Face Recognition using Neural Network and Eigenvalues with Distinct Block Processing

Amit Kumar, Prashant Sharma, Shishir Kumar
Department of CSE, Jaypee University of Engineering & Technology, Guna (MP) INDIA
amitrathi10@yahoo.co.in, sharma_prashant10@yahoo.co.uk, dr.shishir@yahoo.com

***Abstract* –Human face recognition has been employed in different commercial and law enforcement applications. It has also been employed for mug shots matching, bank-store security, crowd surveillance, expert identification, witness face reconstruction, electronics mug shots book, and electronic lineup. A face recognition system based on principal component analysis and neural networks has been developed. The system consists of three stages; preprocessing, principal component analysis, and recognition. In preprocessing stage, normalization illumination, and head orientation were performed. Principal component analysis is applied to obtained the aspects of face which are important for identification. Eigenvectors and eigenfaces are calculated from the initial face image set with the help of distinct block processing. New faces are projected onto the space expanded by eigenfaces and represented by weighted sum of the eigenfaces. These weights are used to identify the faces. Neural network is used to create the face database and recognize and authenticate the face by using these weights. In this paper, a separate network was developed for each person. The input face has been projected onto the eigenface space first and new descriptor is obtained. The new descriptor is used as input to each person's network, trained earlier. The one with maximum output is selected and reported as the equivalent image.**

Key Words: Eigenface, eigenvector, eigenvalue, Neural network, distinct block processing, face recognition.

I. INTRODUCTION

The face is our primary focus of attention in social intercourse, playing a major role in conveying identity and emotion. Although the ability to infer intelligence or character from facial appearance is suspect, the human ability to recognize faces is remarkable. We can recognize thousands of human faces[1,2,3] learned throughout our lifetime and identify familiar faces at a glance even after years of separation. This skill is quite robust, despite large changes in the visual stimulus due to viewing conditions, expression, aging, and distractions such as glasses or change in hairstyle or facial hair. As a consequence, the visual processing of human faces has fascinated philosophers and scientists for centuries.

Computational models of face recognition[5,7,8,9,10,11], in particular, are interesting because they can contribute not only to theoretical insights but also to practical applications. Computers that recognize faces could be applied to a wide variety of problems, including criminal identifications, security systems, image and film processing and human – computer interaction. For example, the ability to model a particular face and distinguish it from a large number of stored face models would make it vastly improve criminal identification. Even the ability to merely detect faces, as opposed to recognizing them can be important. Detecting faces in photographs[20], for instance, is an important problem in automating color film development, since the effect of many enhancement and noise reduction techniques depends on the picture content.

Unfortunately, developing a computational model of face recognition is quite difficult, because faces are complex, multidimensional and meaningful visual stimuli. Thus unlike most early visual functions, for which we may construct detailed models[28] of retinal or striate activity, face recognition is a very high level task for which computational approaches can currently only suggest broad constraints on the corresponding neural activity.

We, therefore, focused my paper towards implementing a sort of early, pre-attentive pattern recognition[11,42] capability that does not depend on having three-dimensional information or detailed geometry. Our goal is to develop a computational model of face recognition which would be fast, reasonably simple, and accurate in constrained environments such as an office or a household. In addition, the approach is biologically implementable and in concern with preliminary findings in the physiology and psychology of face recognition. The scheme is based on an information theory approach that decomposes face images into a small set of characteristics feature images called "eigenfaces"[12,23,24,34,36], which may be though of as the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the eigenfaces ("face space") and then classifying the face by comparing its position in face space with the positions of known individuals.

II. Background And Related Work

Much of the work in computer recognition[9,44] of faces has focused on detecting individual features[5,9,43] such as the eyes, nose, mouth, and head outline, and defining a face model by the position, size, and relationships among these features. Such approaches have proven difficult to extend to multiple views and have often been quite fragile, requiring a good initial guess to guide them. Research in human strategies of face recognition, moreover, has shown that individual features and their immediate relationships comprise an insufficient representation to account for the performance of adult human face identification. Nonetheless, this approach to face recognition remains the most popular one in the computer vision literature.

Bledsoe[16,17] was the first to attempt semi-automated face recognition with a hybrid human-computer system that classified faces on the basis of fiducially marks entered on photographs by hand. Parameters for the classification were normalized distances and ratios among points such as eye corners, mouth corners, nose tip, and chin point. Later work at Bell Labs developed a vector of up to 21 features, and recognized faces using standard pattern classification techniques.

Fischler and Elschlager[40], attempted to measure similar features automatically. They described a linear embedding algorithm that used local feature template matching and a global measure of fit to find and measure facial features. This template matching approach has been continued and improved by the recent work of Yuille and Cohen. Their strategy is based on deformable templates, which are parameterized models of the face and its features in which the parameter values are determined by interactions with the face image.

Connectionist approaches to face identification seek to capture the configurationally nature of the task. Kohonen and Lehtio[30,31] describe an associative network with a simple learning algorithm that can recognize face images and recall a face image from an incomplete or noisy version input to the network. Fleming and Cottrell[19] extend these ideas using nonlinear units, training the system by back propagation.

Others have approached automated face recognition by characterizing a face by a set of geometric parameters and performing pattern recognition based on the parameters. Kanade's face identification system[20] was the first system in which all steps of the recognition process were automated, using a top-down control strategy directed by a generic model of expected feature characteristics. His system calculated a set of facial parameters from a single face image and used a pattern classification technique to match the face from a known set, a purely statistical approach depending primarily on local histogram analysis and absolute gray-scale values.

Recent work by Burt[26] uses a smart sensing approach based on multiresolution template matching. This approch to fine strategy uses a special purpose computer built to calculate multiresolution pyramid images quickly, and has been demonstrated identifying people in near real time.

III. Methodlogy Employed

A. Eigen Face Approach

Much of the previous work on automated face recognition has ignored the issue of just what aspects of the face stimulus are important for face recognition. This suggests the use of an information theory approach of coding and decoding of face images, emphasizing the significant local and global features. Such features may or may not be directly related to our intuitive notion of face features such as the eyes, nose, lips, and hair.

In the language of information theory, the relevant information in a face image is extracted, encoded as efficiently as possible, and then compared with a database of models encoded similarly. A simple approach for extracting the information contained in an image of a face is to somehow capture the variation in a collection of face images, independent of any judgment of features, and use this information to encode and compare individual face images.

In mathematical terms, the principal components of the distribution of faces, or the eigenvectors of the covariance matrix of the set of face images, treating an image as point (or vector) in a very high dimensional space is sought. The eigenvectors are ordered, each one accounting for a different amount of the variation among the face images. These eigenvectors can be thought of as a set of features that together characterize the variation between face images. Each image location contributes more or less to each eigenvector, so that it is possible to display these eigenvectors as a sort of ghostly face image which is called an "eigenface".

Sample face images, the average face of them, eigenfaces of the face images and the corresponding eigenvalues are shown in Figure 1, 2, 3, 4 respectively. Each eigenface deviates from uniform gray where some facial feature differs among the set of training faces. Eigenfaces can be viewed as a sort of map of the variations between faces.

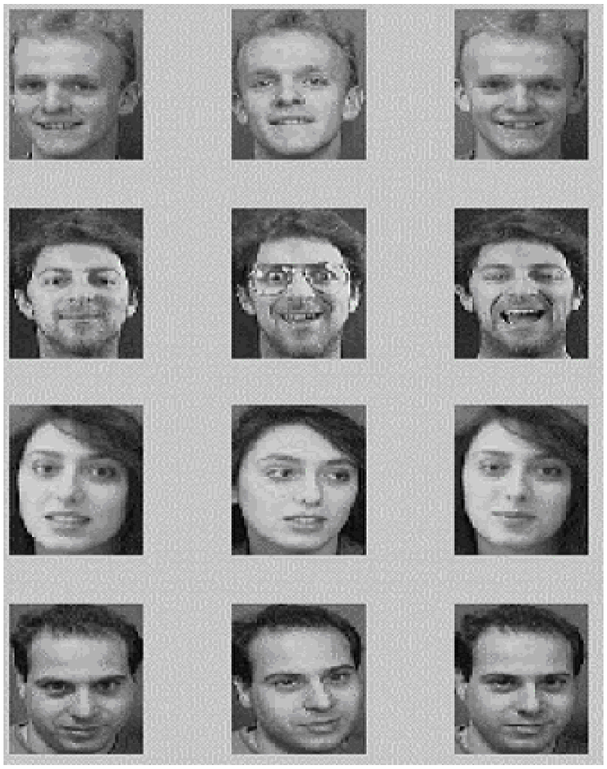

Fig 1 Sample Faces

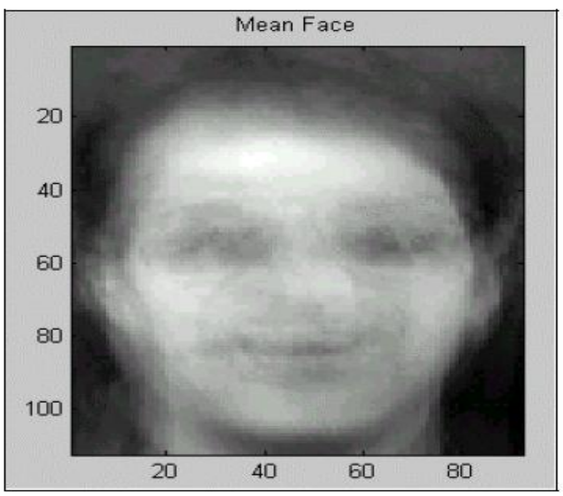

Fig. 2 Average face of the Sample face

Each individual face can be represented exactly in terms of a linear combination of the eigenfaces. Each face can also be approximated using only the "best" eigenfaces, those having the largest eigenvalues, and which therefore account for the most variance within the set of face images. As seen from the Figure 3.4, the eigenvalues drops very quickly, that means one can represent the faces with relatively small number of eigenfaces. The best M eigenfaces span an M-dimensional subspace which we call the "face space" of all possible images.

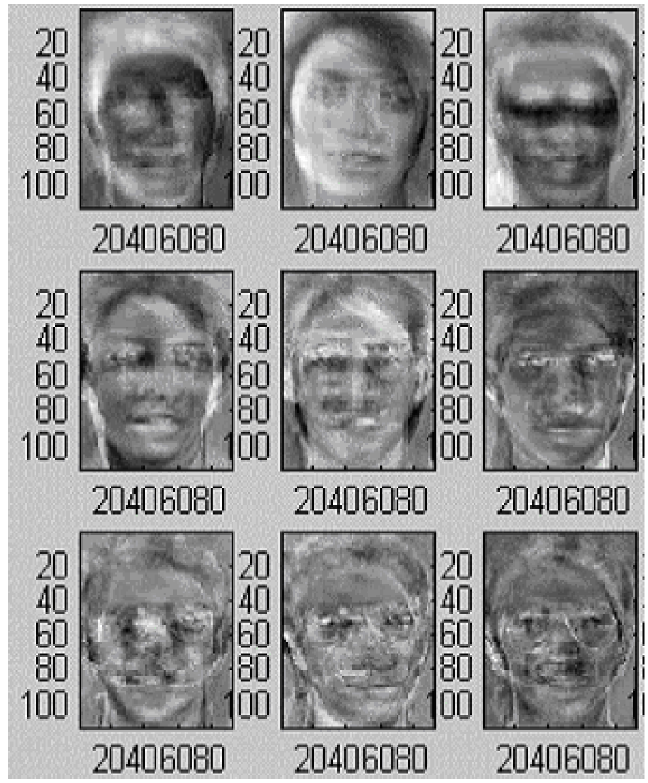

Fig. 3 Eigen face of the Sample face

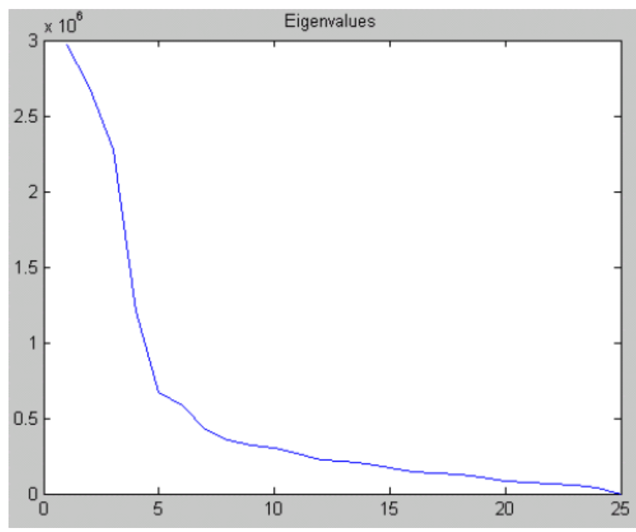

Fig. 4 Eigen values corresponding to Eigenfaces

Kirby and Sirovich[3,25] developed a technique for efficiently representing pictures of faces using principal component analysis. Starting with an ensemble of original face images, they calculated a best coordinate system for image compression, where each coordinate is actually an image that they termed as "eigen picture". They argued that, at least in principle, any collection of face images can be approximately reconstructed by storing a small collection of weights for each face, and a small set of standard pictures (the eigen pictures).

Fig 5 Reconstruction of first image with the number of Eigen Faces

The weights describing each face are found by projecting the face image onto each eigen picture.

Turk and A. Pentland[12] argued that, if a multitude of face images can be reconstructed by weighted sum of a small collection of characteristic features or eigen pictures, perhaps an efficient way to learn and recognize faces would be to build up the characteristic features by experience over time and recognize particular faces by comparing the feature weights needed to approximately reconstruct them with the weights associated with known individuals. Therefore, each individual is characterized by a small set of feature or eigenpicture weights needed to describe and reconstruct them. This is an extremely compact representation when compared with the images themselves. The projected image of the Face1 with a number of eigenvalues is shown in Figure 5. As seen from the figure among the 25 faces database (25 eigenfaces), 15 eigenfaces are enough for reconstruct the faces accurately. These feature or eigenpicture weights are called feature vectors.

If a multitude of face images can be reconstructed by weighted sums of a small collection of characteristic features or eigenspaces, an efficient way to learn and recognize faces would be to build up the characteristic features by experience over time and recognize particular faces by comparing the feature weights needed to reconstruct them with the weights associated with known individuals. Each individual, therefore, would be characterized by the small set of feature or eigenspace weights needed to describe and reconstruct them – an extremely compact representation when compared with the images themselves.

This approach to face recognition involves the following initialization operations:

(i) Acquire an initial set of face images (training set).
(ii) Calculate the eigenfaces from the training set, keeping only the M images that correspond to the highest eigenvalues. These M images define the face space. As new faces are experienced, the eigenvalues can be updated or recalculated.

These operations can also be performed from time to time whenever there is free excess computational capacity. Having initialized the system, the following steps are then used to recognize new face images.

(i) Calculate a set of weights based on the input image and the M eigenfaces by projecting the input image onto each of the eigenfaces.
(ii) Determine if the image is a face at all by checking to see if the image is sufficiently close to “face image”.
(iii) If it is a face, classify the weight pattern as either a known person or unknown.

B. Face Space Visited

Calculate the set of all possible images, those representing a face, make up only a small fraction of it. It has been decided to represent images as very long vectors, instead of the usual matrix representation. This makes up the entire image space where each image is a point. Since faces however possess similar structure (eye, nose, mouth, position) the vectors representing them will be correlated. We will see that faces will group at a certain location in the image space. We might say that faces lie in a small and separate region from other images. The idea behind eigen images is to find a lower dimensional space in which shorter vectors well describe face images, and only those.

In order to efficiently describe this cluster of images, we have chosen the set of directions in the image space along with the variance of the cluster is maximum. This is achieved through the standard procedure of Principal Component Analysis. A direction defined in terms of the coordinates of its extremity is in the image space actually an image. Transforming coordinates amounts to projection onto new coordinates and expressing an image as a linear combination of base images. The identified directions, thus, are images or more precisely eigen-images, and in this case it will be called them eigenfaces because faces are being decribed.

Let a face image l(x,y) be a two-dimensional N x N array of 8-bit intensity values. An image may also be considered as a vector of dimension N^2, so that a typical image of size 256X256 becomes a vector of dimension 65,536, or equivalently a point in 65,536-dimensional space. An ensemble of images, then, maps to a collection of points in this huge space.

Images of faces, being similar in overall configuration, will not be randomly distributed in this huge image space and thus can be described by a relatively low dimensional subspace. The main idea of the principal component analysis is to find the vectors that best account for the distribution of face images within the entire image space.

These vectors define the subspace of face images, which will be called as "face space". Each vector is of length N^2 describes an NXN image, and is a linear combination of the original face images. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and they are face-like in appearance, it will be refer to them as "eigenfaces".

Recognizing similar faces, is equivalent to identifying the closest point to the query in the newly defined face-space. If a person is represented in the database more than once, the problem is to decide to which group of images the query is most similar to. Finally, if the input image is not a face at all, it’s projection into the face space will give inconsistent results, so this case will also be identified.

IV. ROLE OF NEURAL NETWORKS

Neural networks[10,32,42] are composed of simple elements operating in parallel. The neuron model shown in figure 6 is the one which is widely used in artificial neural networks with some minor modifications on it.

The artificial neuron given in this figure has N input, denoted as $u_1, u_2, \ldots u_N$. Each line connecting these inputs to the neuron is assigned a weight, which is denoted as $w_1, w_2, \ldots, w_N$ respectively. The threshold in artificial neuron is usually represented by Φ and the activation is given by the formula:

$$a = (\sum_{j=1}^{n} w_j u_j) + \Phi$$

The inputs and weight are real values. A negative value for a weight indicates an inhibitory connection while a positive value indicating excitatory one. If Φ is positive, it is usually referred as bias. For its mathematical convenience (+) sign is used in the activation formula. Sometimes, the threshold is combined for simplicity into the summation part by assuming an imaginary input u_0 = +1 and a connection weight w0 = Φ. Hence the activation formula becomes

$$a = (\sum_{j=1}^{n} w_j u_j)$$

The vector notation

$$A = w^T u + \Phi$$

is useful for expressing the activation for a neuron. The neuron output function f(a) can be represented as:

Linear:

$$f(a) = K(a)$$

Threshold:

$$f(a) = \begin{cases} 0 & a \leq 0 \\ 1 & 0 < a \end{cases}$$

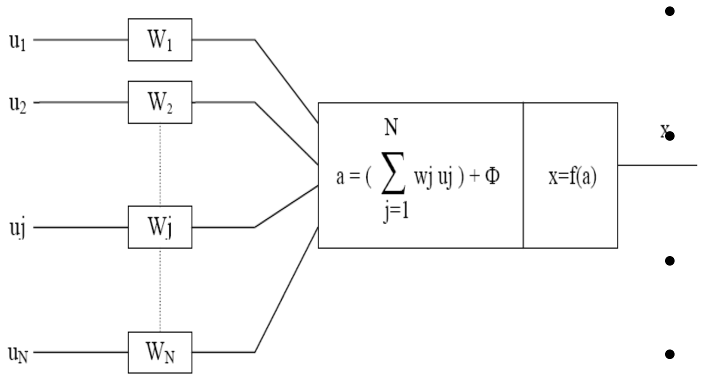

Fig. 6 Artificial Neuron

Ramp:

$$f(a) = \begin{cases} 0 & a \leq 0 \\ a / K & 0 < a \\ 1 & K < a \end{cases}$$

Sigmoid:

$$f(a) = 1/(1 + \exp(-Ka))$$

Functional implementations can be perfprmed by adjusting the weights and the threshold of the neuron. Furthermore, by connecting the outputs of some neurons as inputs to the others, neural network will be established, and any function can be implemented by these networks. The last layer of neurons is called the output layer and the layers between the input and output layer are called the hidden layers. The input layer is made up of special input neurons, transmitting only the applied external input to their outputs. In a network, if there is only the layer of input nodes and a single layer of neurons constituting the output layer then they are called single layer network. If there are one or more hidden layers, such networks are called multilayer networks. There are two types of network architecture: recurrent and feed forward neural network.

The eigenfaces approach to face recognition can be summarized in the following steps:

- Form a face library that consists of the face images of known individuals.
- Choose a training set that includes a number of images (M) for each person with some variation in expression and in the lighting.
- Calculate the M x M matrix L, find its eigenvectors and eigenvalues, and choose the M' eigenvectors with the highest associated eigenvalues.
- Combine the normalized training set of images according to produce M' eigenfaces. Store these eigenfaces for later use.
- For each member in the face library, compute and store a feature vector.
- Create Neural Network for each person in the database
- Train these networks as the faces will be used as positive examples for their own networks and negative examples for all other networks
- For each new face image to be identified, calculate its feature vector
- Use these feature vectors as network inputs and simulate all networks with these inputs
- Select the network with the maximum output. If the output of the selected network passes a predefined threshold, it will be reported as the host of the input face. Otherwise it will be reported as unknown and adds this

member to the face library with its feature vector and network.

V. Methodology Proposed

A. Distinct Block Processing

In this section, a basic yet novel method of face recognition is presented. The technique integrates the concepts of distinct block processing and relative standard deviations. Faces can be encoded in a block wise fashion wherein the standard deviation for each block is encoded.

In the proposed method, each image is segmented into distinct blocks and the standard deviation of each of these is calculated. The difference in standard deviations of two images gives a measure for comparison. The system has been tested using varying facial conditions: controlled condition, varying facial expression, varying pose and varying illumination.

Certain image processing operations involve processing an image in sections called blocks, rather than processing the entire image at once. In a distinct block operation, the input image is processed a block at a time. That is, the image is divided into rectangular blocks, and some operation is performed on each block individually to determine the values of the pixels in the corresponding block of the output image.

Distinct Blocks are rectangular partitions that divide a matrix into m-by-n sections. Distinct blocks overlay the image matrix starting in the upper left corner, with no overlap. If the blocks don't fit exactly over the image, zero padding is added so that they do as needed.

In order to determine the optimal block size for block processing, a simple algorithm is used. Suppose the syntax of the function that runs the algorithm is given as below:

Size = bestblk ([m.n],k)

This function returns for an m-by-n image, the optimal block size for block processing, 'k' is a scalar specifying the maximum row and column dimensions for the block: if the argument is omitted, it defaults to 100. The return value 'size' is a 1-by-2 vector containing the row and column dimensions for the block.

The algorithm for determining 'size' is:

- If m is less than or equal to k, return m.
- If m is greater than k, consider all values between min(m/10,k/2) and k
- Return that value that minimizes the padding required

The same algorithm is then repeated for n, for example

Size = bestblk([640,800],72)

B. Proposed Technique

Proposed Face Recognition system consists of a face database that may store multiple face images of the same person. First of all, these images are not actual face images.

The face image before being stored into the database is first resized into a standard size. It has been taken this as 112 X 92. Then the optimal block size is found for this image size; it comes out as 8 X 5. Then for every block, its standard deviation is calculated as follows:

$$S = \sqrt{\sum (X_i - X)^2 / (n-1)}$$

Where X_i is the individual pixel value, X is the mean of all pixel values in that block and n is the number of pixels in that block.

After the standard deviation for the block is calculated, all the pixels in that block are assigned that value. This is done for all the blocks.

Then these values are stored as 112 X 92 matrix in the database. In case an additional image of the same person has to be stored, it is first converted into the above matrix form and its mean will be taken with the matrix stored in the database corresponding to that person. Then this new matrix is overwritten in the database.

Now to compare the input image, it is also processed in the above form. Then this image matrix is compared with every image matrix in the face database. The comparison is carried out as follows:

For a block in the test image, choose any pixel value. Take a pixel value from the corresponding block of the image matrix in the database, find their absolute differences. Repeat this for all the blocks.

Sum all the absolute differences obtained. This gives a measure to compare the difference between the two images. Perform this calculation for each image stored in the database. The image for which this difference is the least is the one that matches best to the input test image.

C. Training and Test Image Constraints

The only constraint that this technique has is that for all images of the same person, the size of the face in the frame and its relative position in the frame must be more or less same.

VI. Experimental Result And Analysis

This method has been tested on 10 pictures from an ORL Database. This database has one face images with different conditions (expression, illumination, etc.), of each individual. In the following section, detailed information for this database and their corresponding performance results for the proposed face recognition method are given.

The number of networks used for the ORL database are equal to the number of people in the database. The initial parameters of the neural networks used in these tests are given below:

- Type: Feed forward bacpropagation network
- Number of layers: 3 (input, one hidden, output layer)
    - Number of neurons in input layer : Number of eigen faces to describe the faces
    - Number of neurons in hidden layer : 10
    - Number of neurons in output layer : 1

A. Test Results for the Olivetti and Oracle Research Laboratory (ORL) Face Database

The Olivetti and Oracle Research Laboratory (ORL) face database is used in order to test this method in the presence of headpose variations. There are 10 different images of each of 40 distinct subjects.

For some subjects, the images were taken at different times, varying lighting, facial expressions (open / closed eyes, smiling / not smiling), facial details (glasses / no glasses) and head pose (tilting and rotation up to 20 degrees). All the images were taken against a dark homogeneous background. Figure 7 shows the whole set of 40 individuals, 10 images per person from the ORL database.

Since the number of networks is equal to the number of people in the database, forty networks, one for each person were created. Within the ten images, first 4 of them are used for tanning the neural networks, then these networks are tested and their properties are updated with the next 3 pictures for getting the minimum squared error function, and these networks will be used for later use for recognition purposes.

For testing the whole database, the faces used in training, testing and recognition are changed and the recognition performance is given for whole database. For this database, the mean face of the whole database, the calculated top 30 (with the highest eigenvalues) eigenfaces, and their corresponding eigenvalues are shown in Figure 8, Figure 9 and Figure 10 respectively. The recognition performance increases with the number of faces used to train the neural networks, so the recognition rates, with different number of faces used in training, are calculated and given in Table 1.

The number of eigenfaces that is used to describe the faces, and the number of neurons in hidden layer also affects the recognition rate, so the tests are done with different number of eigenfaces and neurons, and the results are given in Table 2.

Histogram equalization enhances the contrast of images by transforming the values in an intensity image so that the histogram of the output image approximately matches a specified histogram. Table 2 also shows the recognition rates with and without histogram equalization. Up to now, the recognition rates are calculated with neural networks only for first choice (largest output), The recognition rates with neural networks other choices are also given in Table 3

B. Computational Time

The images in the database were tested for 10 pictures that were trained with the help of Neural Network and a very small amount of time(<3 seconds) is required to compute the result.

X. Conclusion And Future Work

The Eigenfaces algorithm has been successfully implemented which is fast popular and practical, besides familiarizing the user with the basics of image processing and the various other approaches to the face recognition.

The Eigenface approach to face recognition was motivated by information theory leading to the idea of basing face recognition on a small set of image features that best approximate the set of known face images without requiring that they correspond to our intuitive notions of facial parts and features. Although it is not an elegant solution to the general recognition problem, the Eigenface approach does provide a practical solution that is well fitted to the problem of face recognition. It is fast, relatively simple and has been shown to work well in a constrained environment.

It important to note that many applications of face recognition do not require perfect identification although most require a low false-positive rate. For instance, in searching a large database of faces, it may be preferable to find a small set of likely matches to present to the user.

Subsequently a novel algorithm have been developed which integrates the concepts of “Distinct Block Processing” and Relative Standard Deviation. The work is still in progress and may be enhanced further.

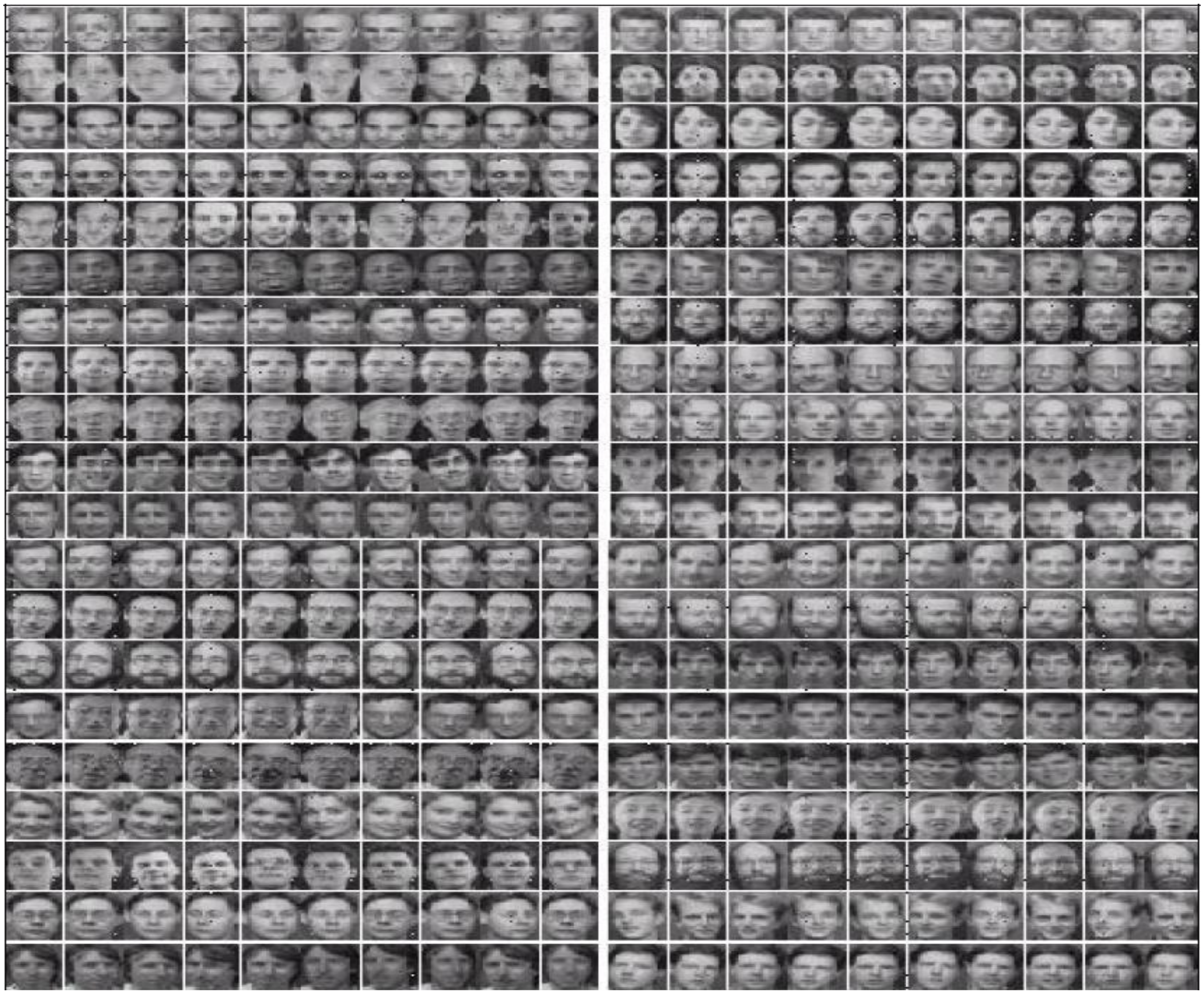

Figure 7 ORL Face Database

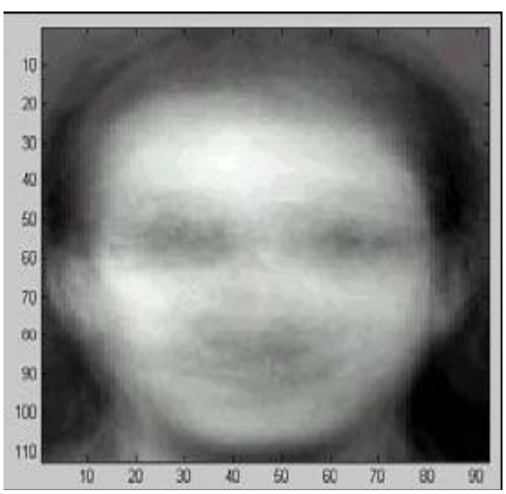

Figure 8 Mean face for ORL Face Database

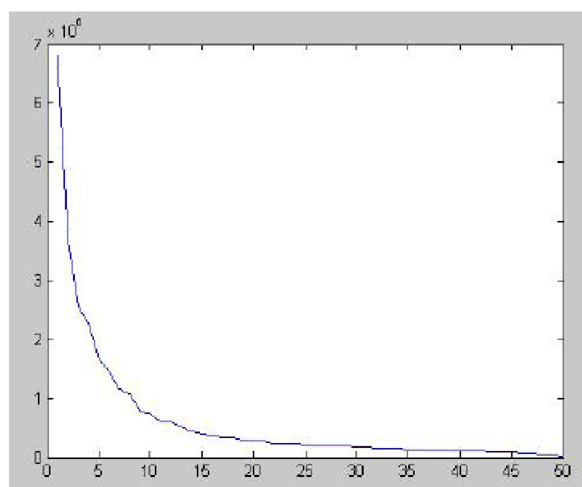

Figure 9 The eigen values for ORL Face Database

Figure 5.4 The top 30 eigen faces for the ORL Face Database

Number of Images used in Training (per individual)	*Number of Images used in Testing (per individual)*	*No of Eigen-faces*	*Hist. Equ.*	*Recogni-tion Rate (%)*
1	9	10	√	8.8
1	9	10	X	9.7
2	8	20	√	25
2	8	20	X	26
3	7	25	√	51.7
3	7	25	X	51.7
4	6	30	√	75
4	6	30	X	76.2
5	5	50	√	87.5
5	5	50	X	90
6	4	60	√	90
6	4	60	X	92.5
7	3	100	√	89.1
7	3	100	X	91.6
8	2	100	√	88.8
8	2	100	X	91.2
9	1	100	√	87.5
9	1	100	X	90

Table 1 Recognition Rate using different Number of Training and Test Images, and w/wo Histogram Equalization. Number of Neurons in Hidden Layer: 15

Number of EigenFaces	*Number of Neurons in Hidden Layer*	**RECOGNITION** *Rate (%)*
40	5	40
	10	58.3
	15	72
	20	74.5
45	5	41
	10	73.5
	15	80.8
	20	88.5
50	5	54.8
	10	73
	15	87.5
	20	90.8
55	5	50
	10	81.8
	15	88.5
	20	91.5
60	5	48.8
	10	81.3
	15	89.5
	20	92.8
65	5	52
	10	81.8
	15	91.5
	20	94.3
70	5	50
	10	86.3
	15	92.3
	20	93.3

Table 2 Recognition Rate using different Number of Eigen Faces and Neuron in hidden layer.
Number of Images used in Training : 5
Number of Images used in Testing : 5
Histogram Equalization : Done for all images

	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Number of Correct Decision (Over 400 Images)	360	21	7	4	2
Recognition Rate	90%	95.2 %	97%	98%	98.5 %

Table 3 Recognition Rate with Neural Networks different choices.

Number of Testing Images: 5

Number of Eigenfaces: 50

Number of H. L. Neurons: 15

REFERENCES

[1] Goldstein, A. J., Harmon, L. D., and Lesk, A. B., Identification of human faces", Proc. IEEE 59, pp. 748-760, (1971).

[2] Haig, N. K., "How faces differ - a new comparative technique", Perception 14, pp. 601-615, (1985).

[3] Kirby, M., and Sirovich, L., "Application of the Karhunen-Loeve procedure for the characterization of human faces", IEEE PAMI, Vol. 12, pp. 103-108, (1990).

[4] Terzopoulos, D., and Waters, K., "Analysis of facial images using physical and anatomical models", Proc. 3rd Int. Conf. on Computer Vision, pp. 727-732, [1] F.Galton, "Personal identification and description 1,1 Nature, pp.173-177,21

[5] Manjunath, B. S., Chellappa, R., and Malsburg, C., "A feature based approach to face recognition", Trans. of IEEE, pp. 373-378, (1992).

[6] Harmon, L. D., and Hunt, W. F., "Automatic recognition of human face profiles", Computer Graphics and Image Processing, Vol. 6, pp. 135-156, (1977).

[7] Harmon, L. D., Khan, M. K., Lasch, R., and Ramig, P. F., "Machine identification of human faces", Pattern Recognition, Vol. 13(2), pp. 97-110,(1981).

[8] Kaufman, G. J., and Breeding, K. J, "The automatic recognition of human faces from profile silhouettes", IEEE Trans. Syst. Man Cybern., Vol. 6, pp. 113-120, (1976).

[9] Wu, C. J., and Huang, J. S., "Human face profile recognition by computer", Pattern Recognition, Vol. 23(3/4), pp. 255-259, (1990).

[10] Kerin, M. A., and Stonham, T. J., "Face recognition using a digital neural network with self-organizing capabilities", Proc. 10th Int. Conf. On Pattern Recognition, pp.738-741, (1990).

[11] Nakamura, O., Mathur, S., and Minami, T., "Identification of human faces based on isodensity maps", Pattern Recognition, Vol. 24(3), pp. 263-272, (1991).

[12] Turk, M., and Pentland, A., "Eigenfaces for recognition", Journal of Cognitive Neuroscience, Vol. 3, pp. 71-86, (1991).

[13] Yuille, A. L., Cohen, D. S., and Hallinan, P. W., "Feature extraction from faces using deformable templates", Proc. of CVPR, (1989).

[14] P.Philips, “The FERET database and evaluation procedure for face recognition algorithms, “ Image and Vision Computing, vol.16, no.5, pp.295-306, 1998
[15] Carey, S., and Diamond, R., "From piecemeal to configurational representation of faces", Science 195, pp. 312-313, (1977).
[16] Bledsoe, W. W., "The model method in facial recognition", Panoramic Research Inc. Palo Alto, CA, Rep. PRI:15, (August 1966).
[17] Bledsoe, W. W., "Man-machine facial recognition", Panoramic Research Inc. Palo Alto, CA, Rep. PRI:22, (August 1966).
[18] Yuille, A. L., Cohen, D. S., and Hallinan, P. W., "Feature extraction from faces using deformable templates", Proc. of CVPR, (1989).
[19] Fleming, M., and Cottrell, G., "Categorization of faces using unsupervised feature extraction", Proc. of IJCNN, Vol. 90(2), (1990).
[20] Kanade, T., "Picture processing system by computer complex and recognition of human faces", Dept. of Information Science, Kyoto University, (1973).
[21] Ching, C. W., and Huang, L. C., "Human face recognition from a single front view", Int. J. of Pattern Recognition and Artificial Intelligence, Vol. 6(4), pp. 570-593, (1992).
[22] E. DEDE, “Face Recognition Using Geometric Features and Template Matching By Dimension Reduction”, MSc Thesis, METU, September 2003
[23] D. Pissarenko (2003). Eigenface-based facial recognition.
[24] P. Belhumeur, J. Hespanha, and D. Kriegman (july 1997). "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection". IEEE Transactions on pattern analysis and machine intelligence 19 (7).
[25] L. Sirovich and M. Kirby (1987). "Low-dimensional procedure for the characterization of human faces". Journal of the Optical Society of America A 4: 519–524.
[26] Burt, P., "Smart sensing within a Pyramid Vision Machine", Proc. of IEEE, Vol. 76(8), pp. 139-153, (1988).
[27] Characterization of human faces". IEEE Transactions on Pattern analysis and Machine Intelligence 12 (1): 103–108.
[28] Rein-Lien Hsu, “Face Detection and Modeling for Recognition,” PhD thesis, Department of Computer Science & Engineering, Michigan State University, USA, 2002.
[29] Henry A.Rowley ,’’Neural Network-based face detection’’ PhD thesis, Carnegie Mellon University, Pittsburgh, USA, May 1999.
[30] Kohonen, T., "Self-organization and associative memory", Berlin: Springer- Verlag, (1989).
[31] Kohonen, T., and Lehtio, P., "Storage and processing of information in distributed associative memory systems", (1981).
[32] Howard Demuth, Mark Beale, “Neural Network Toolbox User’s Guide For Use with MATLAB”, by The MathWorks, Inc.1998.
[33] John Daugman, ‘’Face and Gesture Recognition: Overview’’ IEEE PAMI, vol.19, no.7, July 1997.
[34] M.-H. Yang. N. Ahuja, and D. Kriegman, “Face recognition using Kernel Eigenfaces.” *Advances in NIPS,*Vol. 14, 2002.
[35] Q. Liu and S. Ma. “Face Recognition Using Kernel Based Fisher Discriminant Analysis.” *IEEE Conf. on Automatic Face and Gesture Recognition*, 2002.
[36] M.-H. Yang. “Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods” *IEEE Conf. on Automatic Face and Gesture Recognition*, 2002.
[37] S. Gutta, V. Philomin and M. Trajkovic. “An Investigation into the use of Partial-Faces for Face Recognition” *IEEE Conf. on Automatic Face and Gesture Recognition*, 2002.
[38] B. Scholkopf, “Statistical Learning Kernel Methods”. *NIPS'00*, 2000.
[39] R. Zhang and A. I. Rudnicky. “A large Scale Clustering Scheme for Kernel K-Means” *Proc. of ICPR,*
[40] Fischler, M. A., and Elschlager, R. A., "The representation and matching of pictorial structures", IEEE Trans. on Computers, c-22.1, (1973).
[41] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, “The FERET database and evaluation procedure for face recognition algorithms,” *Image and Vision Computing J*, Vol. 16, No. 5. pp 295-306, 1998.
[42] C..M.Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.
[43] W. Zhao, R. Chellappa, and A. Rosenfeld, “Face recognition: a literature survey”. *ACM Computing Surveys*, Vol. 35:pp. 399–458, December 2003.
[44] V. Bruce, P.J.B. Hancock and A.M. Burton, “Comparisons between human and computer recognition of faces”, *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, Vol., Iss., 14-16 Pages:408-413, Apr 1998