Object Detection In Image Using Particle Swarm Optimization

Ankit Sharma
Electronics Instrumentation & Engineering Department
ankit.sharma2012@gmail.com

Nirbhowjap Singh
Electronics Instrumentation & Engineering Department
nirbhow@gmail.com

***Abstract*— Image matching is a key component in almost any image analysis process. Image matching is crucial to a wide range of applications, such as in navigation, guidance, automatic surveillance, robot vision, and in mapping sciences. Any automated system for three-dimensional point positioning must include a potent procedure for image matching. Most biological vision systems have the talent to cope with changing world. Computer vision systems have developed in the same way. For a computer vision system, the ability to cope with moving and changing objects, changing illumination, and changing viewpoints is essential to perform several tasks. Object detection is necessary for surveillance applications, for guidance of autonomous vehicles, for efficient video compression, for smart tracking of moving objects, for automatic target recognition (ATR) systems and for many other applications. Cross-correlation and related techniques have dominated the field since the early fifties. Conventional template matching algorithm based on cross-correlation requires complex calculation and large time for object detection, which makes difficult to use them in real time applications. The shortcomings of this class of image matching methods have caused a slow-down in the development of operational automated correlation systems. In the proposed work *particle swarm optimization* & its variants based algorithm is used for detection of object in image. Implementation of this algorithm reduces the time required for object detection than conventional template matching algorithm. Algorithm can detect object in less number of iteration & hence less time & energy than the complexity of conventional template matching. This feature makes the method capable for real time implementation. In this thesis a study of particle Swarm optimization algorithm is done & then formulation of the algorithm for object detection using PSO & its variants is implemented for validating its effectiveness.**

Keywords- Cross Correlation Co-efficient, Particle swarm optimization, Pattern Recognition, Template Matching.

I. INTRODUCTION

As humans, it is easy (even for a child) to detect the position of the letters, objects, numbers, etc. However, making a computer solve these types of problems in fast manner is a very challenging task. Object detection is a fundamental component of artificial intelligence and computer vision. Object detection methods are used in various areas such as science, engineering, medical applications, etc. Interest in pattern recognition is fast growing in order to deal with the prohibitive amount of information, we encounter in our daily life. Automation is desperately needed to handle this information explosion. Object detection is necessary for surveillance applications, for guidance of autonomous vehicles, for efficient video compression, for smart tracking of moving objects, for automatic target recognition (ATR) systems and for many other applications. In the last decades the computer's ability to perform huge amount of calculations, and handle information flows we never thought possible ten years ago has emerged. Despite this a computer can only extract little information from the image in comparison to human being. The way the human brain filters out useful information is not fully known and this skill has not been merged into computer vision science. The aim of this thesis is to implement a system in MATLAB that is able to faster detection of object in an image. The system should use both fast and advanced algorithms aiming to achieve the exact position of the object in image. The goal is to develop a system with the potential to be implemented in a real time environment. Therefore the system needs to be very fast.

II. LITERATURE REVIEW

In [4] a new feature based correspondence algorithm for image matching was proposed. In [5] a procedure for digital image correlation is described which is based on least squares window matching. In [6] the method for fast normalized cross-correlation is proposed. In [7] the problem of crafting visual routines for detection tasks is address. In [8] a generic matching algorithm suitable for many applications where feature extraction is difficult or inaccurate is presented. Least squares template matching algorithm is proposed. In [14] a new method for locating object based on valley field is proposed. In [15] problem of locating an object in an image when size and rotation are unknown are examined. In [16] the method of image matching by normalized cross-correlation is proposed. In [17] a novel method which is robust and efficient in extracting objects using Wavelets and Neural Networks is suggested. In [18] the generalized projection function (GPF) is defined. In [19] a novel approach to pattern matching is proposed in which time complexity is reduced by two orders of magnitude compared to traditional approaches. In [20] a robust eye detection algorithm for gray intensity images is presented. The idea of their method was to combine the respective advantages of two existing techniques, Feature based method and template based method, and to overcome their shortcomings.

III. Template Matching & Cross Correlation Cofficient

Template matching is a popular method for pattern recognition. It is defined below:

Definition: Let I be an image of dimension m×n and T be another image of dimension p×q such that p<m and q<n then template matching is defined as a search method which finds out the portion in I of size p×q where T has the maximum cross correlation coefficient with it.

The normalized cross correlation coefficient is defined as:

$$\lambda(x,y) = \frac{\sum_s \sum_t \delta_{I(x+s,y+t)} \delta_{T(s,t)}}{\sum_s \sum_t \delta^2_{I(x+s,y+t)} \sum_s \sum_t \delta^2_{T(s,t)}} \quad (1)$$

Where,

$\delta_{I(x+s,y+t)} = I(x+s,y+t) - I(x,y)$, $\delta_{T(s,t)}$ = T(s,t)-T

s € {1, 2, 3,…..p}, and t € {1, 2, 3,…..q}.

x € {1, 2, 3,……m-p+1}, y € {1, 2, 3,……n-q+1}

$$I(x,y) = \frac{1}{pq} \sum_s \sum_t I(x+s,y+t)$$

$$T = \frac{1}{pq} \sum_s \sum_t T(s,t)$$

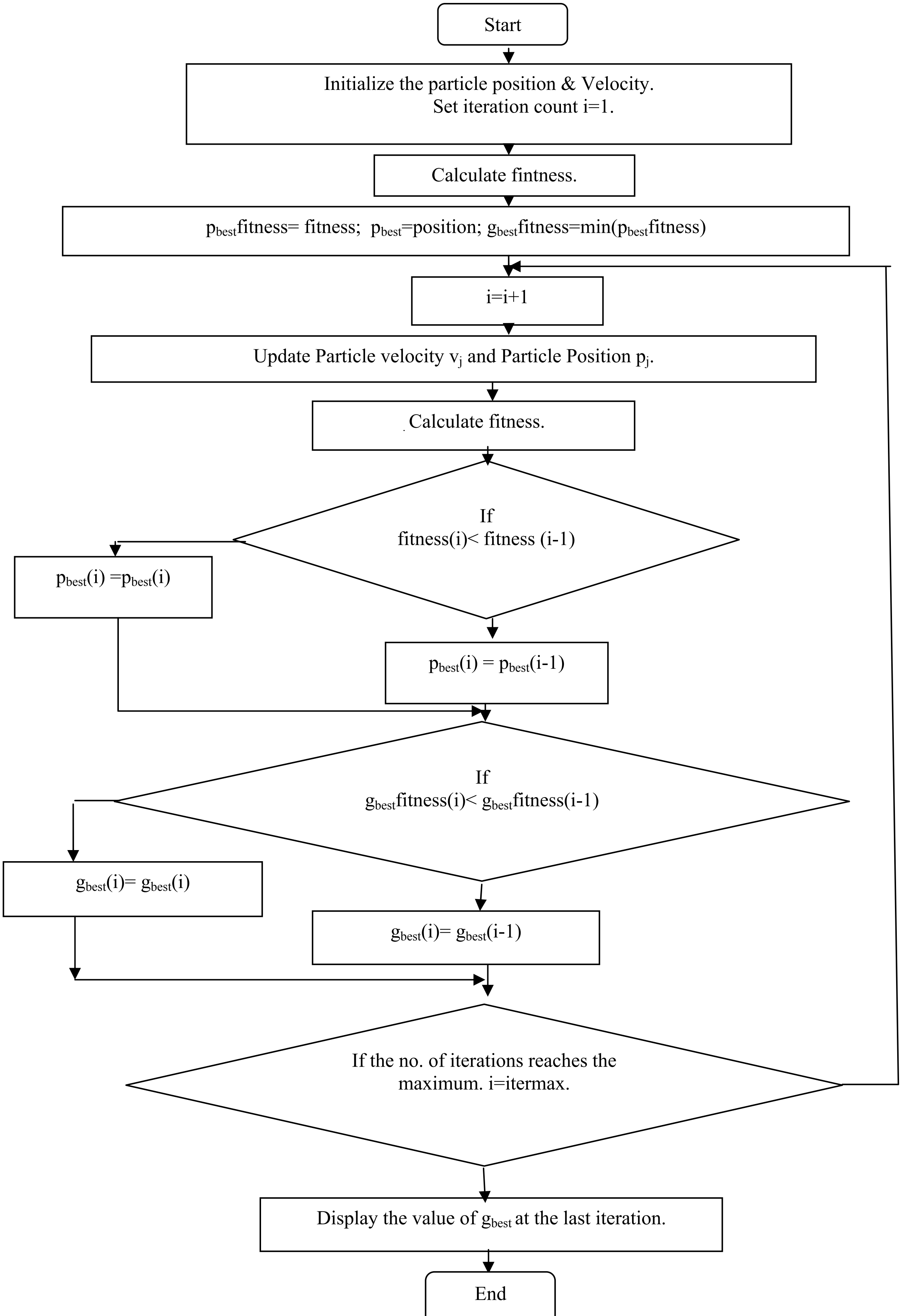
The value of cross-correlation coefficient γ ranges in [-1, +1]. A value of +1 indicates that T is completely matched with I(x, y) and -1 indicates complete disagreement. For template matching the template, T slides over I and γ is calculated for each coordinate (x, y). After completing this calculation, the point which exhibits maximum γ is referred to as the match point.

IV. Particle Swarm Optimization

Particle swarm optimization is a stochastic, population-based search and optimization algorithm for problem solving. It is a kind of swarm intelligence that is based on social-psychological principles and provides insights into social behavior, as well as contributing to engineering applications. The particle swarm optimization algorithm was first described in 1995 by *James Kennedy and Russell C. Eberhart*. The techniques have evolved greatly since then, and the original version of the algorithm is barely used at present. Social influence and social learning enable a person to maintain cognitive consistency. People solve problems by talking with other people about them, and as they interacts their beliefs, attitudes, and behavior changes, the changes could typically be depicted as the individuals moving toward one another in a socio-cognitive space.

The particle swarm simulates a kind of social optimization. A problem is given, and some way to evaluate a proposed solution to it exists in the form of a fitness function. A communication structure or social network is also defined, assigning neighbors for each individual to interact with a population of individuals defined as random guesses as the problem solutions is initialized. These individuals are candidate solutions and are also known as the particles, hence the name particle swarm. An iterative process to improve these candidate solutions is set in motion. The particles iteratively evaluate the fitness of the candidate solutions and remember the location where they had their best fitness value. The individual's best solution is called the particle best or the local best. Each particle makes this information available to their neighbors. They are also able to see where their neighbors have had best fitness value. Movements through the search space are guided by these successes, with the population usually converging, by the end of a trial, on a problem solution better than that of non-swarm approach using the same methods.

The particle swarm optimization (PSO) algorithm is a population-based search algorithm inspired by the social behavior of birds within a flock. The initial intent of the particle swarm concept was to graphically simulate the graceful and unpredictable choreography of a bird flock, the aim of discovering patterns that govern the ability of birds to fly synchronously, and to suddenly change direction with a regrouping in an optimal formation. From this initial objective, the concept evolved into a simple and efficient optimization algorithm. In PSO, individuals, referred to as particles, are "flown" through hyper dimensional search space. Changes to the position of particles within the search space are based on the social-psychological tendency of individuals to emulate the success of other individuals. The changes to a particle within the swarm are therefore influenced by the experience, or knowledge, of its neighbors. The search behavior of a particle is thus affected by that of other particles within the swarm therefore PSO is the kind of symbiotic cooperative algorithm. The consequence of modeling this social behavior is that the search process is such that particles stochastically return toward previously successful regions in the search space. The operation of the PSO is based on the neighborhood principle as social network structure.


Start
Initialize the particle position & Velocity.
Set iteration count i=1.
Calculate fintness.
pbestfitness= fitness; pbest=position; gbestfitness=min(pbestfitness)
i=i+1
Update Particle velocity vj and Particle Position pj.
Calculate fitness.
If
fitness(i)< fitness (i-1)
pbest(i) =pbest(i)
pbest(i) = pbest(i-1)
If
gbestfitness(i)< gbestfitness(i-1)
gbest(i)= gbest(i)
gbest(i)= gbest(i-1)
If the no. of iterations reaches the
maximum. i=itermax.
Display the value of gbest at the last iteration.
End

V. Experimental Results & Discussion

The algorithms of particle swarm optimization & Predator-prey approach are applied for solving object detection problem. The performance has been studied for different images and different templates. The results are discussed as –

A. Test Images & Templates

Different test images and templates on which the developed algorithms are tested are shown below:

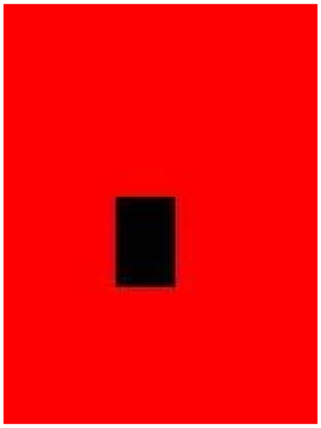


Image (a)

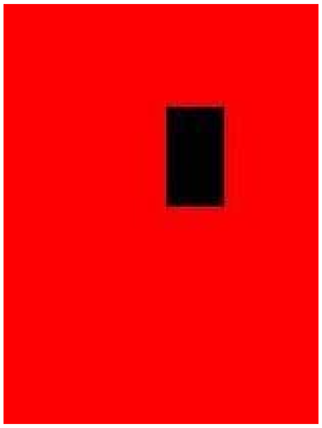


Image (b)

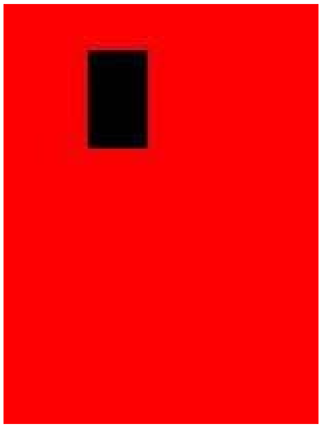


Image (c)

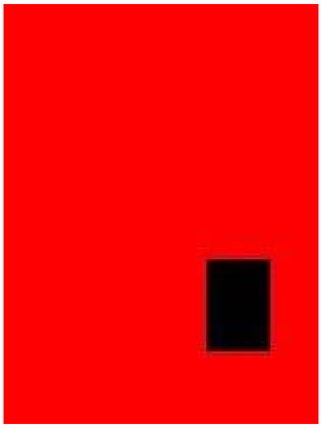


Image (d)

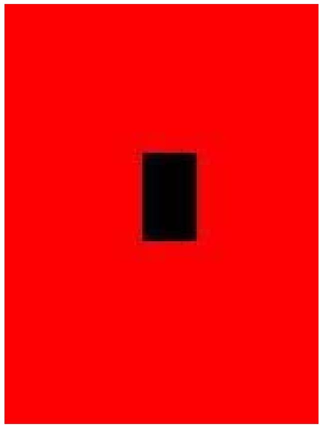


Image (e)

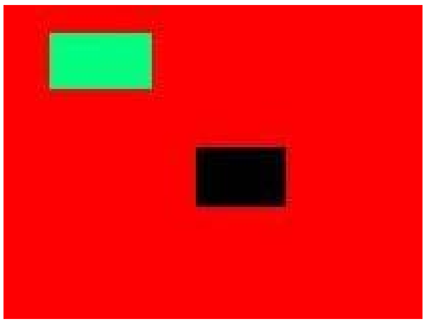


Image (f)

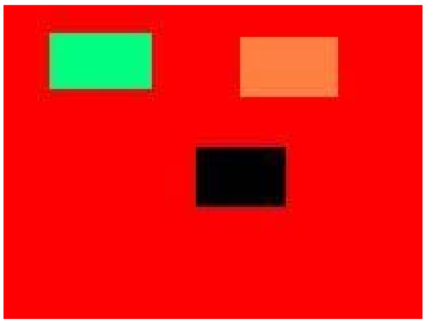


Image (g)

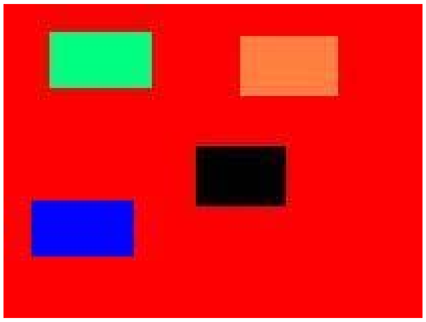


Image (h)

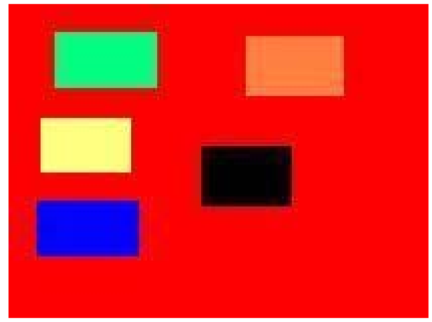


Image (i)

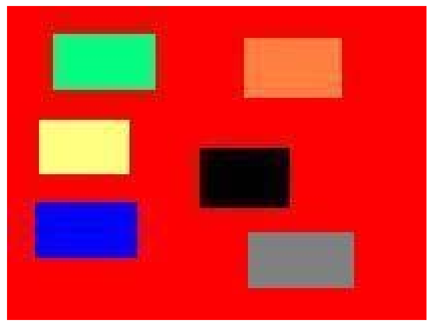


Image (j)

Fig. 1 Different test images

B. PSO Based Technique

The proposed PSO based technique was implemented in MATLAB and run under Pentium-4 (2.79 GHz) machine with 512 MB of RAM. Each test image is tested ten times by PSO based object detection program.

Image	PSO Based Algorithm						Old Algorithm
	Minimum Iteration Taken	Minimum Time Taken (In sec.)	Maximum Iteration Taken	Maximum Time Taken (In sec.)	Average Iteration Taken	Average Time Taken (In sec.)	Time Taken (In sec.)
(a)	7	0.18	131	2.03	38	0.60	3.85
(b)	11	0.17	65	1.00	32	0.50	5.56
(c)	11	0.17	90	1.39	33	0.5	2.87
(d)	18	0.27	170	2.63	51	0.81	7.33
(e)	18	0.27	74	1.14	48	0.70	4.19
(f)	3	0.04	73	1.16	32	0.50	5.70
(g)	4	0.06	130	2.01	38	0.60	5.70
(h)	15	0.23	82	1.27	45	0.70	5.70
(i)	10	0.15	152	2.35	56	0.8	5.70
(j)	6	0.09	82	1.27	44	0.60	5.70

VI. CONCLUSION

In the proposed work, we have successfully employed the PSO based methods to solve the object detection problem. The PSO based algorithm has superior features, including high-quality solution, stable convergence characteristic and good computation efficiency. The results show that the proposed method is capable of obtaining higher quality solution efficiency. It is clear from the results that the proposed PSO based method can avoid the shortcoming of old template matching algorithm and can provide higher quality solution with better computation efficiency.

When the test images are tested on PSO based algorithm for detecting the position of object then it is found that algorithms are capable of detecting the position of object in image with very less time as compared to conventional template matching algorithm. In conventional template matching algorithm the time taken for detection of abject is nearly 5 sec., while in the proposed algorithms the time consumed in the process of object detection is reduced by 10 times.

VII. FUTURE SCOPE OF WORK

Algorithms based on PSO proposed in this thesis for object detection can be implemented in wide range of applications, e.g. to navigation, guidance, automatic surveillance, robot vision, and to the mapping sciences. Any automated system for three-dimensional point positioning can use proposed algorithms for object detection. For a computer vision system, the ability to cope with moving and changing objects, changing illumination, and changing viewpoints is essential to perform several tasks. Object detection is necessary for surveillance applications, for guidance of autonomous vehicles, for efficient video compression, for smart tracking of moving objects, for automatic target recognition (ATR) systems and for many other applications. In all of the above application the proposed algorithm can provide faster.

VIII. REFERENCES

[1] T. Peli, “An algorithm for recognition and localization of rotated and scaled objects”, Proceedings of the IEEE 69 (1981) 483–485.

[2] T. Peli, “An algorithm for recognition and localization of rotated and scaled objects”, Proceedings of the IEEE 69 (1981) 483–485.

[3] A. Goshtasby, S.H. Gage, and J.F. Bartholic, “A Two-Stage Cross Correlation Approach to Template Matching”, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 6, no. 3, pp. 374-378, 1984.

[4] FOERSTNER, W., “Quality assessment of object location and point transfer using digital image correlation techniques. International Archives of Photogrammetry and Remote Sensing” vol. XXV, A3a, Commission III, Rio de Janeiro, 1984.

[5] ACKERMANN, F. 1984. “Digital image correlation: Performance and potential application in photogrammetry”. Photogrammetric Record 11 (64):429-439.

[6] J. P. Lewis, “Fast normalized cross-correlation”, Canadian Image Processing and Pattern Recognition Society, Quebec City, Canada, May 15-19, 1995.

[7] J. Bala, K. DeJong, J. Huang, H. Vafaie, H. Wechsler, “Visual routine for eye detection using hybrid genetic architectures,” International Conference on Pattern Recognition, vol. 3, pp. 606-610, 1996.

[8] Martin Berger, “The framework of least squares template matching.”, BIWI-TR-180, 1998.

[9] Angeline, P. J., “Evolutionary optimization versus particle swarm optimization: philosophy and performance differences”, The Seventh Annual Conf. on Evolutionary Programming, 1998.

[10] Shi, Y. and Eberhart, R. C., “Parameter selection in particle swarm optimisation” Evolutionary Programming VII: Proc. EP 98. New York, pp. 591-600, 1998.

[11] Shi, Y. and Eberhart, R. C., “Empirical study of particle swarm optimisation”, Proceedings of the 1999 Congress on Evolutionary Computation. Piscataway, NJ, pp. 1945-1950, 1999.

[12] Kennedy, J., “Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance”, Proc. Congress on Evolutionary Computation 1999. Piscataway, NJ, pp. 1931-1938, 1999.

[13] Kennedy, J., Eberhart, R. C., and Shi, Y., “Swarm intelligence”, Morgan Kaufmann Publishers, San Francisco. 2001.

[14] Kwan-Ho Lin, Kin-Man Lam and Wan-Chi Siu, "Locating the Eye in Human Face Images Using Fractal Dimensions," IEE Proceedings - Vision, Image and Signal Processing, vol. 148, no. 6, pp. 413-421, 2001.

[15] R.M. Dufour, E.L. Miller, N.P. Galatsanos, “Template matching based object recognition with unknown geometric parameters,” IEEE Transactions on Image Processing, vol. 11, issue.12, pp. 1385 – 1396, 2002.

[16] Feng Zhao, Qingming Huang, Wen Gao, “Image matching by normalized cross-correlation”. IEEE International Conference on Acoustics Speed and Signal Processing Proceedings, 2006.

[17] Mukesh Motwani, Rakhi Motwani, and Frederick C. Harris, Jr. “Eye Detection using wavelets and ANN,” GSPx 2004.

[18] Z.-H. Zhou and X. Geng, “Projection functions for eye detection,” Pattern Recognition, 37(5), pp. 1049-1056, 2004.

[19] Yacov Hel-Or, Hagit Hel-Or, “Real-Time Pattern Matching Using Projection Kernels”, IEEE transactions on pattern analysis and machine intelligence, Vol. 27, No. 9, September, 2005.

[20] Kun Peng, Liming Chen, Su Ruan, Georgy Kukharev, “A Robust and Efficient Algorithm for Eye Detection on Gray Intensity Face,” Lecture Notes in Computer Science – Pattern Recognition and Image Analysis, pp. 302-308, 2005.

[22] Kun Peng, Liming Chen, Su Ruan, Georgy Kukharev, “A Robust and Efficient Algorithm for Eye Detection on Gray Intensity Face,” Lecture Notes in Computer Science – Pattern Recognition and Image Analysis, pp. 302-308, 2005.

[21] Yacov Hel-Or, Hagit Hel-Or, “Real-Time Pattern Matching Using Projection Kernels”, IEEE transactions on pattern analysis and machine intelligence, Vol. 27, No. 9, September, 2005.