Int. J. Advance. Soft Comput. Appl., Vol. 1, No. 2, November 2009
ISSN 2074-8523; Copyright © ICSRS Publication, 2009
www.i-csrs.org

Enhancement of the Face Recognition Using a Modified Fourier-Gabor Filter

Essam Al Daoud

Computer Science Department
Zarka Private University, Jordan
e-mail: essamdz@zpu.edu.jo

Abstract

A modified Fourier-Gabor filter is used to enhance the classification rate of the face recognition. To verify the effectiveness of the proposed method, five well known methods are applied to four datasets; the methods are implemented without and with the suggested filter. The datasets consist of varying lighting conditions, different facial expressions, configuration, orientations and emotions. The experiments show that using the suggested Fourier-Gabor filter enhances the classification rates for all methods, all datasets and all training/testing percentage. The highest classification rates are obtained by using Fourier-Gabor filter with batch linear discriminant analysis (FG-Batch-ILDA), where the average classification rate over the four datasets is 93.8, the next is 93.77 by using Fourier-Gabor filter with linear discriminant analysis (FG-LDA) and 90.85 by using Fourier-Gabor filter with support vector machine (FG-SVM).

Keywords: *Fourier Transform, Gabor Filter, Face Recognition, Linear Discriminant Analysis, Principal Component Analysis, Support Vector Machine.*

1 Introduction

Face Recognition is the process of matching a face to one of many faces in the dataset. Although the current systems are still far away from the capability of the human perception system, numerous techniques have been proposed and much progress has been made toward recognition faces under small variation in illumination, lighting, facial expressions and orientations. Heisele et. al. presented a component-based method and two global methods for face recognition and evaluate them with respect to robustness against pose changes. The component system outperformed both global systems on all tests [1]. Lim and Reinders proposed a method that automatically finds human faces as well as its landmark points in color images based on a fuzzy analysis [2]. Pang et. al. introduced a novel face recognition method based on Gabor-wavelet and linear discriminant analysis [3]. Vukadinovic and Pantic presented a method for fully automatic detection of 20 facial feature points in images of expressionless faces using Gabor feature based boosted classifiers [4]. Bartlett et. al. used a version of ICA derived from the principle of optimal information transfer through sigmoidal neurons. thier method performed on face images in the FERET database under two different architectures, one which treated the images as random variables and the pixels as outcomes, and a second which treated the pixels as random variables and the images as outcomes. Both ICA representations were superior to representations based on PCA for recognizing faces across days and changes in expression [5]. Tan and Triggs showed that combining two of the most successful local face representations, Gabor wavelets and Local Binary Patterns (LBP), gives considerably better performance than either alone: they are complimentary in the sense that LBP captures small appearance details while Gabor features encode facial shape over a broader range of scales [6]. Sahoolizadeh et. al. used combination of PCA and LDA to improve the capability of LDA when a few samples of images are available, neural classifier is used to reduce number misclassification caused by non-linearly separable classes. Their method was tested on Yale face database [7].

2 Principle Component Analysis (PCA)

PCA is a procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. Thus, PCA can be used to capture as much as possible of the variability of the face image. Assume that a set of N sample images $\{x_1, x_2, ..., x_N\} \in \Re^n$, and the mean image:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

And let $\delta_i = x_i - \mu$ be the difference between the input image and the mean, then the covariance matrix C can be calculated as following:

$$C = \frac{1}{N}\sum_{i=1}^{N} \delta_i \delta_i^T$$

PCA determines the orthogonal projection U in (U is the eigenfaces matrix or features matrix, see Fig. 1):

$$y_k = U^T x_k \qquad k = 1,2,...,N$$

That maximizes tr($U^T CU$). This leads to an eigenvalue problem CU=UΛ where Λ=diag(y_1, y_2,..., y_k) is the matrix of eigenvalues of C. Now the eigenfaces matrix can be used to recognize the new faces such that the face x is similar to the face x_b where $||U^T(x-\mu)-U^T \delta_b||^2$ is the minimum.

Fig 1. Sample eigenfaces

3 Linear Discriminant Analysis (LDA)

LDA selects eignvectors U in such a way that the ratio of the between-class scatter and the within class scatter is maximized. PCA on the other hand does not take into account any difference in class. LDA computes the projection U that maximizes the ratio:

$$U_{opt} = \arg \max_U \frac{|U^T S_B U|}{|U^T S_W U|}$$

Where S_B and S_W are the between class scatter matrix and the within class scatter matrix respectively, such that:

$$S_B = \sum_{i=1}^{N} N_i(\mu_i - \mu)(x_k - \mu)^T$$

and

$$S_W = \sum_{i=1}^{M} \sum_{x_k \in C_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

M is the number of the classes, N_i is the number of samples in class i and μ_i is the mean of class i. U_{opt} can be found by solving the generalized eignvalue problem.

LDA assumes that the whole dataset is given in advance, and is trained in one batch. However, in a streaming environment, new samples are being presented continuously, possibly without end. The addition of these new samples will lead to the changes of the original mean vector μ, within class scatter matrix S_W, as well as between-class distance matrix S_B, therefore the whole discriminant eigenspace model should be updated. Thus BATCH-ILDA model can be described as following: Let X and Y be two sets observations, where X is the presented observation set, and Y is a set of new observations. Let their discriminant eigenspace models be $\Omega = (S_{wx}, S_{Bx}, \mu_x, N)$ and $\Psi = (S_{Wy}, S_{By}, \mu_y, L)$, respectively. This updating problem is to compute the new fisherspace model $\Phi = (S_{wv}, S_{Bv}, \mu_v, N + L)$ using fisherspace models Ω and Ψ, for more details see [8].

4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a binary classification method that finds the optimal linear decision surface between two classes. Given some training data, a set of points of the form

$$D = \{(x_i, c_i) \mid x_i \in \Re^p, c_i \in \{-1,1\}\}_{i=1}^{N}$$

where the c_i is either 1 or −1, indicating the class to which the point x_i belongs. Each is a p-dimensional real vector. SVM finds the maximum-margin hyperplane which divides the points having $c_i = 1$ from those having $c_i = -1$. Thus the linear decision surface is :

$$wx+b=0$$

where

$$w = \sum_{i=1}^{m} \alpha_i y_i s_i$$

s_i are the support vectors. The above holds when the data (classes) is linearly separable. The non-linear decision surface changes is:

$$w = \sum_{i=1}^{m} \alpha_i y_i K(s_i, x) + b = 0$$

Where K represents a Kernel. It could be a Radial Basis (Gaussian) Kernel, A linear Kernel, A polynomial Kernel or a custom Kernel. A multi-class pattern recognition system can be obtained by combining two class SVMs. Usually there are two schemes for this purpose. One is the one-against-all strategy to classify between each class and all the remaining; The other is the one-against-one strategy. Guo et. al. [9] proposed a bottom-up binary tree for classification. Suppose there are multi-classes in the data set.By comparison between each pair, one class number is chosen representing the "winner" of the current two classes. The selected classes will come to the upper level for another round of tests. Finally, the unique class will appear on the top of the tree.

5 Independent Component Analysis (ICA)

ICA is a statistical and computational technique for revealing hidden factors that underlie sets of random variables or observations. Typical algorithms for ICA use centering, whitening, and dimensionality reduction as preprocessing steps in order to simplify and reduce the complexity of the problem for the actual iterative algorithm. Whitening and dimension reduction can be singular value decomposition. Whitening means they are linearly transformed so that the components are uncorrelated and has unit variance. ICA algorithm is speeded up by including a "sphering" step prior to learning. The row means are subtracted, and then is passed through the whitening matrix , which is twice the inverse square root of the covariance matrix:

$$W_z = 2 * (Cov(X))^{-1/2}$$

This removes the first and the second-order statistics of the data; both the mean and covariances are set to zero and the variances are equalized. Thus, W^{-1} the inverse of the weight matrix can be interpreted as the source mixing matrix and the U=WX variables can be interpreted as the maximum-likelihood (ML) estimates of the sources that generated the data [5].

6 Fourier-Gabor Filter

The Fourier transform plays a critical role in a broad range of image processing applications. The output of the transformation represents the image in the *Fourier* or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image. For a square image of size $N \times N$, the two-dimensional DFT is given by:

$$Fourier(m, n, imgA) = \frac{1}{N^2} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} imgA(a,b) \; e^{-i2\pi(\frac{ma}{N}+\frac{nb}{N})} = FourierB(m, n)$$

where *imgA(a,b)* is the image in the spatial domain and the exponential term is the basis function corresponding to each point *FourierB(m,n)* in the Fourier space. *FourierB(0, 0)* represents the DC-component of the image which corresponds to the average brightness and *FourierB(N-1,N-1)* represents the highest frequency. In a similar way, the Fourier image can be re-transformed to the spatial domain. The inverse Fourier transform is given by:

$$img(m,n,\ FourierB\ ) = \frac{1}{N^2} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} FourierB(a,b)\ e^{-i2\pi(\frac{ma}{N}+\frac{nb}{N})} = imgA(m,n)$$

Gabor filters are directly related to Gabor wavelets, since they can be designed for number of dilations and rotations. A filter bank consisting of Gabor filters with various scales and rotations is created. The filters are convolved with the signal, resulting in a so-called Gabor space. The Gabor filter most commonly used in face recognition have the form [6]:

$$Gabor(x, y,\ \mu,\ \nu\ ) = \frac{\rho^2}{\sigma^2} \exp\left(- \frac{\rho^2 (x^2 + y^2)}{2\sigma^2} \right) \left(\exp(i\rho * (x, y) - \exp\left(\frac{-\sigma^2}{2} \right) \right)$$

Where

$$\rho = \frac{\pi}{2(\sqrt{2})^{\nu}} e^{i\frac{\pi\mu}{8}}$$

The output locations for each Gabor sub-matrix are specified by *x* and *y*. μ and ν define the orientation and scale of each sub-matrix. In this study, x and y $\in$ {1,2…32}, $\mu \in \{0,1,\ldots 7\}$ and $\nu \in \{0,1,\ldots, 4\}$. Thus 40 Gabor sub-matrix is generated and the size of each one is 32 ×32. Algorithm 1 describes the suggested Fourier-Gabor filter:

Algorithm 1: Face Recognition with Fourier-Gabor filter

Input: Training Set: $\{TR\}_{k=1}^{tm}$, Testing Set: $\{TS\}_{t=1}^{sm}$ and The Method: MD

Output: Classification Rate

Steps:

1- Prepare 5 × 8 Gabor sub-matrix, the size of each sub-matrix is 32 ×32. Call it *GaborBlock(i , j)*, where *i=1,2..5* and *j=1,2,…,8*.(see Figure 2)

2- Apply Fourier transform to each block

FG_B(i,j)=Fourier(m, n, GaborBlock(i , j))

3- For each image in the training set

a. Resize TR_k to 32×32 by using Bilinear interpolation *b.* Apply Fourier transform to TR_k

$$F\_TR_k=Fourier(m, n, TR_k)$$

c. Multiply $F\text{-}TR_k$ by each $FG\text{-}B(i,j)$

$$FG\_TR_k(\mathrm{i,j})= FG\text{-}B(i,j)*F\text{-}TR_k$$

d. Constract the matrix *FG_IMG* from the sub matrices. Thus, the size of this matrix is 160 ×256 (see Fig. 3).
e. Resize the matrix FG_IMG to 100× 100
f. Reshape *FG_IMG* as one column 10000 ×1, and accumulate it with the previous image in the training set. Call the accumulator matrix ACC. Thus, the size of this matrix is 10000×*tm*

4- Use the method MD {LDA, PCA, Batch-ILDA, SVM or ICA} to extract the features
5- Apply N-Nearest Neighborhood to classify the Testing set and calculate the classification rate.

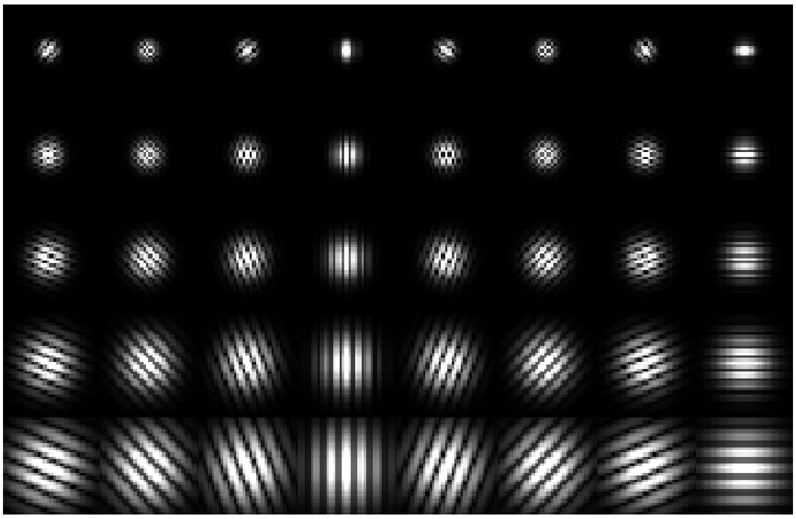

Fig 2. Gabor sub matrices in spatial domain (step1)

Fig 3. An image from ATT dataset after step 3.d is applied

7 Datasets

In this study four datasets are used: AT&T database of faces (ATT), Indian Face Database (IFD), Faces95 from Essex university database and Yale face database (Yale) [10-13]. Fig. 4 – Fig. 7 show samples faces from each dataset. All the datasets have many subjects and several images per individual, different facial expressions, configuration, orientations and emotions are used. The following is a brief description about the datasets:

- AT&T Dataset, formerly "The ORL Database of Faces" (ATT): Ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open, closed eyes, smiling, not smiling) and facial details (glasses, no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position.
- Indian Face Database (IFD): The database contains eleven different images of each of 40 distinct subjects. For some subjects, some additional photographs are included. All the images were taken against a bright homogeneous background with the subjects in an upright, frontal position. The files are in JPEG format. The size of each image is 640×480 pixels, with 256 grey levels per pixel. The images are organized in two main directories - males and females. In each of these directories, there are directories with name as a serial numbers, each corresponding to a single individual. The following orientations of the face are included: looking front, looking left, looking right, looking up, looking up towards left, looking up towards right, looking down. Emotions are: neutral, smile, laughter, sad/disgust.

- Faces 95 Used a fixed camera, a sequence of 20 images per individual was taken. During the sequence the subject takes one step forward towards the camera. This movement is used to introduce significant head (scale) variations between images of same individual. There is about 0.5 seconds between successive frames in the sequence. Number of individuals is 72 with significant lighting changes occur on faces due to the artificial lighting arrangement. Faces 95 Contains images of male and female, the background consists of a red curtain.
- The Yale face database contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink.

Fig 4. Sample faces from ATT Dataset

Fig 5. Sample faces from IFD Dataset

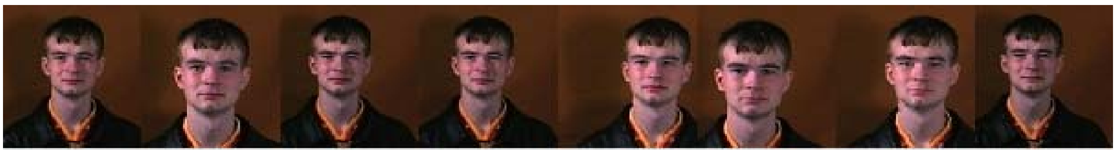

Fig 6. Sample faces from Faces95 Dataset

Fig 7. Sample faces from Yale Dataset

8 Experimental Results

Five standard well known methods are applied to the previous four datasets. Table 1-4 compare the classification rate without and with the suggested filter. Various methods and percentages of training/testing sets are used. For all experiments, the package from [14] and Matlab 7.0 are used to implement the five methods: LDA, PCA, Batch-ILDA, SVM and ICA without and with Fourier-Gabor filters. When

10% of ATT is used for training (one picture per subject for training and the rest for testing), the best classification rate without filter is 74.4 by using LDA, while the best classification rate with Fourier-Gabor filter is 80.2 by using Batch-ILDA, which consider very promising rate. if 50% of ATT is used for training (five picture per subject for training and the rest for testing), then the best classification rate without filter is 97.0 by using SVM, while the best classification rate with Fourier-Gabor filter is 98% by Batch-ILDA or LDA. The best classification rate when 90% of ATT is used for training, where the classification rate is 100% by using Fourier-Gabor filter with LDA, PCA, Batch-ILDA or VSM. Similar results are obtained when these methods and filter are applied to IFD, Faces95 and Yale dataset.

Table 1- Table 4 also show that using the suggested Fourier-Gabor filter enhances the classification rates for all methods, all datasets and all training/testing percentage. On the other Hand, if the classification rates in the datasets are compared, then it can be notice that Yale and ATT datasets are more recognizable than IFD and Faces95 for the training percentage10%- 70% (we can exclude the 90% because its usage is less realistic).

Table 1. The classification rate of ATT without and with the Fourier-Gabor filter

Method	**Percentage of Training Samples**									
	No Filter					**Fourier-Gabor Filter**				
	10	**30**	**50**	**70**	**90**	**10**	**30**	**50**	**70**	**90**
LDA	74.4	89.2	93.5	94.1	95.0	77.7	97.1	98.0	98.3	100
PCA	61.9	88.5	91.5	96.6	97.5	76.3	93.9	96.0	92.5	100
BATCH-ILDA	74.4	88.9	94.0	95.8	97.5	80.2	97.1	98.0	97.5	100
SVM	74.1	90.7	97.0	96.6	95.0	78.3	96.3	97.0	95.8	100
ICA	59.7	74.2	83.0	86.6	92.5	69.4	89.1	90.5	92.5	97.5

Table 2. The classification rate of IFD without and with the Fourier-Gabor filter

Method	**Percentage of Training Samples**									
	No Filter					**Fourier-Gabor Filter**				
	10	**30**	**50**	**70**	**90**	**10**	**30**	**50**	**70**	**90**
LDA	70.4	79.2	84.5	90.5	96.6	82.5	85.6	89.8	95.5	96.6
PCA	66.5	71.7	75.9	81.1	95.0	79.0	80.6	82.8	91.1	96.6
BATCH-ILDA	71.1	77.5	82.5	87.2	95.0	82.3	84.4	90.0	93.8	96.6
SVM	69.3	77.5	84.2	90.0	91.6	82.3	84.1	87.1	93.3	96.6
ICA	55.4	66.7	74.6	81.1	90.0	69.7	71.3	80.2	92.2	93.3

Table 3. The classification rate of Faces95 without and with the Fourier-Gabor filter

Method	Percentage of Training Samples									
	No Filter					Fourier-Gabor Filter				
	10	**30**	**50**	**70**	**90**	**10**	**30**	**50**	**70**	**90**
LDA	25.7	54.0	61.2	60.0	68.0	62.2	86.0	90.0	94.0	98.0
PCA	27.7	52.8	58.8	60.0	74.0	53.5	79.4	80.1	82.6	92.0
BATCH-ILDA	25.7	54.0	61.2	60.0	74.0	62.2	87.4	90.0	90.0	98.0
SVM	27.7	55.1	61.6	65.3	74.0	53.5	82.2	87.3	89.3	100
ICA	25.7	41.7	48.4	46.6	64.0	50.2	78.5	82.5	87.3	94.0

Table 4. The classification rate of Yale without and with the Fourier-Gabor filter

Method	Percentage of Training Samples									
	No Filter					Fourier-Gabor Filter				
	10	**30**	**50**	**70**	**90**	**10**	**30**	**50**	**70**	**90**
LDA	78.5	83.8	89.3	75.5	93.3	90.3	91.4	97.3	97.7	93.3
PCA	78.5	84.7	90.6	84.4	93.3	88.0	88.5	92.0	86.6	93.3
BATCH-ILDA	77.7	83.8	89.3	82.2	93.3	90.3	91.4	97.3	97.7	93.3
SVM	77.7	81.9	90.6	88.8	93.3	90.3	91.4	92.0	88.8	93.3
ICA	65.1	66.6	82.6	71.1	86.6	72.3	76.1	86.6	73.3	93.3

Table 5 and Fig. 8 show that: when 50% of the four datasets is used for training, then the highest classification rates are obtained by using FG-Batch-ILDA, where the average classification rate over the four datasets is 93.8, the next is 93.77 by using FG-LDA, 90.85 by using FG-SVM, and 87.72 by using FG-PCA.

Table 5. The classification rate of the four datasets, where 50% is used for training

	ATT	**IFD**	**Faces95**	**Yale**	**Average**
LDA	93.5	84.5	61.2	89.3	82.125
FG-LDA	98	89.8	90	97.3	93.775
PCA	91.5	75.9	58.8	90.6	79.2
FG-PCA	96	82.8	80.1	92	87.725
BATCH-ILDA	94	82.5	61.2	89.3	81.75

FG-BATCH-ILDA	98	90	90	97.3	93.8
SVM	97	84.2	61.6	90.6	83.35
FG-SVM	97	87.1	87.3	92	90.85
ICA	83	74.6	48.4	82.6	72.15
FG-ICA	90.5	80.2	82.5	86.6	84.95

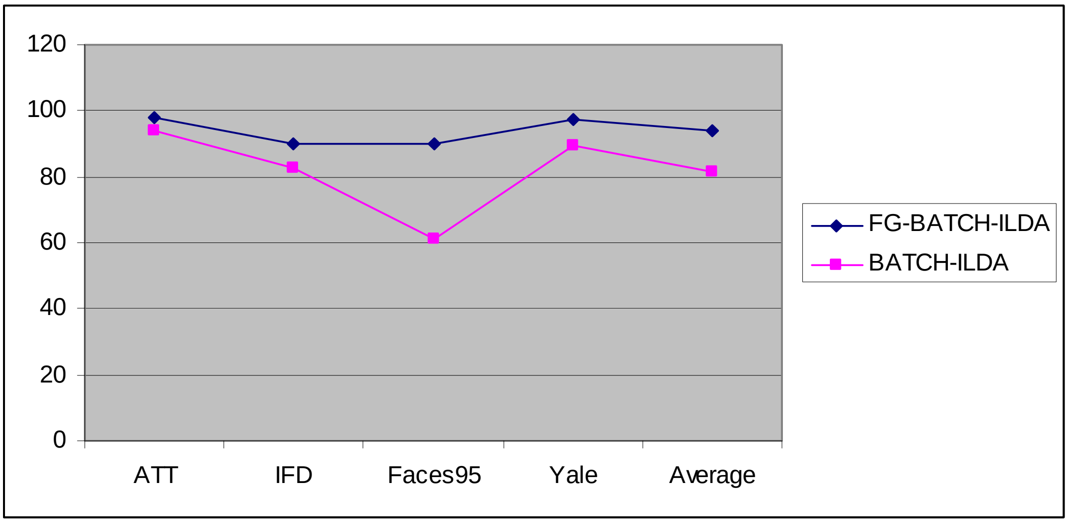

Fig 8. Comparison between Batch-ILDA without and with Fourier-Gabor filter

9 Conclusion

The performance of the proposed Fourier-Gabor filter was demonstrated on various datasets that contain several images per individual, different facial expressions, configuration, orientations and emotions. The results are quite promising; the enhancement in the classification rate is more than 10%. The future work should be dedicated to optimize the Fourier-Gabor parameters and combine the outcomes features with some other extracted features.

References

[1] Heisele, B., Ho, P. and Poggio T**.** Face recognition with support vector machines: global versus component-based approach**.** *Eighth IEEE International Conference on Computer Vision, ICCV 2001*. 2(3), 2001, pp: 688 - 694.

[2] Lim R. , Reinders M. J. Facial Landmarks Localization Based on Fuzzy and Gabor Wavelet Graph Matching. *FUZZ-IEEE*, 2001, pp. 683-686.
[3] Pang Y., Zhang L., Li M., Liu Z., and Ma W. A novel Gabor-LDA based face recognition method. *Lecture Notes in Computer Science,* vol. 3331, 2004, pp. 352-358.
[4] Vukadinovic D. and Pantic M. Fully Automatic Facial Feature Point Detection Using Gabor Feature Based Boosted Classifiers. *IEEE International Conference on Systems, Man and Cybernetics Waikoloa, Hawaii* October 10-12, 2005, pp:1692-1698.
[5] Bartlett M. S., Movellan J. R. and Sejnowski T. J. Face Recognition by Independent Component Analysis. *IEEE Transaction on Neural Networks,* 13(6), 2002, pp: 1450-1462.
[6] Tan X. and Triggs B. Fusing gabor and LBP feature sets for kernel-based face recognition. *In Analysis and Modeling of Faces and Gestures*, 2007, pp: 235-249.
[7] Sahoolizadeh A. H., Heidari B. Z., and Dehghani C. A New Face Recognition Method using PCA, LDA and Neural Network, *Proceedings of World Academy of Science Engineering and Technology*, 31, 2008, pp 7-12.
[8] Pang S., Ozawa S., and Kasabov N. Incremental Linear Discriminant Analysis for Classification of Data Streams. *IEEE Transactions on Systems, Man, and Cybernetics, Part B 35(5)*, 2005, pp: 905-914.
[9] Guo G., Li S. Z., and Chan K. Face Recognition by Support Vector Machines. *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition* 2000**,** pp: 196-206.
[10] AT&T http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html (last Accessed July 2009).
[11] Indian Face Database http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase/ (last Accessed July 2009).
[12] Faces95 from Essex university http://dces.essex.ac.uk/mv/allfaces/ (last Accessed July 2009).
[13] Yale face database http://cvc.yale.edu/projects/yalefaces/yalefaces.html (last Accessed July 2009).
[14] Becker, B.C., Ortiz, E.G. Evaluation of Face Recognition Techniques for Application to Facebook. *Proceedings of the 8th IEEE International Automatic Face and Gesture Recognition Conference*, 2008.