2009 Third UKSim European Symposium on Computer Modeling and Simulation

High Dimensional Problem Optimization Using Distributed Multi-agent PSO

Jalal A. Nasiri $^{\alpha*}$, Amin Milani Fard $^{\beta}$, Mahmoud Naghibzadeh $^{\alpha}$, Modjtaba Rouhani $^{\alpha}$

Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran $^{\alpha}$
School of Computing Science, Simon Fraser University, BC, Canada $^{\beta}$
*Commucation and Computer Research Center **
j.nasiri@wali.um.ac.ir, milanifard@cs.sfu.ca, naghibzadeh@ferdowsi.um.ac.ir,
m.rouhani@ieee.org

Abstract - *Curse of dimensionality is a major difficulty with the classic optimization methods for high dimensional applications in which the problem size grows rapidly and mostly exponential with the number of space. In this work we present a simple yet effective multi-agent approach to apply distributed particle swarm optimization to meet such demand. Lip detection in color images, as a high-dimensional problem, has been investigated and a novel approach for obtaining an optimized lip-map was proposed. Experimental results show 92% correction rate which is 11% increase in comparison to the simple approach. A computational complexity analysis also shows the superiority of the proposed architecture to be used in other large scale application.*

Keywords: *High dimensionality, Multi-agent systems, Particle Swarm Optimization, Lip detection.*

1 Introduction

Optimization problems have always been a matter of concern specially when dealing with high dimensionality. A major difficulty with the classic solutions of such problems is the curse of dimensionality in which the problem size grows rapidly and mostly exponential with the number of space dimensions. Evolutionary algorithms such as evolutionary programming have been studied in the domain of high-dimensional problems (up to 1000-dimensions) [1]. The usual effective mechanism for handling high dimensional problem optimization is cooperative co-evolution with the following divide-and-conquer strategy:

1) *Problem decomposition*: Splitting the object vectors into some smaller subcomponents.

2) *Optimize subcomponents*: Evolve each subcomponent with a certain optimizer separately.

3) *Cooperative combination*: Combine all subcomponents to form the whole system.

With respect to this strategy, problem decomposition, optimizer selection, and subcomponents cooperation are the tree crucial issues. Cooperative co-evolution architecture was firstly proposed by Potter for genetic algorithms (GAs) [2] and then successfully applied to other methods such as evolutionary programming (EP) [1], evolution strategy (ES) [3], and particle swarm optimization [4]. Multi-objective evolutionary algorithms (MOEAs) are stochastic, population-based computational procedures mimicking evolutionary concepts in an attempt to find the global optima of problems with multiple objectives. [5] Parallel multi-objective evolutionary algorithms (pMOEAs) are also used to find optimal solution for multi-objective problems in which the fitness functions are computationally expensive and therefore parallel function decomposition comes handy.

1.1 Contributions

Distributed processing provides us with simultaneous and effective use of distributed computational and informational resources. In this work we focused on constructing a multi-agent distributed system for parallel optimization of high-dimensional problems to reach sooner convergence. We then applied our distributed PSO model for detecting lips in color images as a sample of a high dimensional problem. We also provide a computational complexity analysis to show the effectiveness of our work.

The structure of the paper is as follows: Section 2 briefly describes previous related work, our distributed agent-based optimization method is presented in section 3, and lip detection as a high dimensional optimization problem and experimental result is dedicated to section 4. We finalize our work with a computational complexity comparison in section 5 and a conclusion part in section 6.

978-0-7695-3886-0/09 $26.00 © 2009 IEEE
DOI 10.1109/EMS.2009.107

2 Related Work

Generally real-world multi-objective optimization problems have high computational cost of multiple objective function evaluations which naturally leads to the consideration of parallel and distributed processing. Some optimization methods are parallel by their natural properties and each computational component can be regarded as an independent part. Thus parallel and distributed computing can assign each part to one of the parallel/distributed processors. This idea has been extensively applied to optimize other problems. In [5] a master-slave approach to parallelize PSO was mentioned. With only one master processor tracking the particles and collecting simulations results, the particles are distributed to clusters if slave processors which perform the evaluations. MOEAs use various evolutionary operations to find satisfactory solution, if not the real optimal one. Evolutionary algorithms (EAs) and MOEAs are adaptive stochastic search techniques in the soft computing domain [6]. Generic EAs such as GAs, ES, EP, and genetic programming (GP) are all successfully extended to MOEAs implementations [7]. Authors in [5] presented pMOEA symbolic formulations, describe pMOEA design and implementation issues, propose options for satisfactory resolution, and discuss various practical considerations. The result is a generic design plan with a list of parameter considerations, which researchers should consider in designing and implementing efficient and effective pMOEAs, regardless of their application problem domain.Contributions

3 Proposed Agent-based Optimization

Particle Swarm Optimization (PSO) [8] is a population based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. The system is initialized with a population of random solutions and searches for optima by updating generations. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has found currently. It can be taken as a particle that flying in the n-dimensional space. In order to find the optimal solution, each particle adjusts its flying velocity according to its own flying experience and its companions' flying experience at each iteration. Hence each particle has a position (x_{id}) and a speed (v_{id}) in the multidimensional space. The particle position indicates the possible solution in the multidimensional space and the speed indicates the amount of change between the current and next positions. The algorithm stores the previous best position of each particle (p_{id}) and the best global position of particles (p_{gd}) [8].

$$v_{id} = wv_{id} + c_1 * rand1 * (p_{id} - x_{id}) + c_2 * rand2 * (p_{gd} - x_{id})$$

$$if\,|v_{id}| > v_{\max} \qquad v_{id} = sign(v_{id} v_{\max}) \qquad (1)$$

$$x_{id} = x_{id} + v_{id} \qquad (2)$$

Where w is the inertia weight and c1 and c2 are the acceleration constants. PSO has been successfully applied in many research and application areas. The main reason for choosing PSO as a basis for our work is that there are few parameters to adjust. One version, with slight variations, works well in a variety of applications.

PSO is parallel by its nature since each particle can be regarded as an independent agent. Thus parallel and distributed computing can easily exploit this property and assign each agent to one of the parallel/distributed processors. Compared to the computational time in the fitness evaluations, the calculation of particles' positions and velocities represents a small fraction of the entire simulation. Due to the distributed property of our problem and also need for parallel computing to fasten processing, we decided to use multi-agent architecture.

3.1 Multi-agent system for heterogenous environment

Nowadays Grid technology [9] provides us with simultaneous and effective use of distributed computational and informational resources. We equipped our distributed agent-based PSO with a communication mechanism to work well in Grid based heterogeneous environment which is suitable for many high-dimensional applications. Our multi-agent system architecture is based on *Java Agent DEvelopment* (JADE) framework [10]. JADE is a software development framework aimed at developing multi-agent systems and applications in which agents communicate using FIPA[1] Agent Communication Language (ACL) messages and live in containers which may be distributed to several different machines. JADE uses RMI[2] method for communication. One of the most important characteristics of this tool is that programmer is not required to handle variables and functions concurrency as it is done automatically by the system.

JADE is capable of linking Web services and agents together to enable semantic web applications. A Web service can be published as a JADE agent service and an agent service can be symmetrically published as a Web service endpoint. Invoking a Web service is just like invoking a normal agent service. Web services' clients can also search for and invoke agent services hosted within JADE containers. The Web Services Integration Gateway (WSIG) [11] uses a Gateway agent to control the gateway from within a JADE container. Interaction among agents on different platforms is achieved through the Agent Communication Channel. Whenever a JADE agent sends a message and the receiver lives on a different agent platform, a Message Transport Protocol (MTP) is used to implement lower level message delivery procedures [12]. Currently there are two main MTPs

[1] Foundation for Intelligent Physical Agents (http://www.fipa.org)

[2] Remote Method Invocation

to support this inter-platform agent communication - CORBA IIOP-based and HTTP-based MTP.

Considering high-dimensional applications over separated networks, agent communications has to be handled behind firewalls and Network Address Translators (NATs), however, the current JADE MTP do not allow agent communication through firewalls and NATs. Fortunately, the firewall/NAT issue can be solved by using the current JXTA implementation for agent communication [13]. JXTA is a set of open protocols for P2P networking. These protocols enable developers to build and deploy P2P applications through a unified medium [14]. Obviously, JXTA is a suitable architecture for implementing MTP-s for JADE and consequently JADE agent communication within different networks can be facilitated by incorporating JXTA technology into JADE [13].

In our simple proposed architecture, we use five different types of agents, each having its own characteristics as the followings:

1) Manager Agent (MA): The MA has the responsibility of managing the whole system including other agent's creation. The creation node determination is influenced by problem size.

2) Problem Distributor Agent (PDA): This agent will distribute the problem search space to Swarm Agents.

3) Swarm Agent (SA): SAs are used to optimize sub-problems using PSO approach.

4) Local Optimizer Agent (LOA): There are several LOAs, each of which associated with a group of SAs, which select best solution among them and also let SAs know the best of each group.

5) Global Optimizer Agent (GOA): This agent has the responsibility to select best solution received by LOAs and let them know the best answer. LOAs then send this globally best result to its associated SAs.

In the next section, we show how our distributed PSO model can be used for detecting lips in color images as a sample of a high dimensional problem. We also provide a computational complexity analysis in the section 5 to show the effectiveness of our work.

4 Lip Detection as a High Dimensional Problem

Lip Detection is perhaps one of the most critical preprocessing steps in many human-oriented applications such as speech recognition, lip reading, audio interaction, and dental application [15]-[16]. For this matter researchers have used either RGB or Segmentation approaches for lip detection.

In RGB approach [17] the image is transformed by a linear combination of red, green and blue chrominance components of the RGB color space. To highlight the details of the lip envelop a high pass filter is then applied to the transformed image. The two generated images are then converted to obtain a binary image. The largest object in this binary image is recognized as the lip. The skin region identification has been used to separate the non-skin holes from skin regions in order to extract facial features from the image [18]. A thresholding box is then created and by which the image is searched line by line for the region to meet the certain criteria. In [19] a modified version of the predictive validation technique was mentioned that allows the use of the full covariance matrices used to select the model parameters. Then a Bayesian rule which is based on a subsequent grouping of the mixture components is used to recognize each pixel as lip or non-lip. Another technique is the hybrid edge which is exploited in [20]. In this method, hybrid edges combine pseudo hue and luminance information of the upper, fmiddle and lower section of the lip.

In image segmentation approach, many algorithms have been proposed. For color image segmentation, histogram-based and cluster-based methods have been widely used. In [21], [22] a histogram segmentation technique was proposed which involves performing a fuzzy partition on a two-dimensional histogram based on the maximum fuzzy entropy principle. In [23], the color lip region is segmented using a fuzzy thresholding algorithm with connectivity processing. In [24], a hue filter is used to weight the red hue, which is presumed to be the lip region. Then thresholding is used to segment the lip region. The main idea of lip detection method is based on characteristics of lip in YCbCr color space which demonstrates that lib region have high Cr and low Cb values [25]. In order to detect the lip, the following formula should be computed on every pixel of the image to separate the lip region from non-lip region. The formula is as follows:

$$LipMap = Cr^2\left(Cr^2 - \eta\frac{Cr}{Cb}\right)^2 \qquad (3)$$

$$\eta = .95\frac{\frac{1}{m}\sum Cr(x,y)^2}{\frac{1}{m}\sum Cr(x,y)/Cb(x,y)} \qquad (4)$$

Where $(Cr)^2$,(Cr/Cb) all are normalized to the range [0 1]. This formula is designed to brighten pixels with high Cr and low Cb values. $(Cr)^2$ emphasizes pixels with higher Cr value and also (C_r/C_b) component completes our idea that lip regions have high Cr and low Cb values.

Our simulation results show that this formula does not work properly for various kinds of images including the color of skin. Moreover it is not flexible under various lightening conditions. We need an optimization algorithm that clusters the pixels of lip from non-lips' pixels. In figure 1 some of the lips detected with this formula are demonstrated.

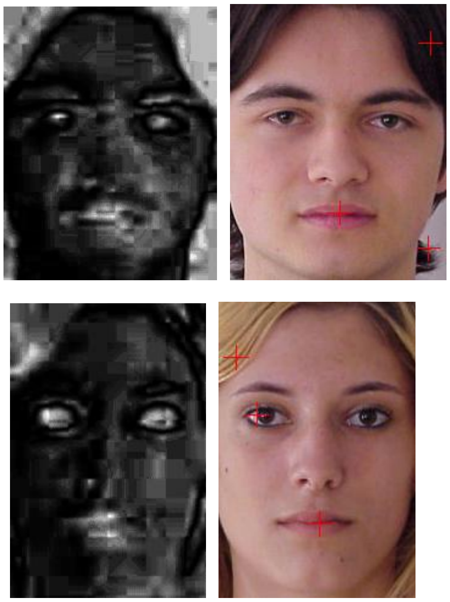


Figure 1. Sample of Lip Detection with formula 3,4.

4.1 PSO Tuning Approach

As discussed earlier, we need an approach that can cluster lip from other face region efficiently. We have proposed a novel approach based on PSO and YCbCr color characteristic of lip. Our goal in the proposed algorithm is to obtain the values of A and B in the following formula. The initial values of A and B are random numbers [0 1]. This should be accomplished in a way to separate the lip's pixels from non-lip ones:

$$PSO-LMap = A*Cr^2\left(A*Cr^2 - B*\frac{Cr}{Cb}\right)^2 \quad (5)$$

In order to achieve the optimal values of A and B, two sets which include the pixels of lip and pixels of skin are created. These two sets are used to tune the A and B values and are concatenated to generate a new set *train*. For every pixels of set *train* formula 5 is calculated to obtain the *lip-map* set. A clustering algorithm such as Fuzzy C-means (FCM) is used to cluster the *lip-map* set. The number of pixels which are clustered as a wrong category is computed. This number is the *fitness* function which should be minimized. Particles fly in two dimensions A and B until the optimized point which is the minimized value of *fitness* function, is reached.

In the following figure 2 and 3, values of lip's pixels and face's pixels are demonstrated before and after PSO tuning. As indicated in the figures, the first image is a high density composition of two set of pixels, however in the second one the pixels are separated properly.

4.2 Experimental Results

This section provides simulation results to evaluate our method with and without PSO tuning. We apply our algorithm on CVL [26] and Iranian databases. Summary of the detection results (including the number of images, detection rates) on the CVL and Iranian databases are presented in Table. 3 and Table. 4, respectively. The detection rate is computed by the ratio of the number of correct detection to that of all the images tested. Sample of detections on CVL and Iranian databases are demonstrated in figure 4 and 5.

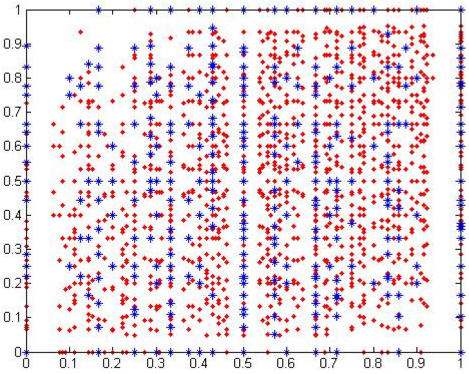


Figure 2. lips and non-lips pixel before PSO tuning.

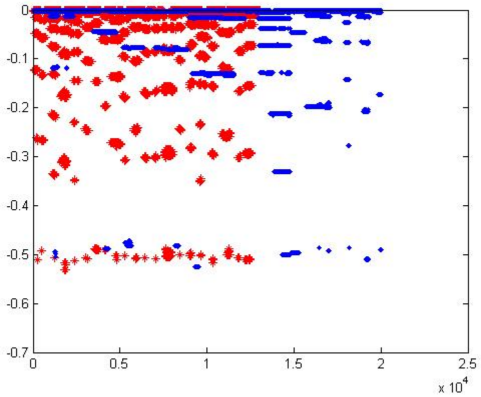


Figure 3. Lips and non-lips pixel after PSO tuning

CVL Database: CVL database consists of head and shoulder images taken from 114 people in 7 kinds of expressions. Among 7 images taken from a person, 3 of them are suitable for our purpose. These three photos are frontal view and with different expressions: serious, smile and grin.

TABLE I. RESULTS ON CVL DATABASE WITHOUT PSO

Expression	Serious	Smile	Grin	Total
No. of image	110	110	110	330
Data Rate (%)	84.54	82.72	80.90	82.72

TABLE II. RESULTS ON THE IRANIAN DATABASE WITHOUT PSO

Gender	Female	Male	Total
No. of image	28	22	50
Data Rate (%)	78.56	81.18	80

TABLE III. RESULTS ON CVL DATABASE WITH PSO

Expression	Serious	Smile	Grin	Total
No. of image	110	110	110	330

Data Rate (%)	92.72	90.99	89.09	91.22

TABLE IV. RESULTS ON THE IRANIAN DATABASE WITH PSO

Gender	Female	Male	Total
No. of image	28	22	50
Data Rate (%)	92.85	91	92

Iranian Database: Iranian database consists of head and shoulder images taken from 50 people. Images in Iranian database are taken under various lighting conditions.

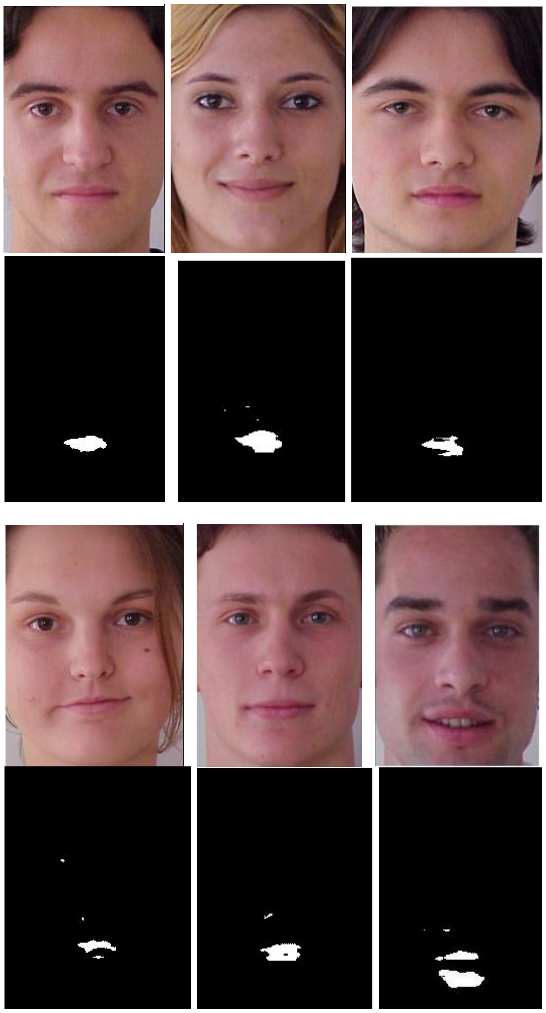
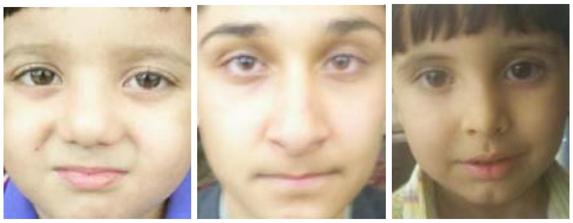


Figure 4. Sample of CVL Detection



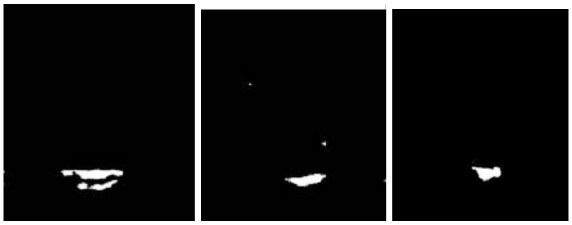


Figure 5. Sample of Iranian Detection

5 Computational Complexity Analysis

Let *n* be number of particles, the serial PSO model needs *O(n)*-time algorithm for computing values by SAs and finding maximum amount. Therefore in *i* iterations to be completed the program runs in *O(i.n)*. On the other hand, in our multi-agent distributed system, sort of parallel computing is done. For simplicity, consider we have an equal amount of *n/m* SAs in each of the *m* sites. Since the serial operation mentioned above must also be done for each separate site, we will have a *O(n/m)* complexity for SAs behavior, *O(n/m)* for LOAs and a maximum computation in *O(m)* for the GOA. Consequently due to parallel characteristic of our architecture, we would finally have a *O(i.(m+n/m))*-time algorithm.

Suppose $T_1(n,m)$ defines the running time of the serial algorithm and $T_2(n,m)$ for the parallel one. We then have:

$$T_1(n,m) \in O(n.i) \quad , \quad T_2(n,m) \in O(i.(m+n/m))$$

$$\frac{T_1(n.m)}{T_2(n.m)} \rightarrow \frac{O(i.(m+n/m))}{O(i.n)} \approx \frac{m^2+n}{n.m}$$

Regarding this outcome we can always define appropriate number of sites (*m*) to reach a faster convergence for an optimization problem.

6 Conclusion

A simple yet effective multi-agent particle swarm optimization method was proposed to overcome curse of dimensionality for large scale problems. Lip detection in color images, as a high-dimensional application, has been investigated and a novel approach for obtaining an optimized lip-map was proposed. Experimental results show 92% correction rate which is 11% increase in comparison to the simple approach. A computational complexity analysis also shows the superiority of the proposed architecture to be used in other large scale application.

7 References

[1] Liu, Y., Yao, X., Zhao, Q., and Higuchi, T., “Scaling Up Fast Evolutionary Porgramming with Cooperative Coevolution,” Proc. of the Congress on Evolutionary Computation, pp. 1101–1108, 2001

[2] Potter, A. M. and De Jong, K. A., “A cooperative co-evolutionary approach to function optimization,” Proc. of the Third International Conference on Parallel Problem Solving from Nature, pp. 249–257, Springer-Verlag, 1994.

[3] Sofge, D., De Jong, K. A., and Schultz, A., “A blended population approach to cooperative coevolution for decomposition of complex problems,” Proc. of the Congress on Evolutionary Computation, pp. 413–418, 2002.

[4] Bergh, F.v.d., Engelbrecht, A. P.,“A Cooperative Approach to Particle Swarm Optimization,” IEEE Transactions On Evolutionary Computation, vol. 3 pp. 225–239, 2004.

[5] D.A. Van Veldhuizen, J.B.Zydallis, G.B.Lamont, “Considerations in engineering parallel multiobjective evolutionary algorithms”, Evolutionary Computation, IEEE Transactions on , vol.7, no.2, pp. 144-173, April 2003.

[6] A. Tettamanzi and M. Tomassini, Soft Computing: Integrating Evolutionary, Neural and Fuzzy Systems. New York: Springer, 2001.

[7] C. A. Coello Coello, D. A. Van Veldhuizen, and G. B. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems. New York: Kluwer, May 2002.

[8] Kennedy, J., Eberhart, R. C., "Particle swarm optimization " Proceedings of IEEE International Conference Piscataway, NJ, pp. 1942-1948, 1995.

[9] I. Foster, C. Kesselman (eds.) The Grid: Blueprint for a Future Computing Inf., Morgan Kaufmann Publishers, pp. 105-129, 1999.

[10] Bellifemine, F., G. Caire, T. Trucco, G. Rimassa, “JADE Programmer’s Guide”, 21 August, 2006.

[11] JADE Board, “JADE WSIG Add-On Guide JADE Web Services Integration Gateway (WSIG) Guide”, 03-March-2005

[12] E. Cortese, F. Quarta, G. Vitaglione, P. Vrba. “Scalability and Performance of the JADE Message Transport System”. Analysis of Suitability for Holonic Manufacturing Systems, exp, 2002.

[13] S. Liu, P. Küngas, M. Matskin, “Agent-Based Web Service Composition with JADE and JXTA”, Proceedings of the 2006 International Conference on Semantic Web and Web Services, SWWS 2006, Las Vegas, Nevada, USA, June 26-29, 2006

[14] J. D. Gradecki, “Mastering JXTA: Building Java Peer-to-Peer Applications”, JohnWiley&Sons,2002.

[15] Guoyun Lv, Rongchun Zhao, Dongmei Jiang, Yan Li, Sahli, H., "A Robust Visual Feature Extraction based BTSM-LDA for Audio-Visual Speech Recognition", Communications and Networking in China, 2007. CHINACOM '07. Second International Conference on 22-24 Aug. 2007 Page(s):637 - 641

[16] Saitoh, T., Konishi, R., "Lip Reading using Video and Thermal Images",SICE-ICASE, 2006. International Joint Conference,Oct. 2006 Page(s):5011 – 5015

[17] Gomez, E., Travieso, C.M., Briceno, J.C., Ferrer, M.A. "Biometric identification system by lip shape". Proceedings of the 36th Annual International Carnahan Conference on Security Technology. 2002. pp. 39 - 42

[18] Chang, T.C., Huang, T.S., Novak, C. "Facial feature extraction from color images". Proceedings of the 12th IAPR International. Conference on Pattern Recognition, Computer Vision & Image Processing. 1994. Vol. 2. pp. 39 - 43

[19] Sadeghi, M. Kittler, J. Messer, K. "Modelling and segmentation of lip area in face images" Proceedings of IEE Conference on Vision, Image and Signal Processing. 2002. Vol. 149, pp. 179 - 184

[20] Eveno, N., Caplier, A., Coulon, P.-Y. "A parametric model for realistic lip segmentation". Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision, 2002. vol.3 pp. 1426 – 1431

[21] H. D. Cheng, Y. H. Chen, and X. H. Jiang, “Thresholding using twodimensional histogram and fuzzy entropy principle,” IEEE Trans. Image Processing, vol. 9, pp. 732–735, Apr. 2000.

[22] H. D. Cheng, J. R. Chen, and J. Li, “Threshold selection based on fuzzy C-partition entropy approach,” Pattern Recogn., vol. 31, no. 7, pp. 857–870, 1998.

[23] S. Lucey, S. Sridharan, and V. Chandran, “Chromatic lip tracking usinga connectivity based fuzzy thresholding technique,” presented at the 5th Int. Symp. Signal Processing Applications ISSPA’99, Brisbane, Australia, Aug. 22–25, 1999.

[24] T. Coianiz, L. Torresani, and B. Caprile, “2D deformable models for visual speech analysis,” in Speechreading by Humans and Machines, D. G. Stork and M. E. Hennecke, Eds. New York: Springer-Verlag, 1996

[25] Hsu Rein-Lien, A. M. Mohamed, and K. J. Anil. Face detection in color images. IEEE Trans. Pattern Analysis and Mechine Intelligence, 24(5), 2002.

[26] Franc Solina, Peter Peer, Samo Juvan, and Jure Kovac. Color-based face detection in the ”15 seconds of fame” art installation. In Mirage 2003, Conference on Computer Vision / Computer Graphics Collaboration for Model-based Imaging, Rendering, image Analysis and Graphical special Effects, pages 38–47, INRIA Rocquencourt, France, March 2003.