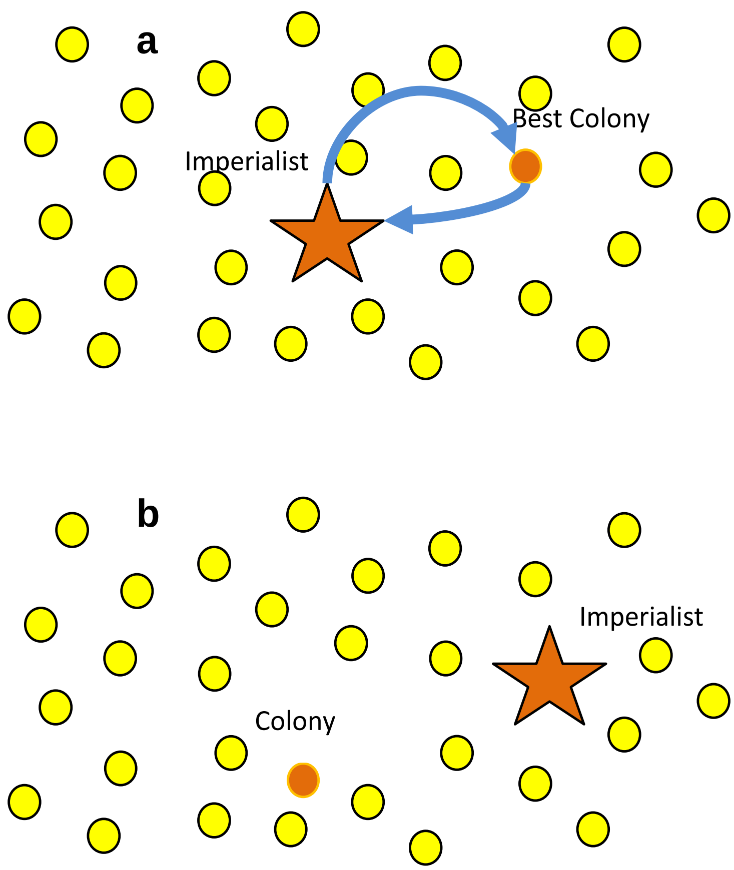

Figure 4. (a) Imperialist and colony position change, (b) Total empire after applying changes.

Castro et al., 2009). Figure 3 shows the motion of a colony into its relevant imperialist. In this motion, θ and x are random numbers with uniform distribution as demonstrated in Equation 3 and d is the distance between colony and the imperialist.

$$x \sim U(0, \beta \times d), \tag{3}$$

$$\theta \sim U(-\gamma, \gamma) \tag{4}$$

where β is a positive number less than 2, d is the space between the imperialist and its colony and γ order the derivation from the original direction.

Figure 4 shows that if this motivation causes the finding of a colony with better situation (lower cost) rather than its imperialist, the position of the colony and imperialist change together.

The total power of an empire depends on both the power of the imperialist country and the colonies, and is defined by:

$$T.C_n = Cost(imperialist_n) + \xi mean\{Cost(\text{colonies of } empire_n)\} \quad (5)$$

In imperialistic competition, all imperialists try to take the asset of other imperialists and develop their own power.

In ICA, this fact is defined as taking the total power of an empire by the sum of imperialist state power and a percentage of the mean power of its colonies. The total power of empire is defined as:

$$P_t = P_{im} + emean\{power(colonies)\}, \quad (6)$$

where P_{im} is the power of imperialist and e is a positive number less than 1. In such condition, powerless empires that cannot compete increase their power, while those that cannot even detain diminution of its power will be collapsed.

Imperialistic competition and the motion of colonies into their relevant imperialist will hopefully make a new world with one empire, such that all the colonies and even imperialist itself have the same position and power. In this term the imperialist contains an array of variables which is an optimal resolving of the problem.

Adaptive PSO

Due to the fact that the adaptive PSO is nonlinear and it has dynamic inertia in every iterative, it has a high ability for dynamic balancing between local and global searching, because of its adaptive type and less parameters, when compared with the other methods, such as ICA and/or GA.

James Kennedy, social psychologist and Russell Eberhart, electrical engineer presented this algorithm (Kennedy et al., 1995). Their first aim was to exploit social models and a social relation in the soft computing that does not depend on ellipse. Their first imagery done in 1995, led them to simulate birds behavior in seed finding. PSO algorithm is known as a random initial value applied to population to start analyzing.

In the standard PSO algorithm, each particle in a swarm population traces its status in the search space by tracking two maximum values that consist of the best position (Pbest) which is found in a distant place and the best fitted particle's position in the total number of populations.

Unlike other evolutionary algorithms, in PSO, each candidate depends on a value that was renowned as velocity. The resolvent candidate is called as the particle that searches the included searching space.

The velocity value is achieved by the basis of experiences and the use of other particle's experience adjustment as a constant value. Velocity updating frames v_{ij} of each particle and its position x_{ij} in standard PSO is presented as:

$$v_{ij}(t+1) = v_{ij}(t)\big(c_1 r_1 (Pbest_i(t) - x_{ij}(t)\big) + c_2 r_2 (Gbest_i(t) - x_{ij}(t)) \quad (7)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (8)$$

In the conventional PSO, particles that cannot become optimal, slowly shift to location of *Gbest*, therefore, the global optimal particle must actively detect new areas and update the *Gbest* to give motives to the search of other particles. In this work, APSO is used to improve the performance. It is used in different particles by various tasks to find the best of them. In this way, the particles with better performance are weighed by more inertia weight and the particles with poor performance are weighed by smaller one to get an optimal point in general.

Therewith, a large inertia weight simplifies a global search while a small inertia weight simplifies a local search. The particles are sorted according to their individual optimal location from the best to the poor. The inertia weight and positive constant of the particle, which is in *i*-th place, are presented as:

$$w_i = w_{\min} + (w_{\max} - w_{\min})^{(m-i)} \big/ (m-1) \quad (9)$$

$$c_{i1} = c_{i2} = (w_i + 1 + 2\sqrt{w_i}) \big/ 2 \quad (10)$$

In Equation 9, m is the population size; inertia weight function w_i is adjusted adaptively. The strategy can balance between global and local search in each iteration step.

APSO algorithm can be summarized as follows:

1. Generate an initial random position and velocity to particles.
2. Compute the fitness function of all particles; indeed, calculate the best position solution and the best personal velocity one.
3. Use levels 2 and 3 to update the position and velocity for each particle. Inertia weights and learning factors (c_{i1} and c_{i2}) are updated by levels 5 and 6.
4. Repeat levels 2 and 3 till stop criteria are met.

Applying the algorithm, single proper values are gained. The value of each algorithm is shown as follows:

GA:
Initial population = 100
Crossover coefficients= 0.8
Mutation coefficient (mutation is done in an un-uniform) = 0.04
Migration coefficient = 0.2

ICA:
β = 2
γ = 0.5

APSO:
Adaptive, almost nothing to change.

Threshold is the chromosome of GA, country of ICA and particle of APSO. Afterwards, the results of GA, ICA and APSO are compared together. Output change values in APSO and ICA should be better than those of GA. Of course, it is essential to know that in most times, this is a better system, but not for ever; this is caused because of its evolutionary congruity. Therefore, may be GA sometimes has better response.

The problem of GA in these responses is through its mutation; therefore it can be concluded that local optimization is better for image thresholding, and because of their global searching, changes in such mutations deflect thresholding system.

EXPERIMENTAL RESULTS

In this paper, a new method is applied to find threshold value based on adaptive and local histogram optimization function; some experiments are applied with various images from petitcolas which include different brightness and contrasts. Real images are: 259 × 259 to 600 × 608 in size.

Table 1. Accuracy of different methods (Accuracy: Number of valid segment images/Total number of images).

Method	Otsu (%)	GA (%)	APSO (%)
Accuracy	39.07	65.76	69.86

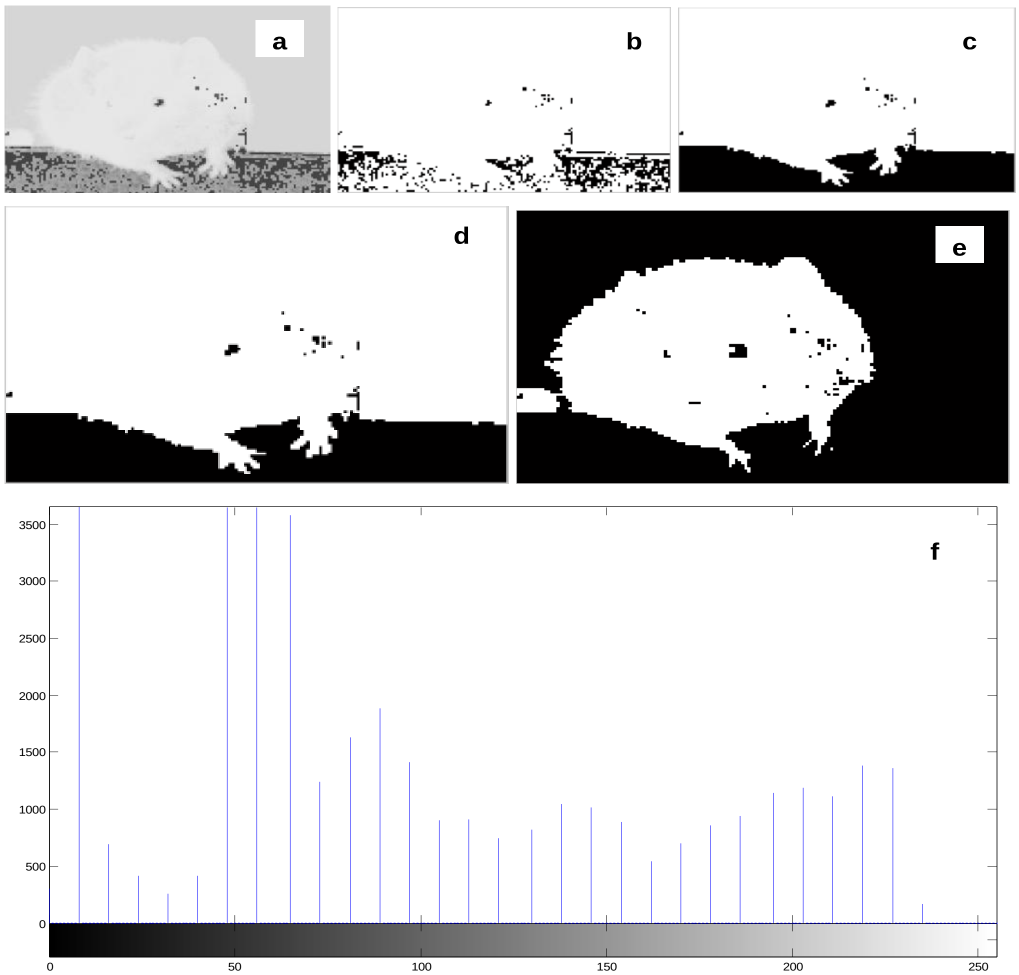

Figure 5. (a) mouse image, (b) Otsu segmentation, (c) Image segmentation by genetic algorithm minimizing (d) Image segmentation by imperialist competitive algorithm minimizing, (e) Image segmentation by adaptive PSO minimizing, (f) Histogram of the original image.

The image and its different thresholding methods (GA, ICA and APSO algorithms) are used to solve the error function to have a minimum error. Results show that local thresholding evolutionary algorithms have better results into global ones and hence, APSO is a proper algorithm for such thresholdings.

After all, we tried Otsu's method (Yu, 2008). GA thresholding (by the aforementioned function), ICA and APSO thresholding were used to present the proposed methods' ability. The first image shows that Otsu's method does not have the proposed methods' ability, but GA is better than Otsu. However, ICA is also not bad, but APSO has better performance. The second and third images have the same results, and in fourth and fifth images, Otsu method has rather good results but it is not the best. The poor result of Otsu's method is because it is very sensitive to the object's shadow.

The findings affirmed the robustness, fast convergence and proficiency of APSO algorithm over other existing techniques. Table 1 shows the results of some databases.

Figures 5 to 7 show the results of the final thresholding on some samples; however, valid segments are determined for human validations.

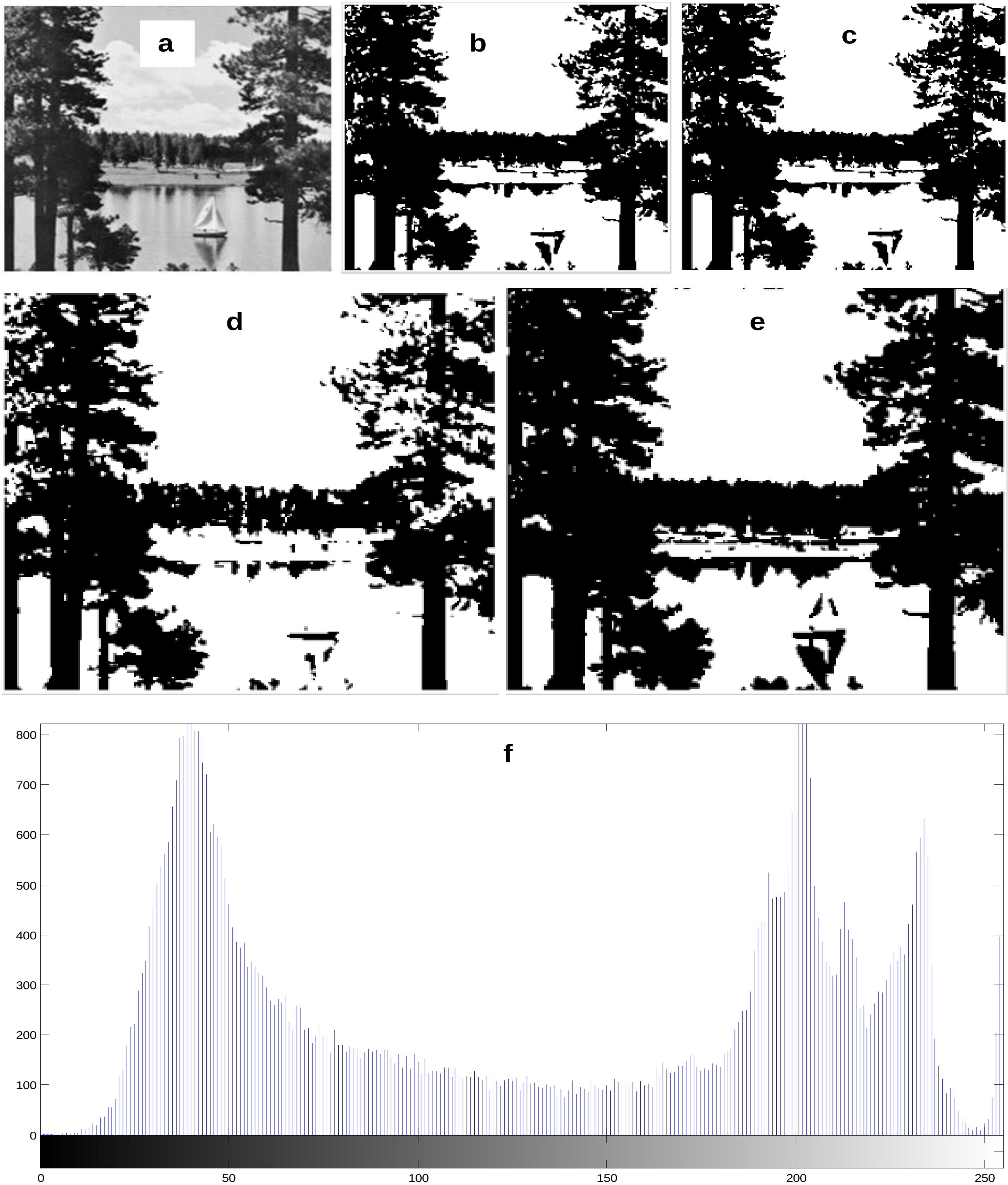

Figure 6. (a) Scene image**,** (b) Otsu segmentation, (c) Image segmentation by genetic algorithm minimizing (d) Image segmentation by imperialist competitive algorithm minimizing, (e) Image segmentation by adaptive PSO minimizing, (f) Histogram of the original image.

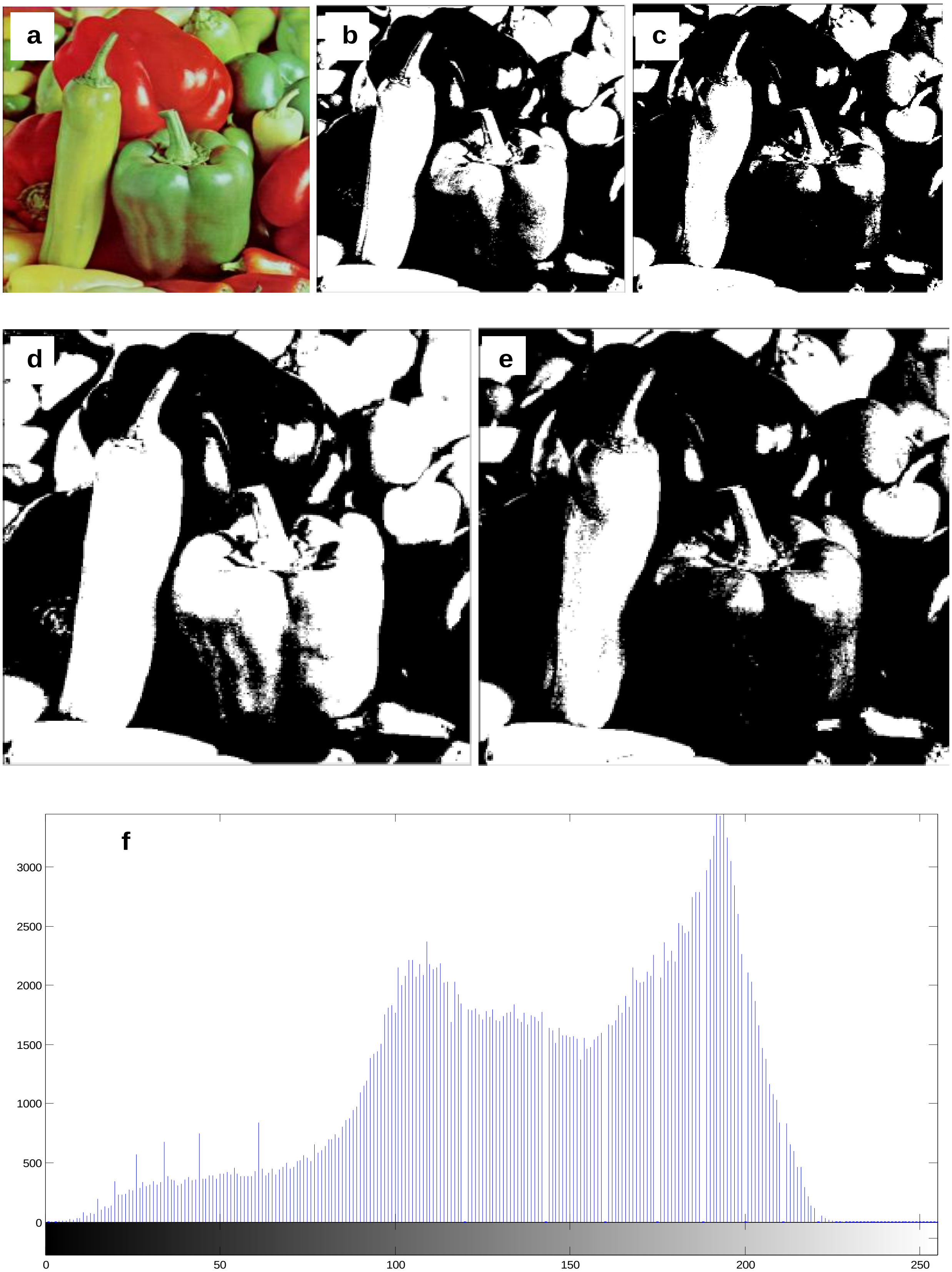

Figure 7. (a) Pepper image, (b) Otsu segmentation, (c) Image segmentation by genetic algorithm minimizing (d) Image segmentation by imperialist competitive algorithm minimizing, (e) Image segmentation by adaptive PSO minimizing, (f) Histogram of the original image.

Conclusions

Thresholding is one of the most important topics in machine vision and image processing segmentation practically, finding a suitable method for thresholding is more difficult when it has an overlap on histogram (Sathya and Kayalvizhi, 2010). thresholding approach, based on adaptive particle swarm optimization algorithm, was proposed. We assumed that the intensity distributions of objects and backgrounds in an image obey Gaussian probability functions. Afterwards, the histogram of a given image is fitted by a mixture of Gaussian probability functions. The adaptive particle swarm optimization is then used to estimate the requested parameters so that the error function reaches to its minimum amount. The experimental results specify that the proposed approach can give an appropriate result.

The described method in this paper is capable to be tuned for any optimization approaches, color segmentation or object detection applications.

REFERENCES

Arias-Castro E, Donoho DL (2009). Does median filtering truly preserve edges better than linear filtering? Annals of Statistics. 37(3): 1172–2009.

Fernandez YE, Sarinana A, Swenson RL (2009). development of a prototype for classification of potato mini-tubers based on artificial vision. 6th International Conference on Electrical Engineering, Computing Science and Automatic Control.

Fu KS, Mui JK (1981). A Survey on Image segmentation Pattern Recognition, 13: 3-16.

Gargari E, Atashpaz- Lucas C (2007). Imperialist Competitive Algorithm: An Algorithm for Optimization Inspires by Imperialistic Competition. 2007 IEEE Congress on Evolutionary Computation, Singapore.

Guo R, Pandit SM (1998). Automatic threshold selection based on histogram modes and discriminant criterion. Machine Vision and Applications, p. 10.

Kennedy J, Eberhart RC (1995). Particle swarm optimization. International Conference on Neural Networks, 4: 1942–1948.

Lai CC, Tseng DC (2000). An optimal L-filter for reducing blocking artifacts using genetic algorithms. Signal Process, 81(7): 1525-1535.

Logeswari T, Karnan M (2010). An improved implementation of brain tumor detection using segmentation based on soft computing. J. Cancer Res. Exp. Oncol., 2(1): 006-014.

Mirzaei K, Motameni H, Enayatifar R (2011). New method for edge detection and de noising via fuzzy cellular automata. Int. J. Phy. Sci., 6(13): 3175-3180.

Otsu N (1979). A threshold selection method from gray-level histograms. IEEE Trans. Systems, Man Cybern., 9(1): 62-66.

Pal NR, Pal SK (1993). A Review on Image Segmentation Techniques. Pattern Recognition. 26(9): 1277-1294.

Sahoo PK, Soltani S, Wong AKC (1988). A Survey of Thresholding Techniques. Comput. Vis. Graph. Image Process, 41: 133-260.

Sathya PD, Kayalvizhi R (2010). PSO-Based Tsallis Thresholding Selection Procedure for Image Segmentation. Int. J. Comput. Appl., 5(4): 39-46.

Shapiro LG, Stockman GC (2001). Computer Vision. Pretence Hall. pp. 137, 150.

Sezgin M, Sankur B (2004). Survey over image thresholding techniques and quantitative performance evaluation. J. Electronic Imaging, 13(1): 146-165.

Snyder W, Bilbro G, Logenthiran A, Rajala S (1990). Optimal thresholding: A new approach. Pattern Recognit. Lett., 11: 803-810.

Sojodishijani O, Rostami V, Ramli AR (2010). A video-rate color image segmentation using adaptive and statistical membership function. Sci. Res. Essays, 5(24): 3914-3925.

Tseng DC, Lai CC (1999). A genetic algorithm for MRF-based segmentation of multispectral textured images. Pattern recognition. Lett., 20(14): 1499-1510.

Yu L (2008). Vehicle extractions using histograms and genetic algorithm based fuzzy image segmentation from high resolution UAV aerial imagery. The International Archives of the Photogrammetric, Remote Sens. Spat. Inform. Sci., 37: 529-534.