International Journal on Emerging Technologies 10(4): 102-110(2019)

ISSN No. (Print): 0975-8364
ISSN No. (Online): 2249-3255

A Comparative Study on Hopfield Network with LBP, PCA and LDA for Face Recognition in Distorted Face Images

Ramesh Chandra Sahoo[1] and Sateesh Kumar Pradhan[2]
[1]Research Scholar, Computer Science & Applications, Utkal University, Bhubaneswar, (Odisha), India.
[2]Professor, Computer Science & Applications, Utkal University, Bhubaneswar, (Odisha), India.

(Corresponding author: Ramesh Chandra Sahoo)
(Received 26 August 2019, Revised 30 October 2019, Accepted 10 November 2019)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: This paper presents a comparative analysis on performance in terms of recognition rate of distorted face images with various feature extraction methods individually and jointly. Several research efforts have been put into the field of face recognition in recent years; however various issues in this still remain open. Here we have attempted Local Binary Pattern Histogram (LBPH), Principal Component Analysis (PCA), LBP with PCA, PCA with LDA (Linear Discriminant Analysis) to compare the rate of accuracy of face recognition with different percentage of distorted face images. For this comparison we introduced 10%, 20%, 30%, 40% and 50% noise in the test images and measured their accuracy of recognition by using KNN for each method. It has been observed that face recognition is more challenging when images are gets distorted, for which we focus on a recurrent auto associative memory model. In this paper we have been implemented an auto-associative memory model called Hopfield network for the same purpose and compare the recognition rate with above mentioned feature extraction methods. In Hopfield model, the classification is carried out through nearest Hamming distance. In this study, experiments have been carried out on two different datasets namely ORL database and FACE94 database. It has been observed that Hopfield model provides better result of recognition as noise percentage is increases.

Keywords: Face recognition, LBP, PCA, LDA, ORL & FACE94 databases, and Hopfield Network.

Abbreviations: *LBP*, local binary pattern; PCA, principal component analysis; LDA, linear discriminant analysis.

I. INTRODUCTION

In recent days of image processing and computer vision, face recognition has become a very popular and emerging area of research. A number of feature based face recognition systems such as Principal Component Analysis (PCA) [1-3], Linear Discriminant Analysis (LDA) [4, 5], Local Binary Pattern (LBP) [6-9], Independent Component Analysis (ICA) [10], Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Histogram of Oriented Gradient (HOG), and many more have been implemented till date. In this paper we studied LBPH, PCA and LDA and combination of these methods for our experimental work with distorted face images and we also implemented Hopfield neural network for storing and recalling of these noisy face images. Most of the work done so far based on two features of facial images, one is geometric features which deal with shape and location deviation, distance between eyes and nose length and other one is appearance features that deal with wrinkles and furrows as appearance variations of the face image. PCA based face recognition system is a statistical method for appearance features that reduces larger dimensionality of data to a smaller valuable intrinsic dimensionality called eigenface space. LDA is a linear projection based feature extraction method where the most discriminant projection vectors that reduce dimensionality of feature space and projected sample form the maximum between class and minimum within class feature space. LBP was first used for texture classification but now-a-days it has been also used for recognition in image processing. LBP basically describes local texture pattern by thresholding the neighborhoods pixel intensity values that produces a binary code. Recognizing noisy face image is a major drawback in above mentioned feature extraction methods. Our focus in this study is recognizing noisy images through an auto associative memory model named Hopfield model. As "it is a content addressable memory network, that will reproduce the original pattern as output even an input pattern is incomplete or corrupt".

In this paper, we implemented LBP Histogram, PCA, LBP with PCA and PCA with LDA based recognition for ORL face database of 400 samples and FACE94 database with 2000 samples. The result and percentage of recognition are illustrated in the experimental analysis section. Our work again demonstrate for storage and recall of these face images with a memory model named Hopfield neural network model by considering the reduced feature vectors of above mentioned feature extraction methods. We introduced 10%, 20%, 30%, 40% and 50% external noise and test the recall efficiency of the Hopfield model.

II. RELATED WORK

In last few decades, several works have been done for face recognition by extracting most useful features from face images. Some of these feature extraction methods like Local Binary Pattern (LBP), Principal Component

Analysis (PCA), and Linear Discriminant Analysis (LDA) are very popular methods in face recognition. Ojala *et al.,* [11], was introduced LBP as a texture descriptor for texture feature extraction. Several studies [17-18] shows a better face recognition result in terms of speed and discrimination performance through LBP method. Meena and Suruliandi, [18], discussed the performance of LBP and modified MLBP (multivariate local binary pattern), CS-LBP and LBPV by showing that the center symmetric LBP had better result for face recognition. Huang. X. *et al.,* [19], proposed an extended local binary pattern version (ELBP) to retain the spatial information of face image and to overcome the limited local appearance of face image. Zhang, *et. al.* [20] proposed a method to describe the local features of face images by using LBP histogram (LBPH). A two class transformation has been carried out from a multiclass as intra-personal or extra-personal class. AdaBoost method was used to achieve 97.9% of accuracy. Li & Chen [21], proposed a combined model for face recognition with PCA and SVM where PCA reduced the dimension of face images and SVM used to separate the classes. Yang, *et al.,* [3], presented two dimensional PCA (2DPCA) that out-performs over PCA for small sample size problems. Chang *et al.,* [22] presented an ear and face recognition using PCA and also shown a multimodal result by combining both. Belhumeur *et al.,* [5], presented Fisher's Linear Discriminant method (LDA), for face recognition providing better result as compare to PCA even in variation in lighting conditions. Zhao *et al.,* [23], proposed a method by combining PCA and LDA for face recognition to improve the generalization capability of LDA for smaller dataset. Soni *et al.,* [24], proposed a method for face recognition by cloud Hopfield neural network with Hebbian learning rule. Dai & Nakano [25], proposed a Hopfield memory model for face recalling method for stored patterns with a pattern matching method.

III. FEATURE EXTRACTION

In digital image processing feature extraction methods are used to reduce redundancy and irrelevancy information from the input image by which one can reduce the training time and space. These algorithms provide most effective information from the original image that is useful for classification, identification and recognition. Feature extraction algorithms have great importance in machine learning, pattern recognition, computer vision, medical image diagnosis, multimedia information retrieval, data mining and many more.

A. Local Binary Pattern

Local Binary Pattern (LBP) is a texture descriptor that outputs a binary number. LBP operator basically assigns a label to each pixel of an image by thresholding the 3 × 3 neighborhood of each pixel with center pixel value. This process is done by dividing an image into several small blocks and extracted features from each block by comparing its surrounding pixels. It was originally introduced by Ojala *et. al.,* [11] with 3 × 3 sub region of 8 neighbourhoods. In a sub region the value of center pixel is the threshold and all 8 neighbourhoods pixel values are compared with the center pixel value, if the neighbourhoods gray scale pixel value is higher than the threshold center gray scale value then one is assigned to that pixel otherwise zero is assigned to the corresponding pixel. The equation how LBP operator works is given as:

$$P_i = \begin{cases} 1 & \text{if } G_i \geq G_c \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

where P_i is the binary value that will assigned to neighbouring pixel $i \in \{1,2,\ldots,8\}$, G_i is the gray level value of i^{th} neighbouring pixel and G_c is gray level value of center pixel of 3-by-3 sub region. Then the resulting binary values of that sub region are concatenated into a 8 bit binary number and its decimal value is used to create the feature vector.

Later on the LBP operator was enhanced (ELBP) for neighbourhoods of different sizes in a circular shape with radius R. In this ELBP S sampling neighborhood points depending upon the radius R are interpolated and the Fig. 2 shows the neighbourhoods pixels on the edge of the circle with radius R.

Then the extended LBP descriptor for every pixels with a radius R is calculated as:

1. For each pixel with coordinate (x, y), choose S neighbouring pixels of radius (R).
2. Measure the intensity difference of the current pixel with the S neighbouring pixels
3. Assign 0 or 1 to form bit vector by thresholding the intensity difference in such a way that all negative numbers are set to zero and all positive numbers are set to one as:

$$S(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (2)$$

4. Replace the intensity value of pixel (x, y) with the decimal value which was calculated by converting the S-bit vector to its corresponding decimal value. Hence the LBP operator of each pixel as:

$$LBP_{S,R}(x,y) = \sum_{s=0}^{s-1} s(g_s - g_c)2^S \quad (3)$$

where g_c and g_s are the intensities of the current pixel c and its s^{th} neighbour. S is number of neighbouring pixel selected in radius R.

A sample LBP operator working principle and its circular neighbour points with various values of S and R are shown in Fig. 1 and 2.

B. Principal Component Analysis

Principal Component Analysis (PCA) is a linear transformation to projecting data into a smaller space from a larger space by reducing the correlation between them in a way that most important information of original data is retained. Basic steps of principal component analysis are explained below:

1. Firstly two dimensional face images of size N-by-N are transformed to one dimensional form as N^2-by-1. All the data set both training and testing are of same size. Let i^{th} face image of vector size containing N^2 pixels be represented as:

$$x^i = \{ x_1^i, x_2^i, \ldots\ldots\ldots x_{N^2}^i \} \quad (4)$$

2. Store all M images in a data matrix of size N^2-by-M and is represented as:

$$X = \{\Gamma_1, \Gamma_2, \ldots\ldots, \Gamma_M\} \quad (5)$$

where Γ_i is the i^{th} image of size N^2-by-1

3. Calculate the average matrix (mean of the data set) of of N^2-by-1

$$\mu = \frac{1}{M}\sum_{i=1}^{M} \Gamma_i \quad (6)$$

4. Subtract the mean from the original faces and store it in a variable called A of size N^2-by-M

$$A_i = \Gamma_i - \mu \quad (7)$$

5. Now obtained the covariance matrix C as:

$$C = \frac{1}{M}\sum_{i=1}^{M} A_i A_i^T = BB^T \quad (8)$$

Where BB^T is a N^2-by-N^2 matrix and

$$B^T = [A_1, A_2, \dots, A_M]^T$$

Obtained the eigenvalues λ_i and eigen vectors v_i of the covariance matrix C.

$$Cv_i = \lambda_i v_i \text{ for } i = 1,2 \dots M \quad (9)$$

Since BB^T is a very large size matrix we may find the eigenvector of B^TB matrix as

$B^TBv_i = \lambda_i v_i$ as BB^T and B^TB have the similar eigenvalues and are related with their corresponding eigenvectors.

6. Now project the samples into PCA subspace by arranging the eigenvectors in descending order by their eigenvalues. Choose the k-largest eigenvalues and store their corresponding eigenvectors in a variable W and k-principal components of the experimental vector x is then given by

$$y = W^T(x - \mu) \quad (10)$$

and then recognized by $x = Wy + \mu$ (11)

C. Linear Discriminant Analysis

Linear discriminant analysis is also used for dimensionality reduction that preserved the class discriminatory information for which feature vectors are most separable after transformation and its working principle is similar to principal component analysis. Steps of LDA are described in following part:

1. As in case of PCA, first collect the dataset X. Let M number of total number of face images are in the dataset of C classes and let assume there are N sample images of each class.

2. Obtained the average (mean) of the training dataset m and average face value of each class mc as:

$$m = \frac{1}{M}\sum_{i=1}^{M} X_i \quad (12)$$

$$m_c = \frac{1}{N}\sum_{j=1}^{N} x_j \text{ for each class} \quad (13)$$

3. Obtained between-class scatter matrix S_B and in-class scatter matrix S_W as:

$$S_B = \sum_{i=1}^{C} N_i \left(m_{C_i} - m\right)(m_{C_i} - m)^T \quad (14)$$

where C is the number of classes N_i is the number of samples in i^{th} class, m is the overall mean and $m_{C_i}$is the mean of i^{th} class.

$$S_W = \sum_{j=1}^{M} \left(X_j - m_{c_j}\right)(X_j - m_{c_j})^T \quad (15)$$

4. Obtained the eigenvalues and eigenvectors from the equation

$$W_F = \frac{S_B}{S_W} = S_B S_W^{-1} \quad (16)$$

and select k eigenvector wi with the largest eigenvalues to form a matrix as

$$W = [w_1, w_2, \dots, w_k] \quad (17)$$

5. Now project samples to a new subspace using W matrix and compute the coordinate Y=XW (18)

IV. HOPFIELD NEURAL NETWORK

Hopfield neural network is a well known recurrent network was invented by Hopfield & Tank (1986); Tank & Hopfield (1987) [12, 13], is effective when used as associative memory Hopfield (1982)[14]. It is a highly parallel content addressable memory, where retrieval is possible although the input is corrupted by noise. It consists of n fully connected neurons, where each neuron is responsible for one pixel from the input pattern and the output is the value of activation function from sum of factors weight and previous values for each neuron as given below:

$$y_i = f(\textstyle\sum_{j=1}^{n} w_{ij} * y_j(t)) \quad (19)$$

where w_{ij} is the weight applied to the output of node j that feeds to node i, y is the output function, f is the hard limit activation function.

The input to the neurons is from bipolar (+1 or –1) value, then f is symmetric hard limit function given as:

$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \quad (20)$$

and the weights are specified by:

$$w_{ij} = \sum_{s=0}^{M-1} x_i^s x_j^s \text{ for } i \neq j \text{ and } w_{ij} = 0 \text{ for } i = j \quad (21)$$

where x_i^j is the element i of the exemplar for pattern j.

and the general weight matrix is given by:

$$W = \begin{bmatrix} 0 \dots & w_{12} \dots & w_{1i} \dots . & w_{1N} \\ w_{21} \dots & 0 \dots & w_{2i} \dots . & w_{2N} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ w_{N1} \dots & w_{N2} \dots & w_{ni} \dots & 0 \end{bmatrix} \quad (22)$$

Then for retrieval of stored images:

1. A pattern P_i which is already stored in the network or a new pattern (with or without distortion) can be present to the Hopfield network at time t = 0 as:

$$y_i(t = 0) = P_i, \forall_i = \{1,2, \dots n\} \quad (23)$$

2. Then randomly update neuron i by using

$$y_i(t + 1) = f(\textstyle\sum_{i=1}^{n} W_{ji} y_i(t)), \forall_j = \{1,2, \dots n\} \quad (24)$$

where $f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$

3. Increment the time or iteration t=t+1

4. Now until all neurons are updated or network reached a stable state repeat step 3 and 4 which means

$$y_i(t + 1) = y_i(t), \forall_i \quad (25)$$

V. EXPERIMENTAL ANALYSIS

In this work, we evaluated and compared the recalling efficiency of face images from ORL face database and FACE94 database by using LBP, PCA, PCA+LDA and HOPNET using Matlab 2014a.

A. Experimental Setup

Dataset: In our experimental work, a compound dataset of 1640 face images of 102 individuals are used from two databases such as ORL dataset and FACE94 dataset.

The Olivetti Research Laboratory (ORL) dataset [15] contains 40 folders with names s1, s2... s40 for 40 distinct subject and in each folder (subject), 10 different images of a person of resolution 112 × 92 pixels were present where images were taken at varying time, different facial expression and varying lighting.

FACE94 is a face database, [16] of 153 individuals (male and female) with 20 images per individual of size 180 by 200 pixels. It contains 113 male, 20 female and 20 male staffs face images of various racial origins. The background is plain green with variation in their head turn, tilt and slant and considerable expression changes

Experiment: In our different experiments, we applied three different scenarios for training and testing dataset such as (i) 80% training and 20% testing, (ii) 70% training and 30% testing and (iii) 50% training and 50% testing and recognition rate is observed for all methods mentioned in this paper. In each of the experiment, we introduced noise in two ways and the average of both ways is being considered here in terms of percentage of recognition rate for these distorted images for every scenarios mentioned in above paragraph and for each dataset. One way is that these noises were introduced by a predefined Matlab function called imnoise with salt & pepper noise for 10%, 20%, 30%, 40% and 50% and measured the recognition rate. The second way is that we explicitly flip the number of bits from 0 to 1 and vice versa for 10% of bits, 20% of bits, 30% of bits, 40% of bits and 50% of bits.

B. Recognition through Local Binary Pattern Histogram

LBP has since been used for texture classification for various image identification methods. In this paper, we combined with extended LBP with histograms for improvement in face recognition. For this implementation, four parameters of LBPH are as:

Radius:- It is for circular LBP from the centre pixel (we consider R=1).

Neighbours: The number of sample pixels from the centre pixel (in our case it is 8).

X-Grid: The number of cells in the horizontal direction (here we fixed it to 8)

Y-Grid: The number of cells in vertical direction (here we fixed it to 8).

Training: For our training, we resizes the face images from both ORL and FACE94 datasets to 96X96 pixels and separately train both the databases for further processing. The ELBP (extended local binary pattern) operator works as follows:

If the center pixel (x_i, y_j), then the coordinates of N neighbours (x_n, y_n) with radius R can be calculated:

$$x_n = x_i + R\cos(\frac{2\pi n}{N}) \quad (26)$$

and

$$y_n = y_i + R\sin\left(\frac{2\pi n}{N}\right) \quad (27)$$

Where R is the radius of the circle and n is the number of sample points on the circle. It is also known as circular LBP or extended LBP. After getting a list of local binary patterns (as discussed in section-II of this paper), convert these binary numbers into a decimal number using binary to decimal conversion and then prepare histogram of those decimal values. Basically these histograms of patterns form a feature vector for the texture of the images. To obtained the similarity between images one can use these histograms by measuring the distance between them.

Then in recognition phase, the test image generate a histogram as explained above and then the best match by Euclidian distance returns by the label associated with that image for training data. A sample histogram image is shown Fig. 3 from our ORL dataset used for training.

The recognition accuracy from our work is shown in Table 1 with different errors in the testing dataset for both the databases. Fig. 4 and 5 demonstrates the bar graphs of these accuracy from ORL and FACE94 databases respectively.

C. Recognition through Principal Component Analysis

PCA approximating a high-dimensional dataset with a lower-dimensional subspace. In our experimental work we obtained eigenfaces from PCA by singular value decomposition (SVD). SVD can also be used as a method for data reduction along which data points reveal most deviation by identifying and ordering the dimensions of data. The singular value decomposition of a m-by-n data matrix D is as:

$$D = U W V^T \quad (28)$$

$$[D] = [U]\begin{pmatrix} w_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & w_n \end{pmatrix}[V]^T \quad (29)$$

Where, U is m-by-m and orthogonal, V is n-by-n and orthogonal and W is m-by-n and diagonal. The columns of U and V are called the singular vectors corresponding to the singular values in the diagonal matrix W. SVD can be used to finding the eigenvalues and eigenvectors of DD^T and D^TD. The eigenvectors of D^TD make up the columns of V , the eigenvectors of DD^T make up the columns of U. Also, the singular values in W are square roots of eigenvalues from DD^T or D^TD. Steps to finding the eigenfaces and projected faces through SVD is given as:

1. Get the data matrix D, which is a m-by-n matrix, where m is the number of feature dimensions and n is the number of face images. In our case we consider 60-by-60 face images of different sizes of dataset for training as we consider 80%, 70%, 50% images for training.
2. Calculate the mean as: $\mu = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i$, where M is the number of images for each database separately and Γ_i is the ith image.
3. Subtract the mean from original data to get the mean minus normalized dataset.
4. Obtained the singular value decomposition of the newly obtained data matrix obtained from step-2 to get U, W, V. where columns of V are principal directions and columns of UW are principal components.
5. Now to reduce the dimensionality of dataset from N to k < N, we select first k columns of U and k X k upper left part of W. their product U_kW_k is the required n X k containing first k PCs. That means the eigen faces are first k columns of U matrix. Let it be E=U(: , 1:k).
6. Then the projected training images can be obtained as:

   $P_I = E^T$ x (μ - *datamatrix*).
7. For recall an image from test dataset get an image.
8. Subtract the obtained in step to from the query image.
9. Now multiply transpose of eigenface matrix obtained in step 5 to the query image to get the projected testing image as $P_T = E^T$ x (μ - *testimage*).
10. Compare the projected test image with projected training dataset by Euclidian distance and display the minimum distance training image.

A sample image from our experimental work is shown in Fig.6 for nine eigenfaces of ORL database. Table 2 shows our recognition rate (RR) in terms of percentage of successful recall for both databases we used for our study with different error percentages from 10% to 50% error. Fig. 7 and 8 demonstrates their bar graph for ORL and FACE94 database respectively.

D. Recognition through PCA Plus LDA

Linear Discriminant Analysis (LDA) basically maximizing the component axes for class separation where as PCA component axes that maximizes the variance. In this study, we applied PCA plus LDA one after another for further reduction of feature space. It has been observed that even with smaller features (39 features for ORL-database as compared to 319 in PCA and 99 features for FACE94 database as compared to 1599 in PCA), we obtained the better result of recognition. Steps for performing LDA are:

1. Collect the training dataset for both databases (ORL and FACE94 database as per three scenarios mentioned above).
2. To get the eigenfaces, we first perform PCA on different datasets.
3. Project the training datasets using eigenfaces into PCA subspace to reduce the dimensionality of training datasets.
4. On these reduced training datasets, we perform LDA to obtained the class representative fisherfaces by calculating within-class (S_W) and between-class (S_B) scatter matrix for each class.

4.a. Obtained eigenvector and eigenvalue from S_W and S_B

4.b. Then obtained fisherfaces from the resulting eigenvectors and eigenvalues computed in step (a)

5. Project the training datasets by using fisherfaces to further reduce the features into LDA subspace
6. For testing an image, first subtract the mean, then project the testing image into LDA subspace and find the Euclidean distance and then show the minimum distance training image.

The result we obtained from both the databases with different error and three different scenarios are given in Table 3 and their bar graphs are shown in Fig. 9 and 10 for ORL and FACE94 database respectively.

E. Recognition through Hopfield Model

In proposed Hopfield model, we performed two types of experiments. In first we store all patterns and recall those by introducing noise of 10%, 20%, 30%, 40%, and 50% and in second case we separate the training and testing datasets in three different scenarios mentioned above and only stores training datasets and recognize for testing datasets by introducing noise of 10%, 20%, 30%, 40%, and 50% with the help of Hebbian learning rule. The procedure for storing and recalling are given as:

Storage Process: Storing the data matrix in Hopfield networks means preparing the weight matrix with Hebbian rule.

1. Firstly we resized each training images of size N X N (60 X 60 in our case), and then convert these to gray scaled images.
2. Apply histogram equalization by modifying the intensity distribution of the histogram to enhance contrast and then convert these images to binary images with a threshold.
3. Prepare the data matrix of size N^2 X M for training or to obtained the weight matrix, where N^2 pixels are in one image and there are M such images. And then convert data matrix to bipolar {+1 or -1}
4. Now obtained the weight matrix as discussed in section-II in this paper of Hopfield neural network part for $W_{ij} = W_{ji}$ and set $W_{ii} = 0$ as it has no self connection.
5. For recall one stored pattern or for recognizing a new pattern with or without noise:

5.a. Load the weight matrix obtained in step 4

5.b. Now present an unknown pattern (X) for testing to the network and retrieve a stored association, for which first initialize the network as

$$x_j(0) = x_j, j = 1,2, \ldots, n$$

Where $x_j(0)$ is the j^{th} element of vector X at iteration t=0.

5.c. Calculate the network output at t=0 as

$$y_i(0) = f(\sum_{j=1}^{n} w_{ij} x_j(0) - \theta_i) \text{ for } i = 1,2, \ldots n \quad (30)$$

5.d. Update the state as

$$y_i(t+1) = f(\sum_{j=1}^{n} w_{ij} x_j(t) - \theta_i) \quad (31)$$

5.e. Repeat iteration until convergence means when input and output remains unchanged or exceeds the maximum iterations.

6. In our work, we compare the testing pattern with all training pattern by hamming distance and the minimum hamming distance image from the stored images is displayed for the corresponding testing pattern.

Tables 4 demonstrates the recalling efficiency of Hopfield network with noise percentage of 10, 20, 30, 40, and 50. And Table 5 demonstrates the recognition accuracy with various data partition methods with different noise percentages. Fig. 1 and 2 are represents the bar graphs of recognition rate for ORL and FACE94 database respectively.

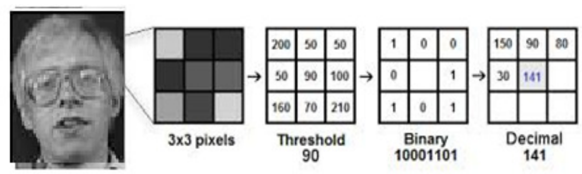
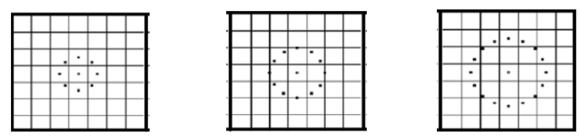


Fig. 1. LBP operator working principle.



(S=8, R=1.0) (S=12, R=1.5) (S=16, R=2.0)

Fig. 2. Circular neighborhood points with three different values of S and R.

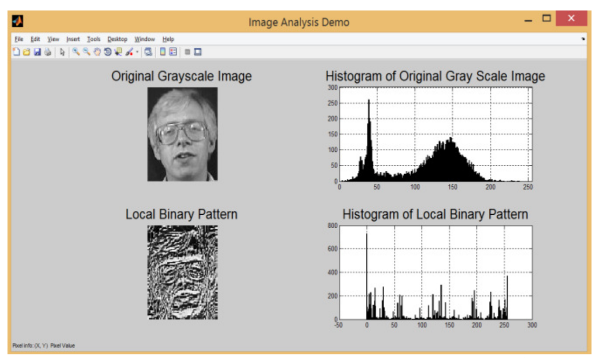


Fig. 3. A sample LBP Histogram Image from ORL database.

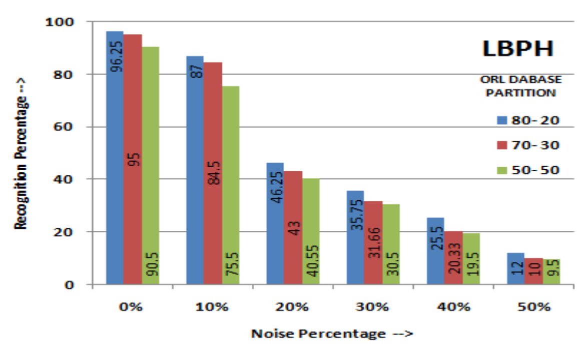


Fig. 4. Recognition Rate graph with different noise Percentages by LBPH method for ORL database.

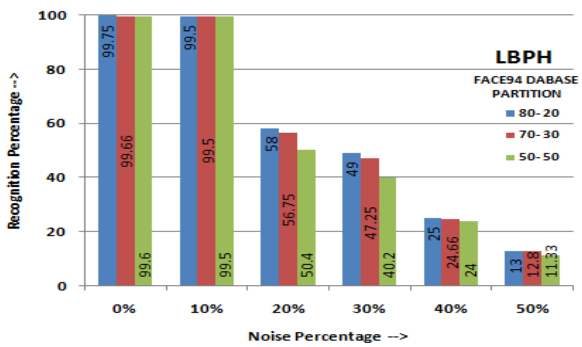


Fig. 5. Recognition Rate graph with different noise Percentages by LBPH method for FACE94 database.

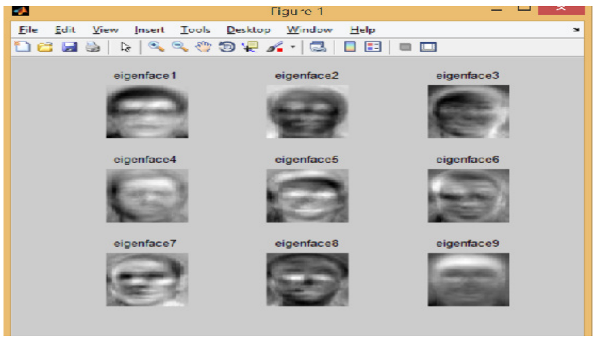


Fig. 6. Nine sample eigenface images from ORL database.

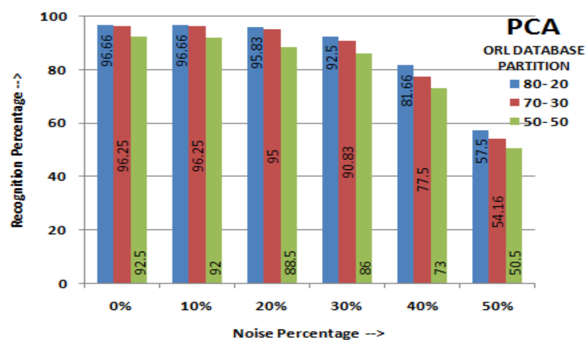


Fig. 7. Recognition Rate graph with different noise Percentages by PCA method for ORL database.

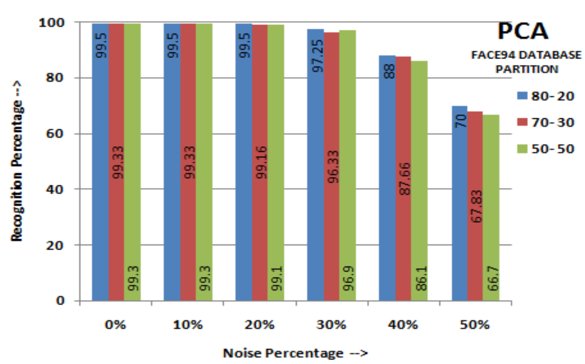


Fig. 8. Recognition Rate graph with different noise Percentages by PCA method for FACE94 database.

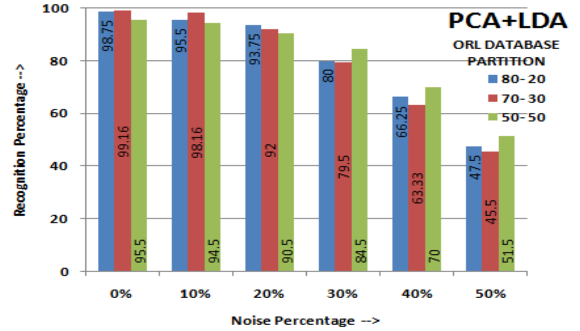


Fig. 9. Recognition Rate graph with different noise Percentages by PCA+LDA method for ORL database.

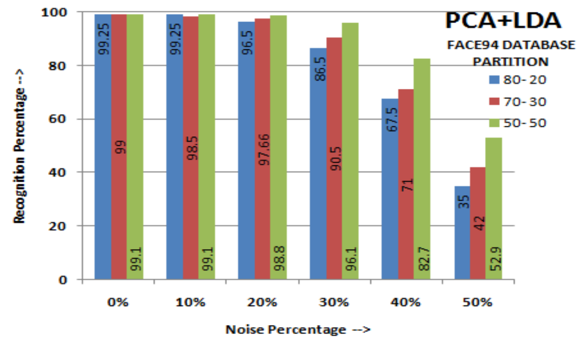


Fig. 10. Recognition Rate graph with different noise percentages by PCA+LDA method for FACE94 database.

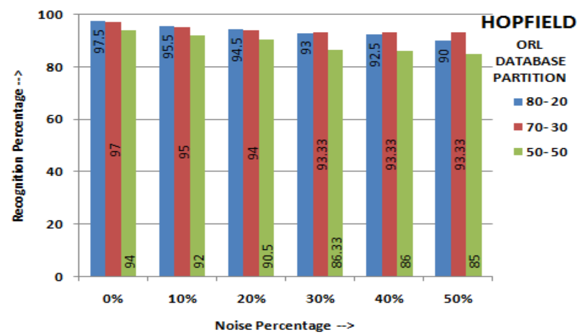


Fig. 11. Recognition Rate graph with different noise percentages by HOPFIELD model for ORL database.

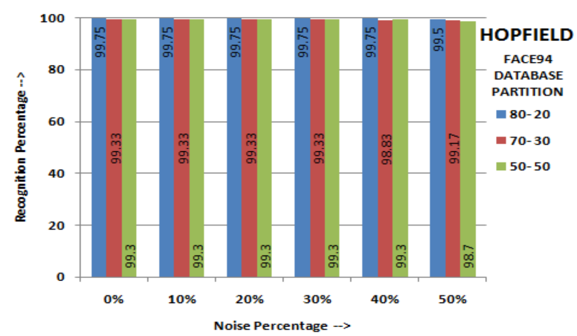


Fig. 12. Recognition Rate graph with different noise percentages by HOPFIELD model for FACE94 database.

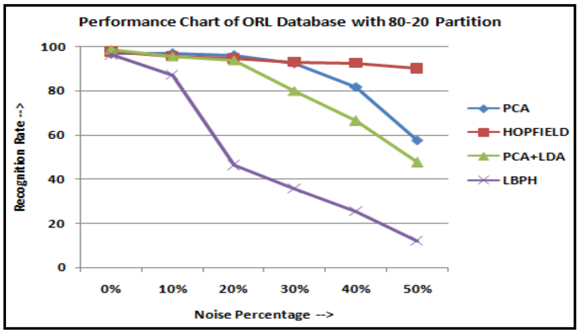


Fig. 13. Comparison graph of different methods with 80-20 data partition on ORL database.

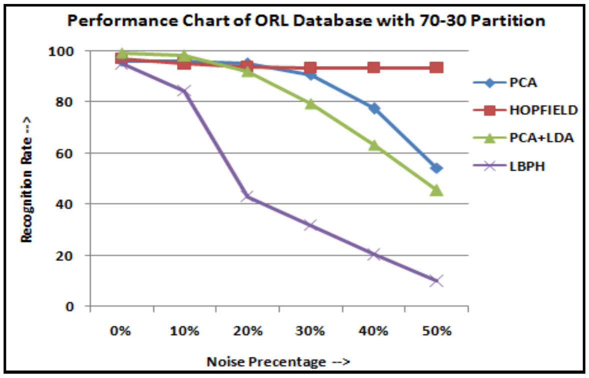


Fig. 14. Comparison graph of different methods with 70-30 data partition on ORL database.

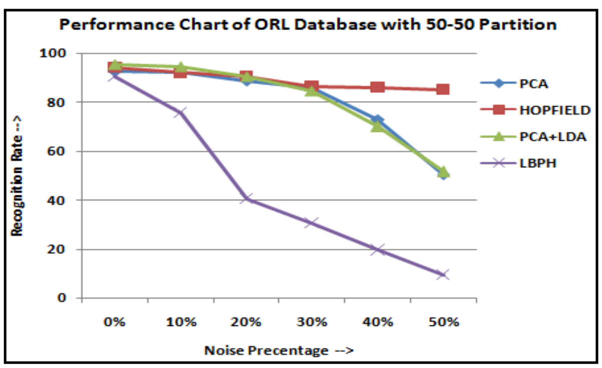


Fig. 15. Comparison graph of different methods with 50-50 data partition on ORL database.

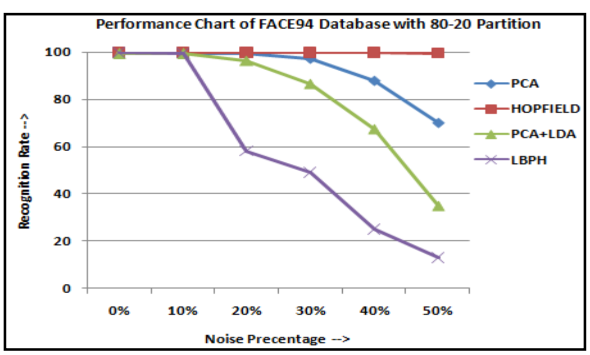


Fig. 16. Comparison graph of different methods with 80-20 data partition on FACE94 database.

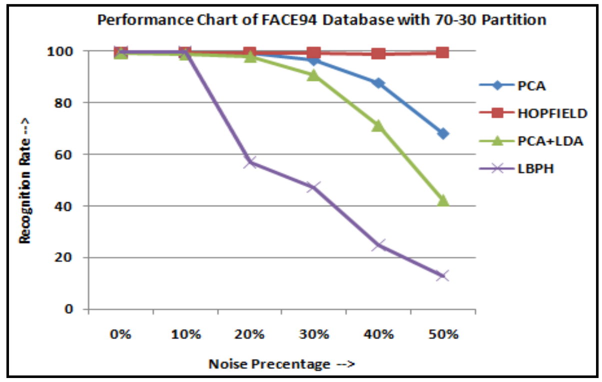


Fig. 17. Comparison graph of different methods with 70-30 data partition on FACE94 database.

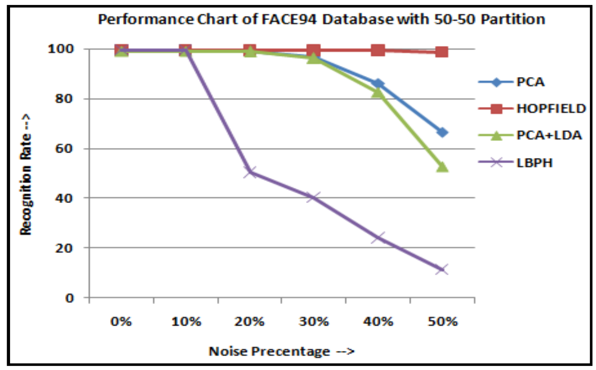


Fig. 18. Comparison graph of different methods with 50-50 data partition on FACE94 database.

Table 1: Recognition rate (in %) of LBPH with noise percentage.

Face Database	Data in Train-Test	No Noise	10% Noise	20% Noise	30% Noise	40% Noise	50% Noise
ORL Database	80-20	96.25	87.00	46.25	35.75	25.50	12.00
	70-30	95.00	84.50	43.00	31.66	20.33	10.00
	50-50	90.50	75.50	40.55	30.50	19.50	09.50
FACE94 Database	80-20	99.75	99.50	58.00	49.00	25.00	13.00
	70-30	99.66	99.50	56.75	47.25	24.66	12.80
	50-50	99.60	99.50	50.40	40.20	24.00	11.33

Table 2: Recognition rate (in %) of PCA with noise percentage.

Face Database	Data in Train-Test	No Noise	10% Noise	20% Noise	30% Noise	40% Noise	50% Noise
ORL Database	80-20	96.66	96.66	95.83	92.50	81.66	57.50
	70-30	96.25	96.25	95.00	90.83	77.50	54.16
	50-50	92.50	92.00	88.50	86.00	73.00	50.50
FACE94 Database	80-20	99.50	99.50	99.50	97.25	88.00	70.00
	70-30	99.33	99.33	99.16	96.33	87.66	67.83
	50-50	99.30	99.30	99.10	96.90	86.10	66.70

Table 3: Recognition rate (in %) of PCA+LDA with noise percentage.

Face Database	Data in Train-Test	No Noise	10% Noise	20% Noise	30% Noise	40% Noise	50% Noise
ORL Database	80-20	98.75	95.50	93.75	80.00	66.25	47.50
	70-30	99.16	98.16	92.00	79.50	63.33	45.50
	50-50	95.50	94.50	90.50	84.50	70.00	51.50
FACE94 Database	80-20	99.25	99.25	96.50	86.50	67.50	35.00
	70-30	99.00	98.50	97.66	90.50	71.00	42.00
	50-50	99.10	99.10	98.80	96.10	82.70	52.90

Table 4: Recognition rate (in %) of Hopfield model with noise percentage storing all images and testing with all images.

Face Database	Data in Train-Test	No Noise	10% Noise	20% Noise	30% Noise	40% Noise	50% Noise
ORL	100-100	100	100	100	100	100	93.25
FACE94	100-100	100	100	100	100	100	95.00

Table 5: Recognition rate (in %) of Hopfield model with noise percentage.

Face Database	Data in Train-Test	No Noise	10% Noise	20% Noise	30% Noise	40% Noise	50% Noise
ORL Database	80-20	97.50	95.50	94.50	93.00	92.50	90.00
	70-30	97.00	95.00	94.00	93.33	93.33	93.33
	50-50	94.00	92.00	90.50	86.33	86.00	85.00
FACE94 Database	80-20	99.75	99.75	99.75	99.75	99.75	99.50
	70-30	99.33	99.33	99.33	99.33	98.83	98.17
	50-50	99.30	99.30	99.30	99.30	99.30	98.70

VI. DISCUSSION AND CONCLUSION

The present study demonstrated the comparison of accuracy of face recognition rate in four different ways. Three feature extraction methods and Hopfield network method with three different training and testing dataset partition has been implemented in this paper. These feature extraction methods are Local Binary Pattern Histograms (LBPH), Principal Component Analysis (PCA) and Principal Component Analysis with Linear Discriminant Analysis (PCA + LDA). In this study a recurrent auto-associative memory model named Hopfield neural network has been implemented and the result of recognition is compared with other three models and their comparison graphs in three training and testing data partition scenarios such as (i) 80% training data and 20% testing data (ii) 70% training data and 30% testing data and (iii) 50% training data and 50% testing data with different noise percentages like 10%, 20%, 30%, 40% and 50% noise are shown in Fig. 6. All these methods were tested with two different datasets such as 400 samples of 40 individuals of ORL dataset and 2000 samples of 100 individuals of FACE94 dataset.

Various line graphs are plotted in Fig.13 through Fig.18 to demonstrate the comparison of all four methods discussed in this paper with each database and with each partition separately. Figs. 13, 14 and 15 represents recognition rate of all four methods implemented in this work for ORL database with 80 – 20, 70 – 30 and 50 – 50 data partitions respectively. Similarly, Fig. 16, 17 & 18 represents recognition rate of all four methods implemented in this work for FACE94 database with 80 – 20, 70 – 30 and 50 – 50 data partitions respectively.

In our study, it has been observed that Hopfield neural network is outperformed when noise percentage increased for both databases. All these four methods discussed in this paper shows nearly same results when no noise is added to the testing data but PCA with LDA method is little ahead among all four methods as it takes efficient time and space since we further reduced the dimension of data from PCA projected data. Hopfield model was tested in two ways; one was with different training-testing data partition as discussed above and in other one, we stored all face patterns into Hopfield memory network and then recalls these stored patterns by adding 10 – 50 percentages of noise and in the later case recalling rate is almost 100% even with 70% of noisy data for both the databases. With 80% of noisy data the recalling rate is 93.25% for ORL database whereas 95% for FACE94 database.

VII. FUTURE SCOPE

Approaches studied in this paper are initially successful and encouraging in face recognition; however more research work is to be done for future work for larger dataset and increased in posed, illumination, and expression variation in face images. In future work we will try to design a more robust system with the help of convolutional neural network to answering these underline causes for real time as well as for dummy faces. Genetic algorithm will be used to optimize Hopfield network as fully connected layer in CNN for storage and recall. The same work will also be evaluated by using fuzzy system and their comparison will be analyzed.

Conflict of Interest. The authors declare that there is no conflict of interest of any kind on this research work.

ACKNOWLEDGEMENT

We owe our deep gratitude to the department of Computer Science and Applications, Utkal University, Odisha, for supporting our research work and also extend our sincere esteems to all authors of reference.

REFERENCES

[1]. Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of cognitive neuroscience*, *3*(1), 71-86.

[2]. Park, M. S., Na, J. H., & Choi, J. Y. (2005). PCA-based feature extraction using class information. In *2005 IEEE International Conference on Systems, Man and Cybernetics*, *1*, 341-345). IEEE.

[3]. Yang, J., Zhang, D., Frangi, A. F., & Yang, J. Y. (2004). Two-dimensional PCA: a new approach to appearance-based face representation and recognition. *IEEE transactions on pattern analysis and machine intelligence*, *26*(1), 131-137.

[4]. Etemad, K., & Chellappa, R. (1997). Discriminant analysis for recognition of human face images. *J. Optical Soc. Am., 14*(8), 1724-1733.

[5]. Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7), 711-720.

[6]. Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 24*(7), 971-987.

[7]. Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *28*(12), 2037-2041.

[8]. Malkapurkar, A., Patil, R., & Murarka, S. (2011). A new technique for LBP method to improve face recognition. *International Journal of Emerging Technology and Advanced Engineering*, *1*(1), 67-71.

[9]. Ahonen, T., Hadid, A., & Pietikäinen, M. (2004). Face recognition with local binary patterns. In *European conference on computer vision* (469-481). Springer, Berlin, Heidelberg.

[10]. Bartlett, M. S., Movellan, J. R., & Sejnowski, T. J. (2002). Face recognition by independent component analysis. *IEEE Transactions on neural networks*, *13*(6), 1450-1464.

[11]. Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, *29*(1), 51-59.

[12]. Hopfield, J. J., & Tank, D. W. (1986). Computing with neural circuits: A model. *Science*, *233*(4764), 625-633.

[13]. Tank, D. W., & Hopfield, J. J. (1987). Collective computation in neuron like circuits. *Scientific American*, *257*(6), 104-115.

[14]. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America* , *79*(8), 2554-2558.

[15]. http://www.cl.cam.ac.uk/Research/DTG/attachive: pub/data/ att_faces.zip

[16]. https: //cswww.essex.ac.uk/mv/allfaces/faces94.html

[17]. Ahonen, T., Pietikäinen, M., Hadid, A., & Mäenpää, T. (2004). Face recognition based on the appearance of local regions. In Proceedings of the 17th *International Conference on Pattern Recognition, IEEE Computer Society* (*ICPR*), *3*, 153-156.

[18]. Meena, K., & Suruliandi, A. (2011). Local binary patterns and its variants for face recognition. In *2011 International Conference on Recent Trends in Information Technology (ICRTIT)* (pp. 782-786). IEEE.

[19]. Huang, X., Li, S. Z., & Wang, Y. (2004). Shape localization based on statistical method using extended local binary pattern. In *Third International Conference on Image and Graphics (ICIG'04)* (pp. 184-187). IEEE.

[20]. Zhang, G., Huang, X., Li, S. Z., Wang, Y., & Wu, X. (2004). Boosting local binary pattern (LBP)-based face recognition. In *Chinese Conference on Biometric Recognition* (pp. 179-186). Springer, Berlin, Heidelberg.

[21]. Li, X., & Chen, G. (2012). Face recognition based on PCA and SVM. In *2012 Symposium on Photonics and Optoelectronics* (pp. 1-4). IEEE.

[22]. Chang, K., Bowyer, K. W., Sarkar, S., & Victor, B. (2003). Comparison and combination of ear and face images in appearance-based biometrics. *IEEE Transactions on pattern analysis and machine intelligence*, *25*(9), 1160-1165.

[23]. Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D. L., & Weng, J. (1998). Discriminant analysis of principal components for face recognition. In *Face Recognition* (pp. 73-85). Springer, Berlin, Heidelberg.

[24]. Soni, N., Singh, A., Kapoor, A., & Sharma, E. K. (2016). Face recognition using cloud Hopfield neural network, International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET): 416-419.

[25]. Dai, Y., & Nakano, Y. (1998). Recognition of facial images with low resolution using a Hopfield memory model. *Pattern Recognition*, *31*(2), 159-167.

How to cite this article: Sahoo, R. C. and Pradhan, S. K. (2019). A Comparative Study on Hopfield Network with LBP, PCA and LDA for Face Recognition in Distorted Face Images. *International Journal on Emerging Technologies*, *10*(4): 102–110.