A Comparative Analysis of Different Classifiers for Face Recognition

Aftab Wahab, Salman Ahmed, Hameed Ullah Jan

Abstract— Face recognition is a challenging task as it involves treating a 3D object as a 2D image. In this paper, comparative analysis is performed for face recognition using different classifiers such as principal component analysis (PCA), linear discriminant analysis (LDA), Support vector machine (SVM), K nearest neighbor (KNN), Local histogram matching (LHM). Simulation examples are presented in which PCA and LDA classifiers have 98.5% recognition rate compared to other classifiers.

Index Terms— Principal Component Analysis (PCA), Principal component (PC), Olivetti Research Laboratory (ORL). Linear discriminant analysis (LDA), Support vector machine (SVM), K nearest neighbor (KNN), and Local histogram matching (LHM).

1 INTRODUCTION

The work on face recognition can be traced back to 1960's. The face recognition problem has been formulated as recognizing three dimensional objects (3D) to two dimensional (2D) images. Earlier it was treated as 2D pattern recognition problem that's why in mid 1970s it used typical pattern classification techniques i.e. the distance between important points in faces. During the 1980 work on face recognition remained largely dominant. Since 1990 it remained largely attractive due to its usage in surveillance related applications and research has focused on how to make face recognition fully automatic by tackling problems such as localization of a face in given image or video clip and extraction of features such as eyes, mouth etc. Among appearance-based approaches, Eigen faces and fisher faces have proved to be effective using large databases. Feature- based graph matching approaches have been quite successful. Feature based methods, compared to holistic approaches, is less sensitive to variations in illuminations and to inaccuracy in face localization but the problem is that feature extraction techniques for this approach are still not accurate, e.g. most of eye localization techniques assume some geometric and textural models and don't work if the eye is closed. The illumination and pose problems are two prominent issues for appearance based approach for recognizing a 3D object from 2D image.Principal component analysis (PCA) plays an important role in the image recognition and image compression. Its application lies in the authentication, identification, law enforcement, finger print recognition and pattern recognition. Main purpose of PCA is to reduce large dimensionality of observed variables into smaller number of artificial variables (principal components).

In section I the dataset am discussed, in Section II different approaches are discussed, in Section III implementation is discussed, in Section IV combination of PCA with other classifier is discussed, in Section V results and experience are discussed and in Section VI conclusion is drawn.

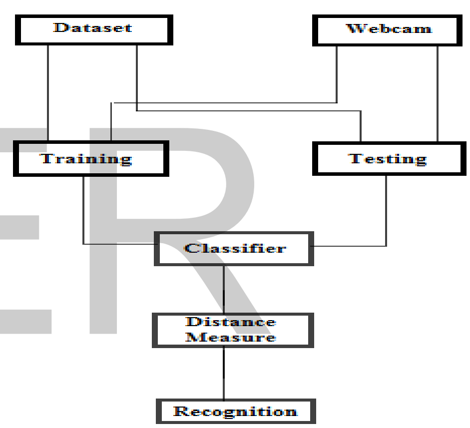

Fig. 1 shows the procedure of recognition.

2 METHODOLOGY

2.1 Dataset

The choice of an appropriate database to be used depends on the task. Currently, there is large number of databases such as Color FERET database, Yale Face database etc. For the large variations in illumination, age, pose such as rotation, occlusion etc. we prefer to use ORL database as literature survey contains most work has been carried out using it. We carried out our work using the popular database of faces known as "The ORL Database of Faces" developed by AT&T. It contains ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lightning, facial expressions (open or closed eyes, smiling or non-smiling) and facial details such as glasses or without glasses. All the images were taken against a dark homogenous background with the subjects in an upright, frontal position. All the files are in PGM format and the size of each image is 92 x 112 pixels, with 256 grey levels /pixel. The images are categorized in 40 directories (directory/person) in which 10 images of each person are presented totaling to 400 images.

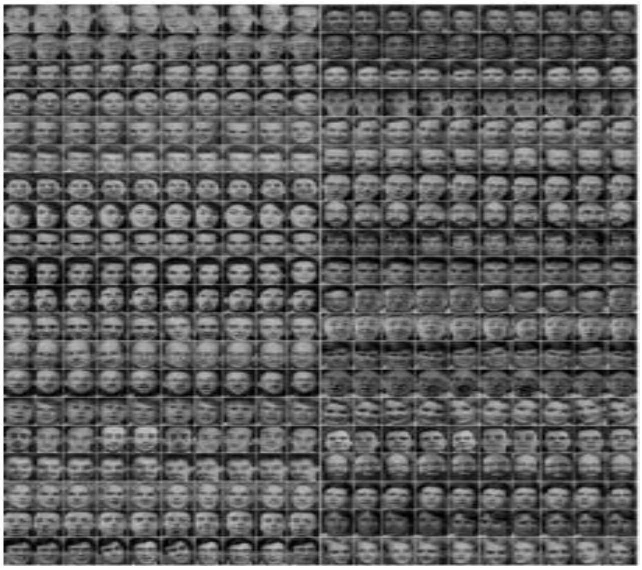

Fig. 2 shows an ORL dataset.

2.2 Different Approaches

The approaches so far in the literature can be classified as either model based and appearance based. The former approach extracts geometrical features while the latter uses intensity parameters such as Eigen faces (also called principal components). Among appearance-based approaches, Eigen faces and fisher faces have proved to be effective using large databases. Feature based methods, compared to holistic Prepare Your Paper before styling approaches, are less sensitive to variations in illuminations and to inaccuracy in face localization but the problem is that feature extraction techniques for this approach are still not accurate. E.g. most of eye localization techniques assume some geometric and textural models and don't work if the eye is closed.

Geometric Approach

The first traditional way to recognize people was based on face geometry. A lot of geometric features such as eye separation, mouth width, nose shape etc. were used for this purpose but recent methods have adopted natural geometric properties of eye as a basis for recognition.

PCA Based Face Recognition

PCA finds the optimal linear least-square representation in (N-1) dimension space, where N is the total facial images. The representation is characterized by a set of N-1 Eigen vectors and Eigen values. We normalize the Eigen vectors to make them orthogonal and then exclude the higher order Eigen vectors as they contain smaller variations. To take those Eigen values with higher covariance or variance.

LDA Based Face Recognition

LDA is an enhancement to PCA and it provides better classification than PCA. LDA does more of data classification unlike PCA which does feature classification. We combine LDA with PCA for better classification. We observed that pure LDA doesn't work well when the testing samples were from the persons not in the training set (experimentation using real time webcam) and when samples with different backgrounds were presented.

LHM Based Face Recognition

In LHM approach the histogram of the training and testing dataset is determined. The range of histogram is from 1 to 256 gray level values. Both histograms are in vector form. The mean of 9 consecutive frequencies are determined both for training and testing feature vector. The mean of both vectors is compared. Those images will be recognized where more matching is found.

KNN Based Face Recognition

K defined the range of neighbors of training dataset. For low and high value of K less recognition rate will be obtained. Maximum recognition rate will be obtained for medium value of K. First, find the training and testing datasets in KNN classifier. Both feature vectors will be compared. The comparison is done based on distance metrics. For recognition take those images with minimum distance. When the KNN is combined with PCA then its recognition reduces instead of increases.

SVM Based Face Recognition

SVM is a supervised learning model. In SVM the data is classified into classes. Here only the boundary of classes is considered. The main aim in SVM is to increase the separation between the boundaries of classes. The recognition rate greatly depends upon the separation of classes. When there is more separation then there will be maximum recognition rate. When SVM is combined with PCA then its recognition rate improves.

3 IMPLEMENTATION

To compute the distance between two vectors the distance measures are used.

The distance measures play an important role in the improvement of recognition rate. The distance measures that we have considered are Seuclidean, Euclidean, CityBlock, Cosine and Mahalanobis. These are discussed in [1].

3.1 Distance Metrics

To compute the distance between two vectors the distance measures are used.
The distance measures play an important role in the improvement of recognition rate. The distance measures that we have considered are Seuclidean, Euclidean, CityBlock, Cosine and Mahalanobis. These are discussed in [1].

- **Euclidean:**

$$DIS_{euclidean} = \sqrt{\sum_i (\text{test}_i - train_i)^2} \quad (1)$$

- **CityBlock:**

$$DIS_{CityBlock} = \sum_i |test_i - train_i| \quad (2)$$

- **Cosine:**

$$DIS_{Cosine} = \frac{\text{a}\,.\,\text{b}}{|a||b|} \quad (3)$$

- **Seuclidean:**

$$SD = \sqrt{\frac{\sum_{i=1}^{n} (\text{vector}_j - vector')^2}{m-1}} \quad (4)$$

$$DIS_{euclidean} = \sqrt{\sum_i (\text{test}_i - train_i)^2} \quad (5)$$

Equation (4) And Equation (5) will be passing in pdist function available in matlab.

3.2 Principal Component Analysis (PCA)

In this approach, the faces are presented in the lower dimensional space using the classifier principal component analysis (PCA). We use a method to extract features of an image using Eigen face method proposed by Turk and Pentland which is based on the Karhunen-Loeve Expansion (PCA). PCA is a technique that effectively and efficiently represents pictures of faces into its Eigen face components. The algorithm for the implementation of PCA is following.

The image faces in our database are

$$Dataset = X_1, \text{K}\ , X_m. \quad (6)$$

Now to obtain the training dataset from the dataset

$$Training\_Dataset = Training_1, \text{K}\ , \text{Training}_n \quad (7)$$

In Equation (7) n shows the total number of images in training dataset

Now to obtain the testing dataset from the dataset

$$Testing\_Dataset = Testing_1, \text{K}\ , \text{Testing}_n \quad (8)$$

Finding the mean of all training images of Eq. (6)

$$\psi = \frac{1}{m}\sum_{i=1}^{m} Training\_Dataset_i \quad (9)$$

In Equation (9) m is the total number images in the training dataset.

Next, subtract the mean from the training faces.

$$\phi_i = Dataset - \psi \quad (10)$$

Next, find the covariance of Equation (8)

$$\text{Covariance} = Cov(\phi_i) \quad (11)$$

Find the variance of the covariance matrix.

$$\text{Variance} = \sqrt{diag(\text{covariance})} \quad (12)$$

$$\boxed{R = \left(\frac{covariance}{variance\,variance'}\right)} \quad (13)$$

Find the Eigen values and Eigen vectors of R.

$$[eigenvectors, eigenvalues] = eig(\text{R}) \quad (14)$$

Now multiplying the eigenvectors with Equation (8)

$$Eigenfaces = \text{eigenvectors} * \phi_i \quad (15)$$

Now picking those Eigen vectors with largest Eigen values or with maximum variance

$$V = eigenfaces(:, \text{end} : -1 : \text{end} - (\text{N} - 1)) \quad (16)$$

In Equation (16) N is the number of principal component.
Formation of feature vectors can be determined by multiplying the transpose of Equation (10) with Equation (15).

$$feature\_\text{vector} = \phi^t{}_i * eigenfaces \quad (17)$$

$$P = Testing\_Dataset - \psi \quad (18)$$

$$Testing\_vector = P^t * V \quad (19)$$

$$vector = [\text{feature\_}\,vetor, \text{Testing\_vector}] \quad (20)$$

$$distance = pdist(\text{vector}, \text{distancemetric}) \quad (21)$$

$$Y = distance(1 : \text{Number\_Rows\_Training}) \quad (22)$$

$$Recognized\_image = \min(\text{Y}) \quad (23)$$

In Equation (22) in place of distance metric different distances can be passed.
Equation (23) shows the recognized image.
The image can be recognized on the basis of minimum distance.

Fig. 3 shows the Eigen Faces of different faces obtained by using the PCA classifier. These Eigenfaces can be obtained by using the Equation (15).

3.3 Linear Discriminant Analysis (LDA)

In this approach, the dataset is classified into classes. The components such as nose, eye, mouth etc are divided into classes. The algorithm for the LDA implementation is following.
Load all the images from the dataset as

$$Dataset = X_1, \ldots, X_n. \tag{24}$$

Taking transpose of dataset

$$Dataset = Dataset^t. \tag{25}$$

Give some of the images to the training dataset and some of the images to the testing dataset.

$$Training = Training_1, \ldots, \mathrm{Training}_n. \tag{26}$$

$$Testing = Dataset - Training. \tag{27}$$

Now taking some labels, which uniquely represent the rows in training dataset?

$$Labels = L_1, \ldots, \mathrm{L}_m. \tag{28}$$

Now using the built-in function in matlab

$$LDA\_groups = classify(\mathrm{Testing, Training, Labels, Type}) \tag{29}$$

Now taking the groups equal to the size of LDA groups

$$Groups = G_1, \ldots, \mathrm{G}_m. \tag{30}$$

The recognition rate can be found by comparing Equation (29) and Equation (30).
The recognition rate found by the above algorithm for LDA is 87.50 %, for Type= diagquadratic, Training=320 images and Testing=80 images. It means that it recognized 70 images correctly out of 80 images.

3.4 Local Histogram Matching (LHM)

In this approach find the histograms for the training and testing, and then compare the mean of the histograms of both.
The algorithm for the LHM is following.
Load all the images from the dataset.

$$Dataset = X_1, \ldots, X_n. \tag{31}$$

$$Training = Training_1, \ldots, \mathrm{Training}_n. \tag{32}$$

$$Testing = Dataset - Training. \tag{33}$$

First to find the histograms for the training and testing dataset
Gray levels are 256.
For I=1 to size rows
For j=1 to size cols;
Index=data (i, j);
Histogram (a) =histogram (a+1) +1;
End
End

Now finds the mean of 9 consecutive frequencies for both training and testing dataset.

$$Mean\_\mathrm{Train} = TM_1, \ldots, \mathrm{TM}_m. \tag{34}$$

$$Mean\_\mathrm{Test} = TestM_1, \ldots, \mathrm{TestM}_m. \tag{35}$$

Both the means are in vector form. Now compare both the vector of means. Those images will be presented as recognized images where more matching in the vectors is found. The recognition rate obtained by this algorithm is 99.75. It means that is recognized 399 images correctly out of 400 images.
For the training set of 200(odd) dataset and 200(even) testing dataset, it mismatches image 4 of person 17 with image 5 of person 7 as shown in **Figure 4.**

3.5 K nearest Neighbor (KNN)

In this approach find the feature vector of both training and testing and then compare the vector of both on the basis of distance metrics. The algorithm for the K nearest neighbor is following
Load all the images from the dataset as

$$Dataset = X_1, \ldots, X_n. \tag{36}$$

Taking transpose of dataset

$$Dataset = Dataset'. \tag{37}$$

Give some of the images to the training dataset and some of the images to the testing dataset.

$$Training = Training_1, \ldots, \mathrm{Training}_n. \tag{38}$$

$$Testing = Dataset - Training. \quad (39)$$

Now taking some labels, which uniquely represent the rows the training dataset?

$$Labels = L_1, \ldots, L_m. \quad (40)$$

Now using the built-in function in matlab for KNN classifier

$$KNN\_groups = (\text{Testing, Training, Labels, K, Distance, Rule}) \quad (41)$$

The recognition rate can be found by comparing Eq. (41) and Eq. (22).

The recognition rate found by the above algorithm for KNN is **97.50%,** for K=4, Distance = CityBlock, Rule=Nearest, Training=320 images and Testing=80 images. It means that it recognized 78 images correctly out of 80 images.

3.6 Support Vector Machine (SVM)

In this approach, the data is also classified into classes. But here only the boundary of the classes is considered. In case of LDA the whole class is considered. When there is more separation between the boundaries of classes then the classifier will work better. The lines are drawn between the classes. These lines are called hyperplanes. The algorithm for the implementation of SVM is following.
Load all the images from the dataset as

$$Dataset = X_1, \ldots, X_n. \quad (42)$$

Taking transpose of dataset

$$Dataset = Dataset'. \quad (43)$$

Give some of the images to the training dataset and some of the images to the testing dataset.

$$Training = Training_1, \ldots, Training_n. \quad (44)$$
$$Testing = Dataset - Training. \quad (45)$$

Now taking some labels, which uniquely represent the rows in the training dataset?

$$Labels = L_1, \ldots, L_m. \quad (46)$$

Now using the user defined function for the SVM classifier.

$$SVM\_\text{groups} = \text{multiclass(Training, Labels, Testing)} \quad (47)$$

The recognition rate can be found by comparing Equation (22) and Equation (47).
The recognition rate obtained by this algorithm is 91.25% for the training images of 320 while testing images of 80.

Fig. 4 shows the Eigen Faces for PCA with LDA, PCA with SVM and PCA with KNN.

4 COMBINATION OF PCA WITH OTHER CLASSIFIERS.

PCA greatly improves the recognition rate of other classifiers such as LDA, SVM and KNN. That's why its combination with classifiers is included in this paper.

4.1 PCA with LDA.

PCA with LDA greatly affect the recognition rate of LDA. PCA improves the recognition of PCA from **87.50%** to **100.0%.** The steps involved in the implementation of PCA with LDA are following.
Load all the images from the dataset as

$$Dataset = X_1, \ldots, X_n. \quad (48)$$

Now using from Equation (9) to Equation (17) on Equation (48) leads to Equation (49).
Now using from Equation (25) to Equation (30) on Eq. (49) gives the recognition rate of **100.0** for some values of principal components. It means that it recognized all of the testing images.

4.2 PCA with KNN.

PCA with KNN can't increase the recognition rate of KNN, but give the same recognition rate as given by the simple KNN but with different value of K (k=1). The steps involved in the implementation of PCA with KNN are following.
Load all the images from the dataset as

$$Dataset = X_1, \ldots, X_n. \quad (50)$$

Now using from Equation (9) to Equation (16) on Equation (50) leads to Equation (51).
Now using from Equation (30) to Equation (41) on Eq. (51) gives the recognition rate of **97.50%** for the training of 320 images and testing of 80 images. It means that it recognized 78 images correctly out of 80 images.

4.3 PCA with SVM

PCA with SVM greatly affect the recognition rate. It improves the recognition rate of SVM from **91.25%** to **95.0%.** The steps involved in the implementation of PCA with SVM are following.
Load all the images from the dataset as

$$Dataset = X_1, K\ , X_n . \qquad (52)$$

Now using from Equation (9) to Equation (17) on Equation (51) leads to Equation (52).

Now using from Equation (30) to Equation (47) on Equation (52) gives the recognition rate of **95.0** for training of 320 images and testing of 80 images. It means that it recognized 76 images correctly out of 80 images.

Figure 1.4 shows the Eigen values of different faces varying the principal component values from 1 to 10 by using the combination of PCA with LDA, PCA with SVM and PCA with KNN.

5 Comparison of Classifiers on the Basis of Results and Experiments

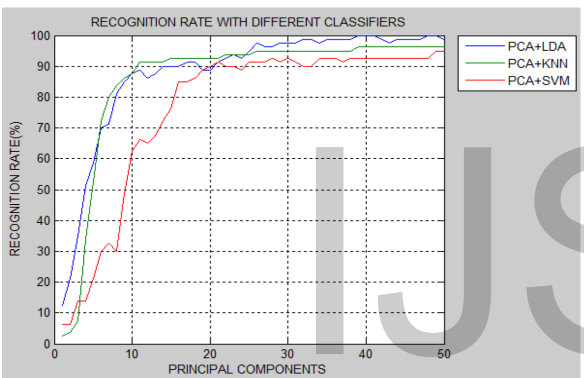

Fig. 5 shows the recognition rate for PCA with LDA, PCA with SVM and PCA with KNN with different principal components

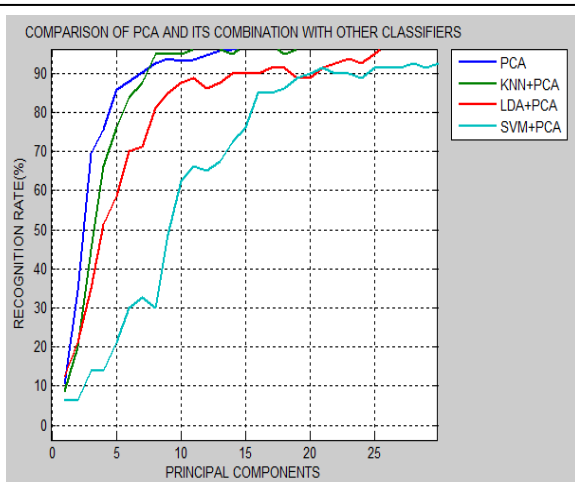

Fig. 7 shows the recognition rate of PCA, PCA with LDA and PCA with SVM. The principal components are from 1 to 30.

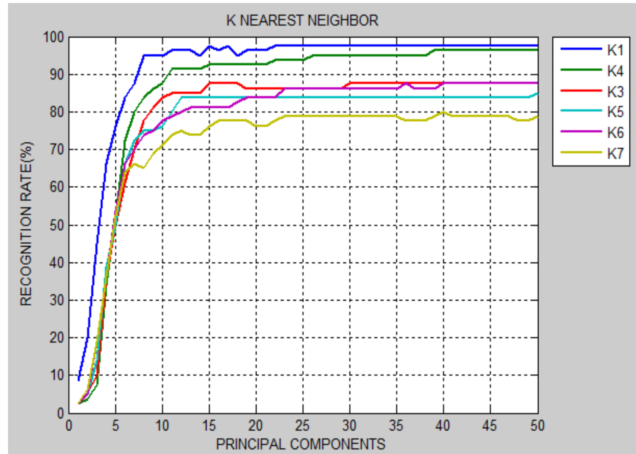

Fig. 6 shows the recognition rate for classifier KNN of different values of K. For k=1 it gives us the best recognition rate which is **97.50 %.**

Table 1.1 shows that when increasing the value of principal component (N) then the recognition rate also increases, but up to certain limit. After some value of principal component (N) the recognition rate tends to reduce. It can be concluding from this table not to take smaller or bigger value of principal component but to take medium value of N.

Principal Compo-nents	**Distance Metrics**				
Principal Compo-nents(N)	**Seucli dean (%)**	**Euclidean (%)**	**Cosine (%)**	**City block (%)**	**Mahala nobis (%)**
1	10.25	10.25	4.25	10.25	10.25
2	34.50	34.50	17.00	36.00	34.25
3	69.50	66.50	50.0	64.50	70.0
4	75.50	76.50	67.25	77.5	74.0
5	85.75	84.0	80.0	85.75	83.75
6	87.75	88.0	85.25	90.0	87.50
7	90.50	91.0	88.25	89.75	89.0
8	92.50	93.25	91.0	93.20	92.25
9	93.75	93.50	92.0	94.0	92.25
10	93.25	94.0	92.25	95.0	94.0
11	93.50	94.50	92.50	95.0	94.0
12	94.50	94.75	92.75	95.75	94.75
13	95.75	95.0	93.0	96.0	95
14	96.0	95.0	93.25	95.75	94
15	96.50	95.0	93.75	96.0	96
16	96.50	95.50	93.75	95.75	96
17	97.25	95.50	93.75	96.0	96
18	97.0	95.75	93.50	96.25	95.75
19	96.5	95.75	93.75	96.25	96.50
20	97.0	95.50	94.5	96.25	96.0
21	97.50	95.75	94.5	97.0	96.25
22	**97.75**	95.75	94.5	97.25	96.25

23	**97.75**	95.75	94.5	97.25	96.25
24	**97.75**	96.0	94.5	97.50	96.25
25	97.50	96.0	94.5	97.50	96.25
26	97.50	96.0	94.5	97.50	96.50
27	97.0	96.0	94.5	97.50	96.50
28	97.0	96.0	94.5	97.50	96.50
29	97.5	96.25	94.75	97.50	96.75
30	97.5	96.25	94.75	**97.75**	96.0

Table 1.1 Principal Component Analysis

Table 1.2 shows that changing the training dataset also affect the recognition rate. When increasing the training dataset then the recognition rate also increases, also when decreasing the training dataset, the recognition rate decreases.

Classifier	Training	Testing	RR (%)	Correctly	Incorrectly
LDA	280	120	88.33	100	20
LDA	320	80	87.50	70	10
LDA	360	40	90.0	36	04
KNN	280	120	97.50	117	03
KNN	320	80	97.50	78	02
KNN	360	40	95.0	38	02
SVM	280	120	88.33	106	14
SVM	320	80	91.25	73	07
SVM	360	40	90.0	36	04
LHM	200	200	99.50	199	01
LHM	160	240	99.0	238	02
LHM	120	280	98.75	273	07

Table 1.2 LDA, KNN, SVM, and LHM

6 CONCLUSION

From the results and experiments of this paper it can be seen that PCA greatly affect the recognition rate of all classifiers.
The recognition rate of all the classifiers can be improved by improving the recognition rate of PCA. The recognition rate of PCA totally depends on the distance measures, the variance used after the covariance matrix and the data type used in the implementation. The recognition rate of all the classifiers also depends on the training dataset.

REFERENCES

[1] Beveridge, R., Bolme, D., Tiexerira, M., and Draper, B., 2003, The CSU Face Identification Evaluation System User's Guide: Version 5.0, Colarado Stater University.

[2] M. Turk and A. Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.

[3] P.N. Belhumeur, J.P. Hespanha and D.J. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class-specific linear projection, IEE Trans. Pattern analysis and Machine intelligence, vol.19, no. 7 , pp. 711-720, 1997.

[4] Suman Kumar Bhattacharyya & Kumar Rahul, "Face Recognition by Linear Discriminant Analysis", International Journal of Communication Network Security, ISSN: 2231 – 1882, Volume-2, Issue-2, 2013, pp 31-35.

[5] M. J. Swain and D. H. Ballard, "Indexing via color histogram", In Proceedings of third international conference on Computer Vision (ICCV), pages 390–393, Osaka, Japan, 1990.

[6] Fazl-e-Basit, Younus Javed and Usman Qayyum, "Face Recognition using processed histogram and phase only correlation ", 3rd IEEE International Conference on Emerging Technology pp. 238-242.

[7] Hasan Demirel, Gholamreza Anbarjafari, "High Performance Face Recognition using Histogram Matching", EMO Bilim, sayı. 22, Kas. 2008, pp: 58-61.

[8] C. C. Burges, A tutorial on support vector machines for pattern recognition, In Proceedings of Int. Conference on Data Mining and Knowledge Discovery, 2(2), pp. 121-167, (1998)

[9] B. Zhao, Y. Liu, and S.W. Xia, Support vector machines and its application in handwritten numerical recognition, In Proceedings of 15thInt. Conference on Pattern Recognition, vol. 2, pp. 720-723, (2000).

[10] S. Bernhard, C.C. Burges, and A.J. Smola, Pairwise classification and support.

[11] G. Goos, J. Hartmanis, and J. van Leeuwen, Lecture Notes in Computer Science.

[12] ORL face database: AT &T Laboratories, Cambridge, U.K...[Online]. Available: http://www. Cam-orl.co. UK /facedatabse.html.

[13] Jackson JE. 1991. A user's guide to principal components. New York, NY: Wiley.

[14] Jolliffe IT. 2002. Principal component analysis, 2nd edn New York, NY: Springer-Verlag.

[15] Horn R, Johnson C. 1985. Matrix analysis. Cambridge, UK: Cambridge University Press.