Proceedings of the World Congress on Engineering 2007 Vol I
WCE 2007, July 2 - 4, 2007, London, U.K.

Image Enhancement Using Particle Swarm Optimization

Malik Braik, Alaa Sheta *and Aladdin Ayesh †

Abstract—**Applications of the Particle Swarm Optimization (PSO) to solve image processing problem with a reference to a new automatic enhancement technique based on real-coded particle swarms is proposed in this paper. The enhancement process is a non-linear optimization problem with several constraints. The objective of the proposed PSO is to maximize an objective fitness criterion in order to enhance the contrast and detail in an image by adapting the parameters of a novel extension to a local enhancement technique. The feasibility of the proposed method is demonstrated and compared with Genetic Algorithms (GAs) based image enhancement technique. The obtained results indicate that the proposed PSO yields better results in terms of both the maximization of the number of pixels in the edges and the adopted objective evaluation. Computational time is also relatively small in the PSO case compared to the GA case.**

Keywords: particle swarm optimization, genetic algorithms, image enhancement

1 Introduction

Particle Swarm Optimization (PSO) is one of the modern heuristic algorithms that can be applied to non linear and non continuous optimization problems. It is a population-based stochastic optimization technique for continuous nonlinear functions [1]. PSO was developed in 1995 by Dr. James Kennedy, a social psychologist, and Dr. Russell Eberhart, an electrical engineer [2]. PSO term refers to a relatively new family of algorithms that may be used to find optimal (or near optimal) solutions to numerical and qualitative problems. It is easily implemented in most programming languages and has proven both very effective and quick when applied to a diverse set of optimization problems [2, 3]. PSO was discovered through simulation of a simplified bird flocking model. Dr. Kennedy and Dr. Eberhart stated in [2] "Particle swarm optimization has roots in two main component methodologies. Perhaps more obvious are its ties to artificial life (A-life) in general, and to bird flocking, fish schooling, and swarming theory in particular. It is also related, however, to evolutionary computation, and has ties to both Genetic Algorithms (GAs) and Evolutionary Programming (EP)." Unlike GAs and EP, PSO is a simple concept and is very easy to implement. The developers of PSO stated in [2] "Particle swarm optimization as developed by [Kennedy and Eberhart] comprises a very simple concept, and paradigms can be implemented in a few lines of computer code. It requires only primitive mathematical operators, and is computationally inexpensive in terms of both memory requirements and speed. Early testing has found the implementation to be effective with several kinds of problems...Particle swarm optimization has also been demonstrated to perform well on genetic algorithm test functions." PSO shares many similarities with GAs.

Many authors explored the use of PSO to solve variety of problems in computer science and engineering [4, **?**, 5]. The use of PSO to solve various problems in pattern recognition and image processing was presented in [6]. In [8], author used PSO to estimate model parameters for software fault detection and diagnosis. Online training algorithm of a Generalized Neuron (GN) was developed using PSO in [9]. Particle swarm optimization for image registration was introduced in [10]. This is why it was quite challenging to adjust or tune the PSO parameters such that the required goals are achieved. An empirical study on the setting of control coefficients in PSO was presented in [11]. When should we use swarm to solve problems was explored in [12]. In this paper, a real-coded PSO is applied to adapt the gray-level intensity transformation in the image. The fitness of each image is taken as a swarm particle and its subjective score is given by the human interpreter.

2 Local Enhancement Model

Local enhancement model apply transformation functions that are based on the gray-level distribution in the neighborhood of each pixel in the given image. In the traditional enhancement technique, the original equation shown in "(1)," is applied to each pixel at location (i, j) using the following transformation [13]:

$$g(i,j) = [\frac{M}{\sigma(i,j)}][f(i,j) - m(i,j)] \tag{1}$$

*Information Technology Department, Prince Abdullah Bin Ghazi Faculty of Science and Information Technology, Al-Balqa Applied University, Jordan, Email: malik@bau.edu.jo, sheta@bau.edu.jo

†Computer Engineering Division, De Montfort University, Leicester, UK. Email: aayesh@dmu.ac.uk

The mean and standard deviation are computed in a neighborhood centered at (i, j). Therefore, they are dependent on the local information. $f(i, j)$ and $g(i, j)$ are the gray-level intensity of pixels in the input and output image, respectively, centered at location (i, j). And lastly, M is the global mean of the image.

3 Proposed Enhancement Model

There are two key steps when applying PSO to optimization problems:

1. The representation of the solution and
2. The fitness function.

The proposed enhancement model is derived from "(1)," and is applied to each pixel at location (i, j) using the following transformation [14]:

$$g(i,j) = [K\frac{M}{\sigma(i,j)+b}][f(i,j)-c*m(i,j)]+m(i,j)^a \quad (2)$$

a, b, c, and k are the parameters defined over the real positive numbers and they are the same for the whole image. Comparing "(1)," to "(2)," the values of the parameters are taken as constants (i.e. $b = 0, c = 1, k = 1,$) and the term $m(i, j)^a$ is taken as 0. In "(2),"$b \neq 0$" prohibits the Not A Number (NAN) values, $c \neq 1$ allows for only a fraction of the mean to be subtracted from the pixel's input gray-level intensity value, while the last term may have brighten and smooth the effects on the image. Accordingly, the proposed equation broadened the spectrum of the transformation output range by modifying the original equation.

PSO task is to solve the image enhancement problem by tuning the four parameters in order to find the best combination according to an objective criterion that describes the contrast in the image. The parameters a, b, c, and k are represented as particles. Each particle represents a candidate solution to solve the optimal enhancement problem. Therefore, the representation of the particle is a string of four real swarms denoting the forth dimension.

The proposed method using PSO has many advantages.

1. It uses a local enhancement technique based on the standard deviation and the mean value of the pixels.
2. It has no interaction with the humans.
3. It uses an objective fitness criterion that is proportional to the number of edges in the image and to a clumping factor of the intensity transformation curve.

3.1 Enhancement Criterion

One of the requirements of the PSO based image enhancement is to choose a criterion that is related to a fitness function. The proposed technique needs the enhanced image to have a relatively high intensity of the edges. Consequently, the fitness criterion is proportional to the number and intensities of the pixels in the edges that might give an over-sized credit to an image that doesn't have a natural contrast.

In fact, we need from the fitness criterion a histogram of the image that should approach the uniform distribution as shown in Figure 1 [13]. The fitness function shown in "(3)," [14] is a good choice for an enhancement criterion:

$$F(Z) = \log(\log(E(I(Z)))) * \frac{n\_edgels(I(Z))}{M * N} * H(I(Z)) \quad (3)$$

$F(Z)$ is the fitness function. $I(Z)$ denotes the original image I with the transformation T applied according to "(1),". The parameters a, b, c, and k are the respective parameters given by the particle $Z = (abck)$. $E(I(Z))$ is the intensity of the edges detected with a Sobel edge detector that is applied to the transformed image $I(Z)$, n_edgels is the number of edgel pixels as detected with the Sobel edge detector. The Sobel detector used here is an automatic threshold detector [15]. M and N are the number of pixels in the horizontal and vertical direction of the image. $E(I)$ is the sum of intensities of the edges included in the enhanced image [16]. Lastly, $H(I(z))$ measures the entropy of the image $I(z)$.

The proposed PSO objective is to find the solution that maximizes $F(Z)$. To achieve these objectives we need to:

1. Increase the relative number of pixels in the edges of the image.
2. Increase the overall intensity of edges, and
3. Transform the histogram of the image to one that approximates a uniform distribution by maximizing the entropic measure [17].

4 PSO Algorithm

PSO is initialized with a group of random particles (solutions). The algorithm then searches for optima through a series of iterations. The particle's fitness value is evaluated on each iteration. If it is the best value the particle has achieved, the particle stores the location of that value as *pbest* (particle best). The location of the best fitness value achieved by any particle during any iteration is stored as *gbest* (global best) [18, 19, 20]. Using *pbest*

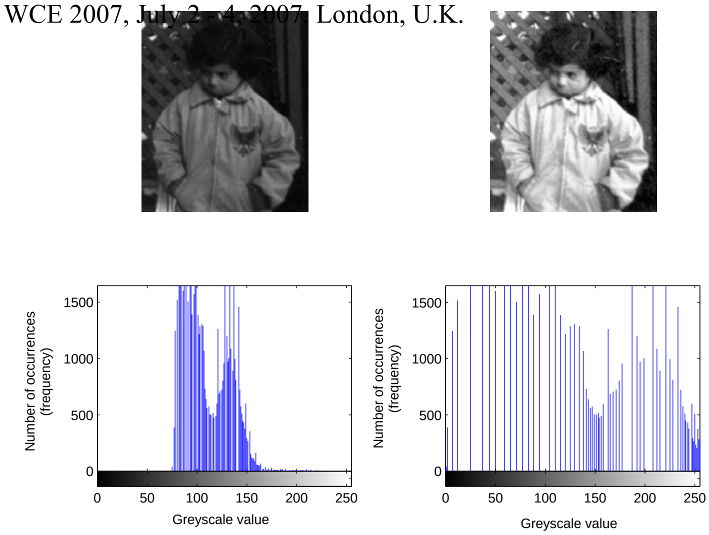

Figure 1: Global Histogram Equalization, Upper left: Unequalized image, Upper right: Same image after histogram equalization, Lower left: Unequalized histogram, Lower right: Equalized global histogram.

and *gbest*, each particle moves with a certain velocity, calculated by Equations 4, 5, and 6 [2, 18].

$$\begin{aligned} V_i = wV_{i-1} \quad &+ \quad c_1 * rand() * (pbest - pL) \\ &+ \quad c_2 * rand() * (gbest - pL) \end{aligned} \tag{4}$$

$$pL = pvL + V_i \tag{5}$$

$$w = \frac{1}{iterNum} \tag{6}$$

V_i is the current velocity, V_{i-1} is the previous velocity, pL is the present location of the particle, pvL is the previous location of the particle, rnd is a random number between $(0, 1)$, $c1$ and $c2$ are learning factors or stochastic factors, and $iterNum$ is the current iteration number. The pseudo code of the PSO procedure was presented in [2, 20] and is given in Figure 2.

```
For each particle
    Initialize particle
For each particle
    Calculate fitness value
    If the fitness value is better than the best fitness
    value (pbest) in history
    set current value as the new pbest
End
Choose the particle with the best fitness value of all
the particles as the gbest
For each particle
    Calculate particle velocity according to Eq. 4
    Update particle position according to Eq. 5
End
Continue while maximum iterations or minimum error
criteria is not attained
```

Figure 2: The Pseudo code of the PSO procedure

5 Genetic Algorithms

At each generation, each individual is evaluated and recombined with others on the basis of its fitness. The expected number of times an individual is selected for recombination is proportional to its fitness relative to the rest of the population. New individuals are created using crossover and mutation.

- Crossover operates by selecting a random location in the genetic string of the parents (crossover point) and concatenating the initial segment of one parent with the final segment of the second parent to create a new child. A second child is simultaneously generated using the remaining segments of the two parents.
- Mutation provides for occasional disturbances in the crossover operation by inverting one or more genetic elements during reproduction [21, 22, 23].

The pseudo code of the standard GAs is as shown in Figure 3 [24, 23]:

```
Begin GA
g=0 generation counter
Initialize population
Evaluate population P(g) i.e., compute fitness values
While not done do
    g=g+1
    Select P(g) from P(g-1)
    Crossover P(g)
    Mutate P(g)
    Evaluate P(g)
End while
End GA
```

Figure 3: The Pseudo code of the GAs procedure

6 PSO and GA Control Parameters

In the objective enhancement criterion we need to find the solution of the fitness function $F(z)$ where a, b, c, and k are set to be the swarms.

The following combinations of the control parameters are used for running PSO. The number of particles is 30. Dimension of particles is four since the parameters need to be tuned are 4. Range of particles is the positive real numbers. The maximum change one particle can take during one iteration is 20. Learning factors or acceleration constants are equal to 1.3. The searching is a repeat

ISBN:978-988-98671-5-7

process and the stop condition or the maximum number of iterations the PSO executes is set to 200. Inertia weight is set at 0.6 and 0.9. Using the above control parameters, the PSO is executed and the results are obtained.

The following combinations of the control parameters are used for running GAs. The chromosome structure had four parameters to be estimated. The selection mechanism of using GAs is binary tournament and K-elitism with K = 5. GAs was used with population size, crossover probability and mutation probability of 1000, 0.9, 0.03, respectively.

7 Comparison between GAs and PSO

Most of the evolutionary techniques have the following procedure:

1. Random generation of an initial population.
2. Reckoning of a fitness value for each subject.
3. Reproduction the population based on fitness values.
4. If the requirements are met, then stop. Otherwise go back to 2.

From the above procedure, we can learn that PSO shares many common points with GAs [1, 20]. Both GAs and PSO are initialized with a population of random solutions and search for the optimum by updating generations. Both have fitness values to evaluate the population. However, the information sharing mechanism in PSO is significantly different [1, 20, 25].

- In GAs, each possible solution within the population of a biological individual is coded in so called chromosome. Chromosomes share information with each other. Each chromosome is assigned a fitness score according to how good a solution to the problem based on a given fitness function. The solutions are taken according to their fitness values and used to construct new solutions by a hope that the new solutions will be better than the old solutions and a generation is complete. Thus, the whole population moves like a one group towards an optimal area [23, 21, 24, 22].
- In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Only the particle with the best fitness value of all the particles gives out the information to others, so it is a one-way information sharing mechanism, where the evolution only looks for the best solution. Then, all the particles tend to converge to the best solution quickly even in the local version in most cases. In GAs, the new solutions are created using two main evolution operators known as crossover and mutation. However, PSO does not have the evolution operators and particles update themselves with the internal velocity [20, 25].

8 Results and Discussion

The optimization problem considered in this paper is to solve the enhancement problem using PSO. Our objective is to maximize the number of pixels in the edges, increase the overall intensity of the edges, and increase the measure of the entropy. After that, the histogram of the enhanced image approaches the required uniform distribution. In order to evaluate the PSO-based enhancement method, we compared the proposed method with GA-based enhancement method using four selected images. They are the Cameraman, Tire, Pout and House. The size of each image is varying. For example, the Cameraman has a 256x256 pixels.

For each PSO or GA run we report three values:

- The performance of the algorithms by computing the objective evaluation function in terms of the fitness value
- The computational time per run of each algorithm
- The efficiency in terms of the number of edgels which gives an indication of the performance of the proposed algorithm.

The objective evaluation criterion in terms of fitness score is employed to rank the proposed method; the results are given in Table 1 for typical runs. It can be shown that the results obtained using PSO when compared with the results obtained using GA reveals the following fact:

- The fitness value using PSO is more when compared with the fitness value using GAs for the same number of generations.
- The computational time for PSO based enhancement was found 130.5 seconds whereas the time taken for GAs based enhancement was found 182.4 seconds.
- The computational time is less in case of PSO when compared with that of GAs since PSO does not perform the selection and crossover operations as in the GA case.
- The image that contains the highest number of edgel pixels can be rated as having high detail contents as shown in Table 2.

It is clear from Table 2 that the PSO-based method achieves the best detail content in the enhanced images

ISBN:978-988-98671-5-7

Table 1: The fitness value of both PSO and GAs Using 200 Generations

Image/Fitness	PSO-based	GAs-based
Cameraman	128.821	102.988
Tire	136.398	130.030
Pout	10.450	2.972
House	250.345	240.342

Table 2: The number of edgels as detected with Sobel automatic edge detector

Image	Original	GA	PSO
Cameraman	2485	2575	2674
Tire	1823	1917	2020
Pout	1492	2040	2048

when compared with the number of edgels in the enhanced image using GAs and both are greater than the number of edgels in the original image. This ensures that the PSO method yields better quality of solution compared to GAs. Thus, the above facts reveal the superior properties of PSO when compared with GAs. So, the proposed PSO method yields high quality solutions with better computation efficiency. It can be shown from Figure 4, that the brightness and contrast of the enhanced images using PSO and GAs appear visibly and is more than the brightness and contrast of the original images. Also, it can be shown clearly, that the brightness of the enhanced images using PSO is better than the brightness of the enhanced images using GAs. The convergence process of the 4 images is shown in Figure 5.

PSO has been successfully applied for image enhancement application and demonstrated that PSO gets better results in a faster, cheaper way compared with GA evolutionary method. Also PSO is more attractive than GA is that there are few parameters to adjust compared with the large number of parameters adjusted when GA is run. All in all, these reported values and the results shown in Figure 4 give a good explanation of the superior of using PSO for image enhancement compared to GAs.

9 Conclusions and Future Work

In this paper, a new approach to automatic image enhancement using real-coded PSO is implemented by specifying a suitable fitness function proportional to the number and intensity of the edgel pixels and to the entropic measure of the image. The objective of the algorithm was to maximize the total number of pixels in the edges thus being able to visualize more details in the images. The algorithm is tested on four selected images. The results obtained are tabulated and compared with the results ob-

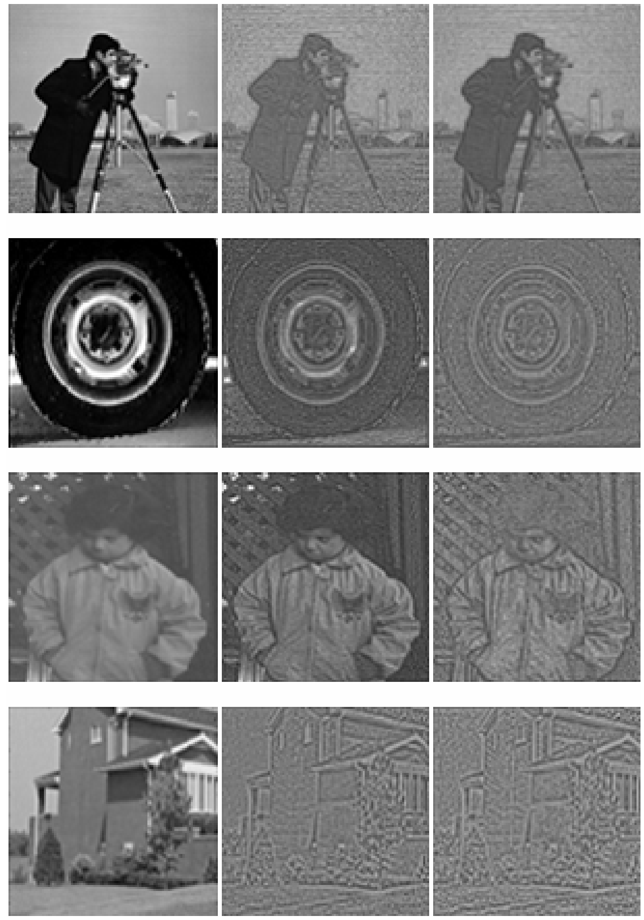

Figure 4: Enhancement results: left-original image; middle-GA based method; right-PSO based method. For the images a) Cameraman b) Tire c) Pout d) House.

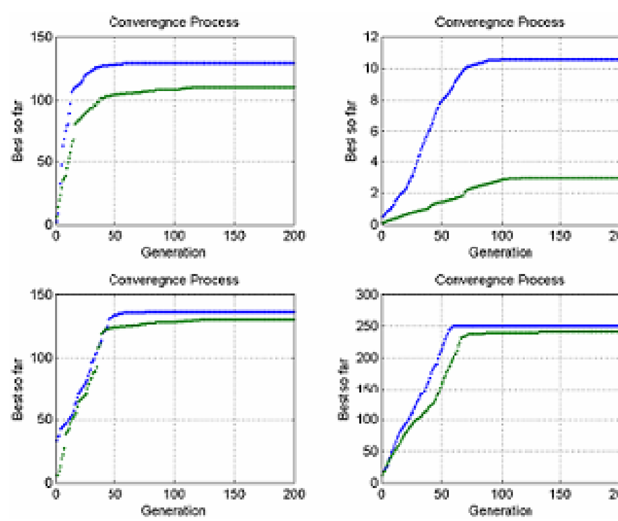

Figure 5: Convergence process of the tested images-blue color, PSO-based method; green color, GA-based method; upper left-Cameraman; upper right-Tire; lower left-Pout; lower right-House.

tained using GAs. It is clear from the obtained results that the proposed PSO based image enhancement is better than the GAs based image enhancement in terms of quality solution and computational efficiency. The proposed PSO based image enhancement method may be extended in several ways, such as: fine tuning of the PSO parameters in order to reduce the number of particles and reducing the maximum number of iterations. Another extension is to code local parameters of the method that applies to each neighborhood.

References

[1] J. Kennedy, R. C. Eberhart, and Y.Shi, *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco, 2001.

[2] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ*, vol. 5, no. 3, pp. 1942–1948, 1995.

[3] M. A. Talal and A. A. Mohamed, "Simulation-based optimization for repairable systems using particle swarm algorithm," Proceedings of the 2005 Winter Simulation Conference, Department of Statistics and Operations Research Kuwait University, 2005.

[4] H. Sun, Y. Pan, and Y. Zhang, "Pso based gabor wavelet feature extraction method," in *Proceedings of the International Conference on Information Acquisition, 21-25 June 2004*, pp. 422 – 425, 2005.

[5] J.-I. Lin and T.-Y. Cheng, "Dynamic clustering using support vector learning with particle swarm optimization," in *Proceedings of the 18th International Conference on Systems Engineering*, pp. 218–223, 2005.

[6] Y.-C. Chen, H.-C. Wang, and T.-J. Su, "Particle swarm optimization for image noise cancellation," in *First International Conference on Innovative Computing, Information and Control - Volume I (ICICIC'06)*, pp. 587–590, 2006.

[7] M. G. H. Omran, *PSO methods for pattern recognition and image processing*. PhD thesis, University Pretoria, November, 2004.

[8] A. Sheta, "Reliability growth modeling for software fault detection using particle swarm optimization," pp. 10428–10435, Proceedings of IEEE Congress on Evolutionary Computation (CEC 2006), Vancouver, BC, Canada, 2006.

[9] R. Kiran, S. R. Jetti, and G. K. Venayagamoorthy, "Online training of generalized neuron with particle swarm optimization," pp. 10187–10194, Proceedings of IEEE Joint Conference on Neural Networks, Vancouver, BC, Canada, 2006.

[10] H. Talbi and M. Batouche, "Particle swarm optimization for image registration," in *Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 19-23 April 2004*, pp. 397–398, 2004.

[11] N. M. Kwok, D. K. Liu, K. Tan, and Q. P. Ha, "An empirical study on the setting of control coefficient in particle swarm optimization," pp. 3165–3172, Proceedings of IEEE Congress on Evolutionary Computation (CEC 2006), Vancouver, BC, Canada, 2006.

[12] T. J. Richer and T. M. Blackwell, "When is a swarm necessary?," pp. 5618–5625, Proceedings of IEEE Congress on Evolutionary Computation (CEC 2006), Vancouver, BC, Canada, 2006.

[13] R. Gonzalez, R. Woods, and S. Eddins, *Digital Image Processing using* MATLAB. Upper Saddle River, NJ Jensen: Prentice Hall,2nd Edition, 2004.

[14] C. Munteanu and A. Rosa, "Towards automatic image enhancement using genetic algorithms," *LaSEEB-ISR-Instituto Superior Tcnico*, 2001.

[15] P. L. Rosin, "Edges: saliency measures and automatic thresholding, machine vision and applications," *Springer Verlag*, vol. 9, pp. 139–159, 1997.

[16] J. S. DaPonte and F. M. D., "Enhancement of chest radiographs with gradient operators," *IEEE Transactions on Medical Imaging*, vol. 7, no. 2, pp. 109–117, 1988.

[17] A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice Hall, 1991.

[18] Y. Zheng, L. Ma, L. Zhang, and J. Qian, "On the convergence analysis and parameter selection in particle swarm optimization," pp. 1802–1807, Proceedings of International Conference, Machine Learning and Cybernetics, 2003.

[19] K. Yasuda, A. Ide, and N. Iwasaki, "Adaptive particle swarm optimization," pp. 1554–1559, Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 2003.

[20] Y. Zheng, L. Ma, L. Zhang, and J. Qian, "Empirical study of particle swarm optimizer with an increasing inertia weight," pp. 221–226, Proceedings of IEEE Congress on Evolutionary Computation (CEC 2003), Canbella, Australia, 2003.

[21] J. Holland, *Adaptation in Natural and Artificial Systems*. PhD thesis, University of Michigan Press, Ann Arbor, 1975.

[22] K. DeJong, *An Analysis of Behavior of a Class of Genetic Adaptive Systems, Doctoral dissertation*. PhD thesis, University of Michigan, Dissertation Abstracts International, 1975.

[23] D. Goldberg, "The design of innovation: Lessons from and for competent genetic algorithms," *Addison-Wesley, Reading, MA*, 2002.

[24] D. Goldberg, "Genetic algorithms in search, optimization and machine learning," *Kluwer Academic Publishers, Boston, MA*, 1989.

[25] H. K. Dong, "Improvement of genetic algorithm using PSO and euclidean data distance algorithm," *International Journal of Information Technology*, vol. 12, no. 3, 2006.