International Journal of Innovative Research in Electronics and Communications (IJIREC)
Volume 1, Issue 1, April 2014, pp.27-32
www.arcjournal.org

MPSO- PCA Based Facial Recognition

Sandeep
sandeeppundri@gmail.com

Pushparaj Pa
pushprajpal@gmail.com

Sachin Gupta
sachinguptasrm@gmail.com

Abstract: *In this paper an algorithm to solve the problem of automatic face recognition is presented. The novelty of the algorithm is the ability to combine the computer vision tasks with Particle Swarm Optimization (PSO) to improve the execution time and to obtain better recognition results. The efficiency of face recognition system is improved by using a fitness function to measure the similarity of an input face compared with a database of faces. The use of the fitness function helps to obtain more accurate results in a faster way. The results obtained are excellent even when the system was tested in uncontrolled environments. A comparison of the results obtained with the algorithm without PSO versus the algorithm using PSO is also presented. The algorithm is also implemented on the colored images of the human faces.*

Keywords: *Particle Swarm Optimization, Face Recognition, Eigen faces, Evolutionary Computer Vision.*

1. Introduction

Among the different fields in which Evolutionary Computation (EC) techniques can be used as an aid to design solutions, computer vision (CV) is certainly one of the most popular [1]. Evolutionary computer vision is a new research area whose main goal is to study computer vision using evolutionary and genetic computational approaches.

Genetic and evolutionary algorithms can be applied to those fields whose task requires an optimum performance in many possible scenarios that characterize real-world problems.

Computer vision and image understanding represents one of the most challenging tasks because of the complexity involved in order to provide computers with human-like perception capabilities, allowing them to sense the environment, understand the sensed data, identify patterns, take appropriate actions and learn from experience to enhance future performance [2].

One of the most active research areas in the study of computer vision field is face recognition [3]. Face recognition is a high level visual problem used to help fight crime and terrorism, for access control to physical spaces, for interpreting human actions, for ubiquitous and pervasive computing, etc. The typical algorithms used for face recognition are:

Kernel Class-Dependent Feature Analysis (KCFA) [4],Tensorfaces [5], manifold learning methods [6], kernel methods [7], Neural Networks (NN) [8], Support Vector Machines (SVM) [9], Local Features Analysis (LFA) [10], Independent Component Analysis (ICA) [11], Linear Discriminant Analysis (LDA) known as fisher faces [12], and Principal Components Analysis (PCA) known as eigenface [13].

This paper focuses in the solution of a typical face recognition problem using digital images by means of Particle Swarm Optimization (PSO). The novelty of the algorithm is the possibility to combine the computer vision tasks with Particle Swarm Optimization (PSO) in order to improve the speed and the performance of the recognition stage and to provide better solutions.

The paper is organized as follows: the related works are presented in section II. The face recognition algorithm using PSO is presented in section III. The tests and the analysis of the results obtained are shown in section IV. Finally, conclusions and suggestions for further works are given in section V.

2. Related Works

The use of evolutionary algorithms in computer vision is a new research area and there exists only a few works in the literature, some of them are described following. In [14] a solution for a typical problem of quality control and inspection of work pieces is presented. The main goal is to measure the roundness of a circular work piece. Using PSO different features were computed such as Maximum Inscribed Circle (MIC), Maximum Circumscribing Circle (MCC), Minimum Zone Circle (MZC) and Least Square Circle (LSC). The results obtained reveal that PSO based method effectively solve the problem of compute MIC, MCC and MZC.

An algorithm to solve the problem of polygonal approximation is presented in [15]. The information of the object boundary is obtained using PSO. The boundary is divided into a finite number of segments; each segment is connected using a line and as a result a reduction of the storage size (compression) of the original image is obtained.

A method that uses PSO and fuzzy logic is presented in [16]; the combination is used to solve a problem of image segmentation. The work presented at [1] shows several applications of evolutionary computer vision such as: evolutionary design of digital filters, signal detection in electrocardiography (ECG), object detection and tracking algorithms, on-line learning, etc. The paper shows a review of the opportunities offered by the evolutionary computation for researchers in the computer vision field.

A face recognition system using PSO is presented at [17]. PSO is used for the crucial stage of feature selection in order to reduce the number of features obtained for the recognition stage. The algorithm is applied to coefficients extracted by two feature extraction techniques: the discrete cosine transforms (DCT) and the discrete wavelet transform (DWT).

The proposed PSO-based feature selection algorithm is utilized to search the feature space for the optimal feature subset where features are carefully selected according to a well defined discrimination criterion.

Finally, in [18] a genetically inspired learning method for Facial Expression Recognition (FER) is presented. The method can discover the features automatically in a genetic programming-based approach that uses Gabor wavelet representation. The features are used to train a support vector machine classifier that is used for recognizing the facial expressions.

3. Face Recognition using PSO

The problem of face detection and recognition is a typical task of computer vision. The goal is to compare an image with a data base of images and to determine which image is the most similar. In this section a novel solution to this problem is presented. The solution proposes the combination of computer vision with PSO.

3.1. Particle Swarm Optimization Algorithm

Particle Swarm Optimization is an evolutionary computational paradigm created by Kennedy and Eberhart in 1995 [19]. The original idea was to simulate the movement of organisms in a bird flock, PSO is a population based optimization algorithm, where each particle is an individual, and the swarm is composed by particles. The problem solution is formulated as a search space and each position at that space is a correlated solution of the problem [20].

In order to use PSO as a mechanism of optimization several definitions were made. Initially, a population of particles inside a multidimensional space with random speeds and positions are determined for the particular problem. Each particle is evaluated using a fitness function. The value of the fitness function is compared with the local best (*lbest*) of each particle, when the value obtained is better than the lbest this is changed to the current position of a multidimensional space. Then, the value is compared with the best value obtained from the population named global best (*g best*). Immediately, the speed of each particle is modified, the algorithm is repeated until the stop criterion is completed.

The particles are the main part of the PSO algorithm. The particles are defined by three vectors. The vector x defines the current position of a particle at a time t. The vector l best stores the best solution obtained. The vector v stores the speed and the direction followed by each particle inside the search space.

Once the particles are defined inside the search space, the movement of each particle can be expressed mathematically.

xt(t+1) = xt (t) + vt(t+1) (1)

- ***xi (t+1)***=[*x*in,*x*i2,…,*x*in]^T contains the position of each particle.
- ***vi(t+1)***=[vin,vi2,…,vin]^T contains the speed of each particle.
- ***lbesti(t+1)***=[pin,pi2,…,pin]^T represents the best value of the fitness function for each particle.
- ***w*** is the inertia weight.
- ***c1, c2*** learning weights controlling the cognitive and social components.
- ***rnd( )*** are random numbers generated on [0,1].

With all these considerations, an optimization problem can be solved using PSO. The main challenge is to define an efficient fitness function to quantify the optimality of a solution and to determine how well the program is able to solve the face recognition problem.

3.2. Face Recognition

The ability to find images inside a database without the necessity of one to one comparisons, offers a better performance in the processing time and less computer complexity.

To explain the solution of a typical problem of face recognition (without PSO) we used the algorithm mainly based on the calculation of eigenface proposed by [13]. The algorithm is dived in the training phase and the recognition phase. In the training phase, the first step is to obtain the initial set of face images called the training set. In the second step, an image is treated as a vector and the eigenvectors and eigenvalues of the matrix are computed. Fig. 1 shows an example of 400 face input images for the training phase represented as vectors.

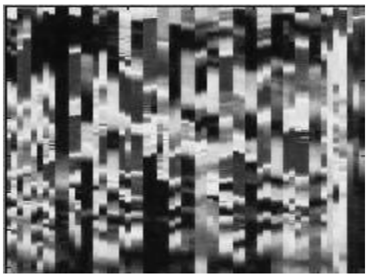

Figure 1. *The matrix containing 40 test images, each image is represented by a vector in each column.*

The matrix containing 40 test images, each image is represented by a vector in each column. The eigenvectors are ordered and only the M images with higher associated eigenvalues are stored as shown in Fig. 2.

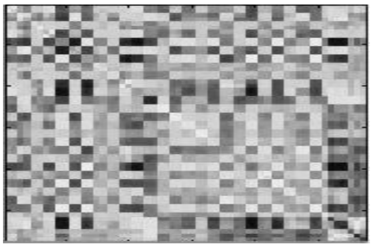

Figure 2. *Graphic representation of the eigenvalues obtained from the covariance matrix of Fig. 1.*

The M images represent the face space. The eigenvectors are used as a set of features to characterize the variations between images; an individual eigenvector is called an eigenface. An image can be approximated using only the best eigenface that are selected using a threshold. Figure 2. Graphic representation of the eigenvalues obtained from the covariance matrix of Fig. 1.

In the last step, a distribution of M-dimensional weight space is calculated by projecting the face image onto the face space. For the recognition phase, it is determined whether the image is a face or not by comparing the similarity with the face space. If the image is a face, the set of weights and the M eigenface of the input image are calculated by projecting the input image onto each of the eigenface. Then, the Euclidean distance between two images is calculated using the formulae. After that, the index of the recognized image is obtained and classified as a known or unknown person. The percentage of effectiveness for recognition using this approach is about 70% of 40 images tested, and the process to recognize a face is very slow.

3.3. Evolutionary Face Recognition

It is necessary to enhance and to speed up the results obtained in previous section this is made by adding a stage of PSO. In order to include the PSO different parameters need to be defined. In [2] is mentioned that the use of evolutionary algorithms in computer vision requires knowledge of the application domain and abstraction of the problem domain in terms of evolvable structures through the selection of the appropriate representation.

First, we need to determine if the problem can be solved using and evolutionary approach. To do this, the problem of face recognition is mapped into the PSO environment. We design a PSO algorithm with the restriction of finding only the best M approximations in N iterations both selected by the system user. The values of the cognitive (*c1*) and social (*c2*) parameters were set to two for faster convergence of the algorithm. The inertia weight (*w*) is set to 0.8; this is used for the convergence behavior and to regulate the tradeoff between global and local exploration abilities of the swarm. The parameter (*rnd*) is used to maintain the diversity of the population and is uniformly distributed between zero and one. Required, the image is converted to the grayscale plane. An image of the training set is selected randomly and changed to the grayscale plane. Then, the image is normalized by subtracting the mean value from all the pixels values contained in the image; this new matrix is called the matrix *A*.

The matrix *L* is obtained by multiplying *A* by its transposed, immediately the eigenvalues (*V*) and eigenvectors (*D*) of *L* are computed. *V* is the modal matrix and *D* is a canonical diagonal matrix. We select only the eigenvalues above a threshold of one and its associated eigenvectors. The selected eigenvectors are multiplied by *A* to obtain the eigenface matrix (*E*) of the selected image.

A projection is obtained by multiplying the transpose of the eigenface matrix by the vector representing the selected image. Then, a matrix of differences (*DI*) is obtained by subtracting the mean value of the image to the input image. The projection of the input image is obtained by multiplying the transpose of the eigenface matrix (*E*) by the matrix of differences (*DI*). The PSO algorithm begins with a random change in the position of the initial particles; the velocities of each particle are calculated and associated randomly. The first iteration of PSO starts by using (1) to obtain the new velocity of a particle. Then, the best local position (*lbest*) of the particle is obtained and the fitness function for the current particle is tested.

The fitness function is the sum of Euclidean distance and Mahalanobias distance between the two images compared.

Finally, the global best (*gbest*) position is searched by determining if the *lbest* is better than *gbest* and then the change of positions is made. The algorithm performs as if the particles are moving inside the search space and comparing the different solutions with the fitness function until the maximum number of iterations is reached.

4. Tests And Results

The test was carried out using a training set of 400 images with 10 samples for each subject from two databases prepared from ORL database and Essex Database.

The ORl database images have resolution of 92 x 112 and Essex database (colored images) have the resolution of 180x200. The set consists of faces with images of men and women of different ages. The test was conducted using test database having 40 images of different subjects from both the databases. The similarity measure between the test images and training images is done using fitness function which is optimized.

The fitness function is the sum of Euclidean distance and Mahalanobias distance between the two images compared.

Table1. *Recognition rate and error rate for Test databases*

S. No	Algorithm	Recognition rate (%)	Error rate(%)
1.	Facial recognition based on PCA	92.5%	7.5%
2.	PSO optimized facial recognition based on PCA (for ORL Database images)	97.5%	2.5%
3.	MPSO-PCA based facial recognition (for colored images from Essex database.)	97.5%	2.5 %

Figure3. *Test image and equivalent image for ORL Database*

Figure4. *Test image and the equivalent image from the Essex database gallery*

5. Conclusions

In this paper a novel algorithm for evolutionary face expression recognition using PSO to improve the execution time and to obtain better recognition results was presented.

A novel algorithm for evolutionary face expression recognition using PCA-PSO to improve the computation time and to obtain better recognition results is presented. PSO is a very helpful tool for problems where the search space is large. As the results show, the efficiency of the PCA-PSO algorithm is better than the existing PCA based Facial recognition algorithms. The recognition rate for original PCA is 92.5 % for the test databases while for PCA-PSO is 97.5 %. So recognition rate has improved by 5 %. Since PSO-PCA algorithm for FR performed better with improved recognition rate it could be used for better accuracy in facial image recognition. Also it could perform well for facial recognition of colored images.

REFERENCES

[1] S. Cagnoni, “Evolutionary Computer Vision: A Taxonomic Tutorial”, Proceedings of the 8th International Conference on Hybrid Intelligent Systems (HIS), pp. 1-6, Barcelona, Spain, September 2008.
[2] G. Olague, S. Cagnoni and E. Lutton, “Introduction to the Special Issue on Evolutionary Computer Vision and Image Understanding”, Pattern Recognition Letters, vol. 27, no. 11, pp. 1161-1163, August 2006.
[3] [3] A. K. Jain, P. Flynn and A. Ross, Handbook of Biometrics, First edition, New York, USA, Springer + Business Media, 2008.
[4] C. Xie, M. Savvides and B. Vijaya Kumar, “Kernel Correlation Filter Based Redundant Class-Dependence Feature Analysis (KCFA) on FRGC2.0 Data”, IEEE Workshop on Analysis and Modeling of Faces and Gestures (AMFG), pp. 32–43, November 2005.
[5] M. A. O. Vasilescu and D. Terzopoulos, “Multilinear Analysis of Image Ensembles: TensorFaces”, Proceedings of the European Conference on Computer Vision (ECCV), pp. 447–460, May 2002.
[6] S. Roweis and L. K. Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding”, Science, vol. 290, no. 5500, pp. 2323– 2326, December 2000.
[7] J. S. Taylor and N. Cristianni, Kernel Methods for Pattern Analysis, Cambridge University Press, 2004.
[8] S. Nazeer, N. Omar and M. Khalid, Face Recognition System using Artificial Neural Networks Approach, 1st ed. Chennai, India, IEEE MIT Campus, Anna University, ICSCN, 2007.
[9] B. Heisele, P. Ho and T. Poggio, “Face Recognition with Support Vector Machines: Global versus Component-based Approach”, Proceedings of the IEEE International Conference on Computer Vision (ICCV), vol. 2, pp. 688–694, July 2001.
[10] P. S Penev. and J. J. Atick, “Local Feature Analysis: A General Statistical Theory for Object Representation”, Network: Computation in Neural Systems, vol. 7, no. 3, pp. 477–500, 1996.
[11] M. Bartlett, J. Movellan and T. Sejnowski, “Face Recognition by Independent Component Analysis”, IEEE Transactions on Neural Networks, vol. 13, no. 6, pp. 1450– 1464, November 2002.
[12] H. Yu and J. Yang, “A direct LDA Algorithm for High-Dimensional Data with Application to Face Recognition”, Pattern Recognition, vol. 34, no. 10, pp. 2067–2070, October 2001.
[13] M. Turk and A. Pentland, “Eigenfaces for Recognition”, Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71–86, March 1991.
[14] S. Te-Hsiu, “Applying Particle Swarm Optimization Algorithm to Roundness Measurement”, Expert Systems with Applications, vol. 36, no. 2, pp. 3428–3438, March 2009.
[15] J. Wang, Z. Kuang, X. Xu and Y. Zhou, “Discrete Particle Swarm Optimization Based on Estimation of Distribution for Polygonal Approximation Problems”, Expert Systems with Applications, vol. 36, no. 5, pp. 9398-9408, July 2009.
[16] S. Das and A. Konar, “Automatic Image Pixel Clustering with an Improved Differential Evolution”, Applied Soft Computing, vol. 9, no.1, pp. 226-236, January 2009.
[17] R. Ramadan and R. Abdel-Kader, “Face Recognition Using Particle Swarm Optimization-Based Selected Features”, International Journal of Signal Processing, Image Processing and Pattern Recognition, vol. 2, no. 2, pp. 51-66, June 2009.
[18] J. Yu and B. Bhanu, “Evolutionary Feature Synthesis for Facial Expression Recognition”, Pattern Recognition Letters, vol. 27, no. 11, pp. 1289-1298, August 2006.
[19] J. Kennedy and R. Eberhart, “Particle Swarm Optimization”, Proceedings of the IEEE International Conference on Neural Networks, vol. 6, pp. 1942–1948, Piscataway, NJ, December 1995.
[20] D. Y. Sha and C. Y. Hsu, “A New Particle Swarm Optimization for the Open Shop Scheduling Problem”, Computers an Operations Research, vol. 35, no. 10, pp. 3243-3261, October 2008.