

Fast Automatic Face Recognition from Single Image per Person Using GAW-KNN

Hasan Farsi*

Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran
hfarsi@birjand.ac.ir

Mohammad Hasheminejad

Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran
mhashemi@birjand.ac.ir

Received: 27/Aug/2013

Revised: 22/Feb/2014

Accepted: 07/June/2014

Abstract

Real time face recognition systems have several limitations such as collecting features. One training sample per target means less feature extraction techniques are available to use. To obtain an acceptable accuracy, most of face recognition algorithms need more than one training sample per target. In these applications, accuracy of recognition dramatically reduces for the case of one training sample per target face image because of head rotation and variation in illumination state. In this paper, a new hybrid face recognition method by using single image per person is proposed, which is robust against illumination variations. To achieve robustness against head variations, a rotation detection and compensation stage is added. This method is called Weighted Graphs and PCA (WGPCA). It uses harmony of face components to extract and normalize features, and genetic algorithm with a training set is used to learn the most useful features and real-valued weights associated to individual attributes in the features. The k-nearest neighbor algorithm is applied to classify new faces based on their weighted features from the templates of the training set. Each template contains the corrected distances (Graphs) of different points on the face components and the results of Principal Component Analysis (PCA) applied to the output of face detection rectangle. The proposed hybrid algorithm is trained using MATLAB software to determine best features and their associated weights and is then implemented by using delphi XE2 programming environment to recognize faces in real time. The main advantage of this algorithm is the capability of recognizing the face by only one picture in real time. The obtained results of the proposed technique on FERET database show that the accuracy and effectiveness of the proposed algorithm.

Keywords: EBGM; Face Recognition; PCA; Weighted Feature; WGPCA.

1. Introduction

There has been a great deal of progress in face detection methods. This field has been concerned researchers of several disciplines such as image processing, pattern recognition, computer vision, neural network and computer graphic [1] because of its high accuracy and low intrusiveness [2]. The performance of the most face recognition algorithms can be seriously affected by the limited number of training sample per person [2].

There are two major steps in face recognition algorithms. First, it is necessary to know whether any human face is available on the image. This is called face detection step. In the second step, by means of extracted features from the image, the aforementioned face should be recognized which is called face classification step.

In the face detection step, there are number of algorithms that can detect frontal face position in an image. Learning based approaches [3] estimate the complex non-convex face and non-face from training images. In these methods, the problem is to select sufficient pattern to adequately characterize non-face space. Some other methods [4] use multiple features such

as color, shape and texture to segment faces from the background. Although these methods are fast, they suffer from complex backgrounds. One of the fastest and most reliable algorithms used in this step is HAAR-like algorithm which is used to reliably detect objects in real time. HAAR-like features were introduced by Viola et al [5] and are successfully used in much object detection and face detection methods [6]. The goal of face detection is to specify existence of face in an image that is a frame obtained from a surveillance camera or an individual stored image and if exists, accurately determination of face position in the image.

Face recognition refers to an automatic method that identifies an individual face image and pre-trained templates on a database. Face recognition techniques can be divided into three main categories. The earliest methods are local feature-based matching methods [7, 8]. These approaches are based on detection of individual facial features such as eyes and nose, and comparing to the corresponding features stored in the database. Elastic Bunch Graph Matching (EBGM) and its variations are included in this category. In EBGM local information extracted with Gabor filters is used for discrimination. Second category is holistic feature-based matching

* Corresponding Author

methods [9,10]. These methods are centered on using the whole face region as raw input for face recognition system. Principal Component Analysis (PCA), Linear Discriminate Analysis (LDA), Independent Components Analysis (ICA), Fisher's Linear Discriminate (FLD) and their variations are examples of this category. The last one is hybrid matching method that uses both local and holistic features [11].

After extracting facial features from input face image the image is classified as one of predefined target faces. In this stage classification is performed using neural network, statistical methods or any other pattern recognition methods. It has been shown that hybridizing genetic algorithm (GA) and a K-Nearest Neighbor (KNN) classification, as a statistical method, can be used in classification [12]. This method improves the performance of the k-nearest neighbor algorithm and can be used in optimization problems. In the presented method 82 points are extracted from face using Active Shape Model (ASM) and some feature vectors obtained from PCA. Since there is 3321 distances between 82 points, these distances and PCA vectors are stored as template for targets. GA is used to determine weights related to each feature.

It is not a simple matter to find an optimal vector of attribute weightings. In this paper we show how to use a GA-WKNN and human face component harmony in face recognition.

This paper is organized as follows: In section 2 the k-nearest neighbors with genetic algorithm is presented. In section 3 the proposed full automated hybrid face recognizer is introduced in details. Experimental results and a comparison of existing and the proposed methods are given in section 4 and a conclusion is drawn in the last section.

2. The K-nearest Neighbours Classification and Genetic Algorithm

The Nearest Neighbor (minimum distance) classification algorithm (NN) is based on the idea that, given a data set of classified examples, an unclassified individual should belong to the same class as its nearest neighbor in the data set [13]. The measurement of proximity, or similarity, between two individuals is given by equation (1).

$$d_{ij} = \left\{ \sum_{a=1}^n (x_{ia} - x_{ja})^2 \right\}^{\frac{1}{2}} \quad (1)$$

Where x_{ia} , is value of the a th attribute for the i th datum. In K-NN, a new instance is classified by looking at its nearest neighbors. Then the unclassified individual is assigned to the class of the majority of its k closest neighbors. This algorithm is simple, quick, and effective. It corresponds to incorrect results, however, in the case of face recognition because portion of features that are significant in the classification process is small with respect to the number of less important features. Given a vector of features, the KNN treats each feature as equal in

the classification process. So, it is necessary to use Weighted KNN (WKNN) instead.

Assigning proper weights to the extracted features modifies the importance of features to reflect its relevance in recognition. Muscular structure of human face results in variations in face components positions and shape also illumination conditions, position of face according to the camera and many other factors makes the aforementioned variety of importance.

Genetic algorithm is a heuristic algorithm used in optimization and machine learning inspired from processes of biological evaluation. John Holland created the genetic algorithm field [14]. GA can efficiently solve the large parameter optimization problems and is used to find an optimal solution for the problems. Since genetic algorithm do not rely on problem specific knowledge, it can be used to find solutions that is difficult to find by classic mathematics. Chromosomes represent solutions within the genetic algorithm. Chromosomes are grouped into population (set of solutions) on which the genetic algorithm operates. In each step (generation), the genetic algorithm selects chromosomes from a population (selection is usually based on the fitness value of the chromosome) and combines them to produce new chromosomes. These offspring chromosomes form a new population (or replace some of the chromosomes in the existing population) in the hope that the new population will be better than the previous ones. Populations keep track of the worst and the best chromosomes, and stores additional statistical information which can be used by the genetic algorithm to determine the stop criteria. In the case of determining face features weighting, we have about 4000 face features that should be evaluated. Because weights are real values and there is large number of features, the search area for optimal values is infinite and it is impossible to find optimal values by classic mathematics. The only option is use of heuristics algorithms, and we use GA.

3. Proposed Algorithm

As mentioned all face recognition algorithms consist of two different steps, face detection and face classification. In face classification step, first some useful features should be extracted from the image by an accurate image analyzing technique and then image should be classified according to extracted features from the primary image.

3.1 Face detection

There are number of algorithms that can detect frontal face position in an image. In this section HAAR-like feature is used in Adaboost algorithm to reliably detect faces in real time. This approach is used because of its speed and accuracy. This technique is based on the idea of wavelet template that defines shape of a template in terms of wavelet coefficients of the image. With a scaling factor of 1.1, as recommended in [15], the input image is scanned

across location and scale. According to AdaBoost algorithm [3] a set of weak binary classifiers is learned by a training set. Each classifier is a function composed of a rectangular sum followed by a threshold. The features consist of boxes of different sizes and locations.

3.2 Local and holistic feature extraction

In this article, the recognition system has only one picture of the target face that it should gain the maximum use of the picture to be robust to head rotation and illumination variations. Therefore, integration of global and local features is implemented to achieve maximum use of the training set.

3.2.1 Local features

In this stage, face features are extracted from the output square region in face detection block. Facial ground truth is found according to Active Shape Model (ASM) algorithm as local features. ASM is a powerful tool for shape localization problems. In ASM, a set of points are assumed to be a flexible model of image structure whose shape can vary. A set of points are aligned automatically to minimize the variance in distance between equivalent points [16]. To achieve higher speed and accuracy, the algorithm search area for each component is restricted. As shown in Fig.1, the location of 82 points is specified in input face image.

According to our work on FERET [17], ORL [18] and face94, face95, face96 and grimace databases [19], every facial component has a region and a harmony exists about place and size of these components.



Fig. 1. Of Point Extraction Results from FERET and Face94 Databases

The first component that will be found is eye. Because of kind of HAAR-like template, we use eyes which are around the height middle of the face boundary square. In worst case, pupil was about $\pm 1/10$ far from middle of the height. So, for assurance we consider it to be $\pm 1/5$. Therefore, if upper right corner of the square would be center of coordinates, with respect to the pupil and eye width, vertical search region for eyes would be $[2/5w \ 3/5w]$, where w denotes length of the extracted face boundary square. Different components should be sequentially analyzed.

Table 1: Proposed search boundary for each facial component

Graph name	X		Y	
Left eye	$\frac{1}{2}w$	$\frac{9}{10}w$	$\frac{3}{10}w$	$\frac{3}{5}w$
Right eye	$\frac{1}{10}w$	$\frac{1}{2}w$	$\frac{3}{10}w$	$\frac{3}{5}w$
Left eyebrow	$\frac{1}{2}w$	$\frac{9}{10}w$	$ule^{1-\frac{1}{5}}w$	$\text{Min}(licy^2, locy^3)$
Right eyebrow	$\frac{1}{10}w$	$\frac{9}{10}w$	$ure^{4-\frac{1}{5}}w$	$\text{Min}(ricy^5, rocy^6)$
Between eyebrows	$\frac{rbi^7 + lbi^8}{2}$			
Nose	$\frac{1}{4}w$	$\frac{1}{2}w$	bey ⁹	uply ¹⁰
Left wrinkle	leox ¹¹	leix ¹²	locy	licy
Right wrinkle	reox ¹³	reix ¹⁴	rocy	ricy
Lower jaw boundary	0	w	$-\frac{w}{2}$	oey ¹⁵
Nose tip	$\frac{w}{4}$	$\frac{3w}{4}$	bey	uply

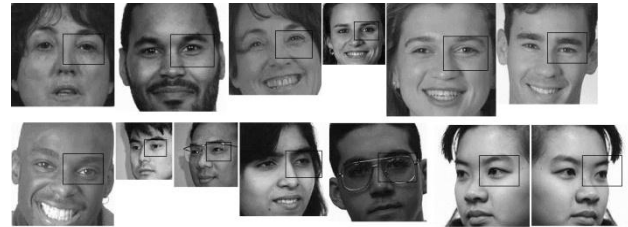


Fig. 2. Search Area for Left Eye Model inside the Result of HAAR-like Detection.

We extract the points from the following components: eyes, eyebrows, nose, under eyes wrinkles, lips, lower jaw boundary and finally individual points such as point between eyebrows (middle of inner points of eyebrows to find rotation radius), concavity at the upper part of nose, nose tip and middle of nose holes. TABLE I shows search boundary for each component. The other points are inside one of models.

As shown in Fig.2, for all tested facial images in FERET database, left eye is inside its corresponding proposed region. These images have been chosen from those of different size and rotation state.

Rotation of head around vertical axis changes distances between points, especially horizontal ones. To eliminate this effect, a distance correction stage is applied. With respect to the place of nose tip and eyes, and eyes versus mid line, degree of rotation of head around vertical axis is found and corrected.

¹ y of upper point found on left eye

² y of left eye inner corner

³ y of left eye outer corner

⁴ y of upper point found on right eye

⁵ y of right eye inner corner

⁶ y of right eye outer corner

⁷ found coordinate of right eyebrow inner point

⁸ found coordinate of left eyebrow inner point

⁹ y of point between eyebrow

¹⁰ Maximum y of lip model

¹¹ x of left eye outer corner

¹² x of left eye inner corner

¹³ x of right eye outer corner

¹⁴ x of right eye inner corner

¹⁵ minimum of outer eye point on y axis

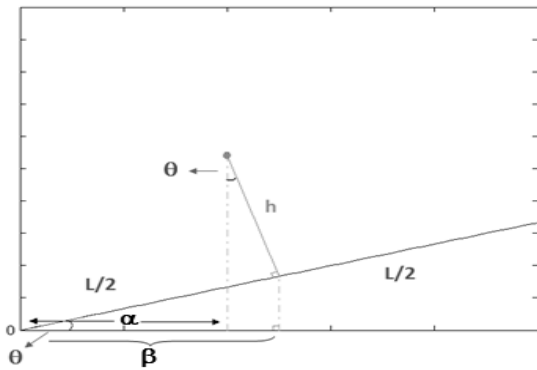


Fig. 3. Top View of Face, his distance between nose tip and a vertical plane passing center of eyes points.

In a front view of face, projection of face on the x-y plane and for a face perpendicular to camera in most cases, as shown in Fig.3, horizontal position (x) of nose tip is in the middle of left and right eye, on the x axis, but in a rotated face this is changed and x of nose tip is equal to $L \times \cos(\alpha)$ instead of $L/2$ and L is being seen as $L \times \cos(\alpha)$. Real values can be calculated from a , b and h using equation (2).

$$L = \frac{\beta}{\cos\left(\arcsin\left(\frac{\beta-\alpha}{h}\right)\right)} \quad (2)$$

For a real time automatic face recognition system, calculations can be reduced by means of a lookup table to correct important distances rapidly.

3.2.2 Holistic features

Holistic features represent a global appearance of sample. Principal Component Analysis (PCA) is a holistic feature that is widely used in face recognition. PCA is a statistical method for reducing dimension of data set while retaining the majority of the variation present in a dataset.

Recent works on face recognition have introduced linear and nonlinear dimensionality reduction techniques based on PCA. KPCA (kernel PCA), 2DPCA, B2DPCA are examples of these techniques. PCA identifies the linear combinations of variables and ignore the high order correlation value [9].

3.2.3 Specifying feature weights

Both holistic and local facial features of target and input data are compared using KNN algorithm. To improve the effectiveness of KNN approach, GA is used to select important features [20]. This decreases process time that is important in real time recognition. It has been shown that specifying proper weights increases the efficiency for every feature [12]. This stage is performed once by the researcher and there is no need for weight determination in the real time recognition application. The real time application just loads predetermined weights at startup and performs real time process. Determination of effective weights for a WKNN face recognition algorithm with more than 3000 feature vector is a hard optimization problem with an infinite search space. Since 82 points are extracted from the facial image, there is $82 \times (82-1)/2 = 3321$ distances between these points

which should be analyzed for every target. Every gene can have infinite number of real values between 0 and one. The comparison should be performed for every target and for every generation.

Solving this problem is impossible with classic mathematics. In GA-WKNN process chromosomes are vectors of real valued weights between 0 and 1.

Therefore optimization problem shown in (3) should be solved.

$$\min_w \sum_{n=1}^N w_m \cdot |x_{nm} - v_{nm}| \quad (3)$$

Subject to $w_m \geq 0$

Where w_m is weight for m^{th} feature, x_{nm} is m^{th} feature for n^{th} target and v_{nm} is m^{th} feature for an input face image that its correct target is x_n .

3.3 Face classification

Face classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs. In the training stage both features will be extracted normalized and stored in a database. In the test stage features of receiving new input is compared to the stored templates with respect to predetermined weights and classified to a category that has minimum weighted Euclidean distance from the input feature vector. There may be an input face image that does not belong to any of the stored categories.

Therefore, in order to prevent misclassification a threshold is assumed to categorize input face images that are not similar enough to the selected target, as an unknown face. Fig.4 depicts the block diagram of the face recognition system. This diagram consists of three main parts, Train, test and weight determination which is implemented once and not included in the real time face recognition.

In the training process position of target face is first determined. Found face region is then sent to perform local and global feature extraction. Normalization process is needed before PCA feature extraction and after distance calculation to reduce face image variability due to lighting and size.

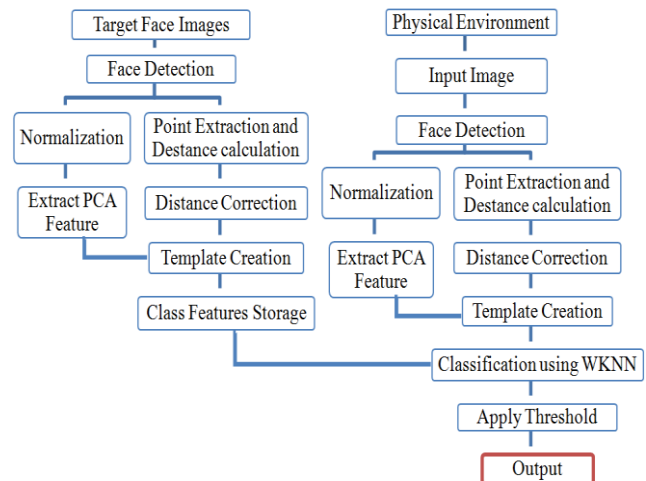


Fig. 4. The Block Diagram of the Proposed Face Recognition System

Image is rescaled to 128×128 pixels and histogram equalization is performed before PCA and distances are normalized with respect to the detected square region.

In the test stage the same process is done and resulting template is compared to the stored target templates

Weight determination process is performed once and it is not included in recognition process.

4. Simulation Results and Discussion

4.1 Weight determination

As mentioned before, weight determination is performed once, and is not included in the real time process. Since process speed is not an issue in the weight determination, in this stage we used MATLAB optimization toolbox. To gain the accuracy of optimization toolbox, we first extract templates from training faces using an application developed in Delphi programming environment. This application uses OpenCV library [15] in face detection phase. Using the extracted templates, genetic algorithm has been implemented with 200 iterations, 20 population, 20 targets and 111 training samples. It takes 138 minutes with a core i7 2.2GHz system to find optimum weights and increase recognition rate from 84.6% to 93.7% of correct classifications. For every class only one fa picture is used as a known sample. Fig. 5 shows the convergence of population to the optimum weights for this experiment. In this and next experiment the important matter is finding proper weighting and the code is not optimized in the case of speed.

In a second experiment, we use a computer sever with 16 CPU of 2.9 GHz and Matlab 2010a optimization toolbox on windows server 2008. Using the aforementioned templates, 502 samples in 197 categories were tested with 500 iterations and population of 30. It took about 26 hours to reach 94% of accuracy. The convergence has been shown in Fig. 6.

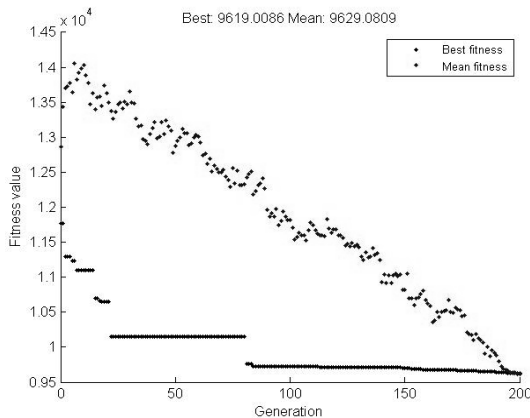


Fig. 5. First GA Experiment: with a 20 target database and 111 samples best weights has been determined for best classification results.

In both experiments equation (4) was used as fitness function.

$$\text{Cost} = 500 \times \text{NMC} + \sum_{i=1}^k d_i \tag{4}$$

Where NMC is number of misclassifications, d_i is distance of the i th sample from correct target and k is number of samples. A typical distance between target and probe image was about 50. Therefore, we should regulate the cost function with respect to that distance. As mentioned, we considered the number of misclassifications which is more important than the distance. We have empirically used the coefficient of 500 in the equation to consider the importance. Due to adaption of the genetic algorithm, little changes in the coefficient do not have any effect on the final results.

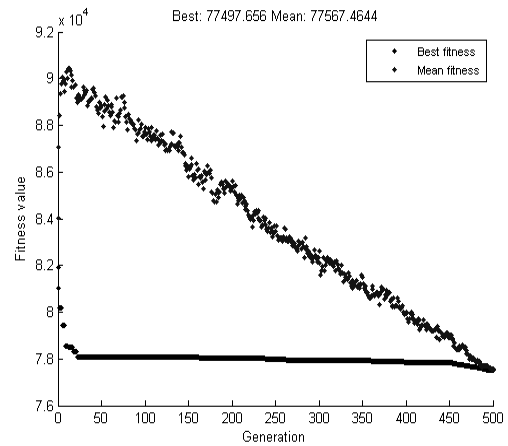


Fig. 6. Second GA Experiment: with a 197 target database and 502 samples best weights has been determined for best classification results.

4.2 Classification results

After obtaining optimal weights, classification is applied to FERET database with 699 stored targets. One image of fa series is chosen as target and other fa, fb, hl, hr, ql, qr, ra, rb, rc, rd, re has been tested separately. As we may not expect to recognize a 90 degree rotated face from a frontal face image (an example of the situation is shown in Fig. 7), recognition percentage is very low in this situation. Frontal facial image in different illumination, hair style and clothing can be recognized with a high correct decision percentage using the proposed method, and recognition percentage is reduced in series other than fa and fb. Table (2) shows detection and recognition results for different series. An example of some FERET image series has been shown in Fig. 8.

4.3 Method comparison

Most of recent works on face recognition have implemented more than one image per person as training set [21,22]. Q. Gao Et. Al. [23] proposed novel subspace method called sequential row-column independent component analysis (RC-ICA) for face recognition and implemented experiments 400 gray level frontal view from 200 persons from FERET database. One fa was used as training and an fb image as probe for each person. PCA, 2D-PCA, BDPCA, W-BDPCA, ICA, EICA and the proposed row-ICA and RC-ICA methods are used for feature extraction, and then a nearest neighbor classifier is employed for classification. Fig.9 shows experimental

result of the experiment in addition to our proposed hybrid WGPCA method. Results clarify that our proposed method outperforms all those methods in spite of more target classes and more variation of probe face images.

Movellanet. al. [24] reported face recognition results using PCA and ICA for three different conditions of same day different expression, different day similar expression and different day different expression. Most of correct recognition percentage was under 90%. In these experiment best results is obtained in same environmental condition and deferent facial expression that is about 90% of correct classification.



Fig. 7. Left picture shows one of re faces and right picture shows one of fa faces as it is clear from their names.

Table 2: Detection and recognition results for different series of feret database with 699 target classes having one fa series facial image per class

Series name	Detection	Recognition
fa	99.72%	97.40%
fb	99.76%	94.10%
hl	33.57%	8%
hr	38.62%	9.40%
ql	77.90%	35.82%
qr	94.20%	54.25%
ra	42.34%	8.51%
rb	91.89%	54.90%
rc	99.06%	66.82%
rd	61.76%	12.70%
re	18.62%	0%

As it is shown in Fig.10, 94.1% of correct decisions over different day, expression, lighting and size is better than all results of the report.

5. Timing profile

For the final experiment, we designed a full automatic application using Delphi XE2 programming environment and the proposed algorithm. Completely apart from the weight determination stage, speed is a critical issue in the current experiment. Therefore, we developed a standalone application that automatically detects and recognizes faces as a known target or an unknown face. This stage does not need to use MATLAB or any of its toolboxes. The only connection between this real time application and previous mentioned parts is weighting vector which is found using MATLAB optimization toolbox, and used in real time application. Table 3 shows the timing profile of the final recognizer application. As depicted in table 3, if normal HAAR-Like algorithm is applied without considering possible head rotation around roll¹ axis it took up 194ms to train a new class and about 194ms to

¹ Axis perpendicular to frontal view plane of face

extract features and compare an input face image to 50 predefined classes using core 2dou 2.4 with 2Gof RAM computer system. Using a multi-threaded application some of items in the timing profile such as distance calculations and PCA can be executed in parallel. This will significantly reduce the process time.



Fig. 8. An Example for some FERETImage Series

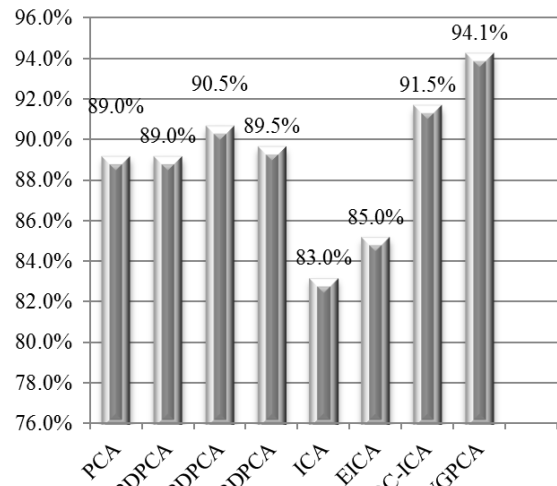


Fig. 9. Comparison of WGPCA with Other Methods Using fa and fb FERET Series

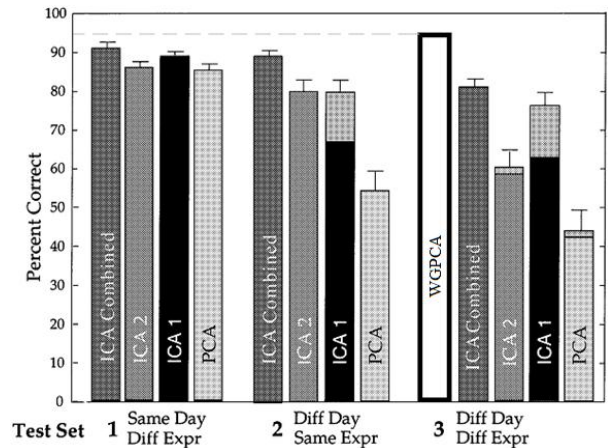


Fig. 10. Face Recognition Performance of WGPCA and Reported Results of Movellanet. Al.

Table 3 timing profile of the recognizer application

Task	Time(ms)
1 Light and contrast normalization	1
2 HAAR-Like	36
3 HAAR-Like with rotation	222
4 Point extraction	138
5 Distance measure and multiplication to weights	5
6 Calculating head rotation and distance correction	2
6 PCA	10
7 Array saving	1
8 comparison to 50 target templates using KNN	1

6. Conclusions

This research shows the influence of using multi-core architecture to reduce the execution time and thus increase performance of some software fault tolerance techniques. According to superiority of N-version Programming and Consensus Recovery Block techniques in comparison with other software fault tolerance techniques, implementations were performed based on these two methods. Finally, the comparison between the two methods listed above showed that the Consensus Recovery Block is more reliable. Therefore, in order to improve the performance of this technique, we propose a technique named Improved Consensus Recovery Block technique. In this research, satellite motion system which known as a scientific

computing system is consider as a base for our experiments. Because of existing any error in calculation of system may result in defeat in system totally, it shouldn't contains any error. Also the execution time of system must be acceptable. In our proposed technique, not only performance is higher than the performance of consensus recovery block technique, but also the reliability of our proposed technique is equal to the reliability of consensus recovery block technique. The improvement of performance is based on multi-core architecture where each version of software key units is executed by one core. As a result, by parallel execution of versions, execution time is reduced and performance is improved.

References

- [1] A. Parsi, M. Salehi, A. Doostmohammadi, "Swap Training: A Genetic Algorithm Based Feature Selection Method Applied on Face Recognition System," *World of Computer Science and Information Technology Journal*, pp. 125-130, 2012.
- [2] X. Tan, S. Chen, Z. Zhou, F. Zhang, "Face recognition from a single image per person: A survey," *The journal of pattern recognition society*, pp. 1725-1745, 2006.
- [3] S. H. Lin, S.Y. Kung, "Face recognition/detection by probabilistic decision-based neural network", *IEEE Trans. Neural Networks* 8 (1), pp. 114–132, 1997.
- [4] L. Liu, N. Sang, S. Yang, R. Huang, "Real-time skin color detection under rapidly changing illumination conditions," *IEEE Trans. Consumer Electronics*, Vol. 57, pp. 1295- 1302, 2011.
- [5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," In *Proc. of IEEE conference on computer vision and Pattern Recognition*, pages 38-44, 1998.
- [6] C. Kim, S. I. Choi, M. Turk, C. H. Choi, "A New Biased Discriminant Analysis Using Composite Vectors for Eye Detection," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 42, pp. 1095-1106, 2012.
- [7] A. J. Goldstein, L. Harmon, and A. Lesk, "Identification of human faces," *Proceedings of the IEEE*, pp. 748-760, 1971.
- [8] L. D. Harmon, M. K. Khan, R. Lasch, and P. F. Ramig, "Machine identification of human faces," *Pattern Recognition*, pp. 97-110, 1981.
- [9] S. Chitra, G. Balakrishnan, "A survey of face recognition on feature extraction process of dimensionality reduction techniques," *Journal of Theoretical and Applied Information Technology*, Vol. 36 No.1, Feb. 2012.
- [10] X. Fan and B. Verma, "Selection and fusion of facial features for face recognition," *Expert Systems with Applications: an International Journal*, vol. 36, pp. 7157-7169, Apr. 2009.
- [11] N. Hundewale "Face recognition using combined global local preserving projections and compared with various methods", *International Journal of Scientific & Engineering Research*, Volume 3, Issue 3, Mar. 2012.
- [12] J. D. Kelly, J. L. Davis, "A hybrid genetic algorithm for classification," *Learning and Knowledge Acquisition*, pp. 645-650, 1991.
- [13] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed., Academic Press, New York, 1990.
- [14] J. H. Holland (1975) *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan; re-issued by MIT Press (1992).
- [15] R. Lienhart, A. Kuranov, V. Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection". *Microprocessor Research Lab, Intel Labs*. [online]. Available:<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.139.4825&rep=rep1&type=pdf>
- [16] L. Dang, F. Kong, "Facial feature point extraction using a new improved Active Shape Model," *Image and Signal Processing (CISP)*, 2010 3rd International Congress, vol. 2, pp. 944-948.
- [17] P. J. Phillips, H. Moon, S. Rizvi, P. Rauss, "The FERET evaluation methodology for face recognition algorithms," *IEEE Trans. Pattern Analysis*, pp. 1090–1103, 2000.
- [18] F. Samara, A. Harter, "Parameterisation of a stochastic model for human face identification," *Proceedings of Second IEEE Workshop on Applications of Computer Vision*, pp. 138-142, Dec. 1994.
- [19] Dr Libor Spacek *Computer Vision Science Research Projects*, Available: <http://cswww.essex.ac.uk>
- [20] D. Mohamad, "Multi local feature selection using genetic algorithm for face identification," *International Journal of Image Processing*, vol. 1, pp. 1-10, 2008.
- [21] J. Wang, J. You, Q. Li, Y. Xu, "Orthogonal discriminant vector for face recognition across pose," *Pattern Recognition* 45, pp. 4069–4079, 2012.
- [22] I. Naseema, R. Togneri, M. Bennamoun, "Robust regression for face recognition," *Pattern Recognition* 45, pp. 104-118, 2012.
- [23] Q. Gao, L. Zhang, D. Zhang, "Sequential row– column independent component analysis for face recognition" *Neurocomputing* 72, pp.1152-1159, 2009.
- [24] M. S. Bartlett, J. R. Movellan, T. J. Sejnowski, "Face recognition by independent component analysis" *IEEE Trans. On neural networks*, vol. 13, no. 6, pp. 1450-1464, Nov. 2002.

Hasan Farsi received the B.Sc. and M.Sc. degrees from Sharif University of Technology, Tehran, Iran, in 1992 and 1995, respectively. Since 2000, he started his Ph.D in the Centre of Communications Systems Research (CCSR), University of Surrey, Guildford, UK, and received the Ph.D degree in 2004. He is interested in speech, image and video processing on wireless communications. Now, he works as associate professor in communication engineering in department of Electrical and Computer Eng., university of Birjand, Birjand, IRAN. His Email is: hfarsi@birjand.ac.ir.

Mohammad Hasheminejad received the B.Sc. degree in Bio-electrical engineering from University of Isfahan, Isfahan, Iran, in 2003. He received the M.Sc. degree in communication engineering from Maleke ashtar university of technology, Tehran, Iran, in 2008. He is currently Ph.D student in Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran. His area research interests include Image Processing and retrieval, Pattern recognition, Digital Signal Processing and Sparse representation. His email address is: mhashemi@birjand.ac.ir.