IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 7, February 2017)

A Novel Method for Face Recognition and Facial Expression Identification using PCA, Neural Network and Random Forest Classifier

Sanjay Kumar Natvarsinh Solanki
M. Tech. Scholar
Jagadguru Dattatray College of Technology, Indore (India)
sanjaysolanki3@gmail.com

Kuntal Barua
Assistant Professor
Jagadguru Dattatray College of Technology, Indore (India)
kuntal.barua@gmail.com

Khushboo Sawant
Assistant Professor
Jagadguru Dattatray College of Technology, Indore (India)
sawantkhushboo@gmail.com

Abstract –**The main aim of this paper is to analyze the method of Principal Component Analysis (PCA) and its performance when applied to face recognition. In previous work [10], they use Euclidian Distance for recognition. Here we use Neural Network and Random Forest Classifier which has much higher recognition accuracy than other methods. This algorithm creates a subspace (face space) where the faces in a database are represented using a reduced number of features called feature vectors and the classifier calculates the similarity score for performance evaluation which will provide improved results in terms of recognition accuracy.**

Keywords – **Euclidian Distance, Principal Component Analysis (PCA), Neural Network, Random Forest Classifier.**

I. INTRODUCTION

The biometric recognition techniques, also termed as Biometrics is the process of identifying an individual with due of their physiological and behavioural patterns. Being the high diversity in characteristics, biometrics has achieved high interest among the researchers to classify an individual. Some of most profitable features that possess unique behaviour like face, fingerprint, hand geometry, iris, signature, DNA (Deoxyribonucleic Acid) etc. have respective importance to identify an individual. Out of these face consists of numerous distinctive points that could be associated in an algorithm of classifying patterns. Face biometrics do not require an individual to be present near identification system as the systems are self-sufficient to execute algorithms using face images as input. This property is exploited in various security applications such as surveillance at public places (for example: airports, banks, traffic signals, train platforms, bus stations etc.). Images captured at any location, filter the face areas and match with existing database of suspected people. The systems are designed to work unremitted with less human efforts and acceptable level of precision.

Biometrics recognizes the humans in its study through one or more traits of intrinsic or physical behaviours. The biometrics is gaining considerable interest and is accepted as technology that meets with growing concerns of security in sensitive areas for example: intelligence, forensic, government and commerce. Biometrics is a self-sufficient system with capability of an individual recognition via various biometric traits that could be physiological and behavioural. The fundamental use of a system could accumulate single or a combination of physiological and behavioural traits like fingerprint, face, hand geometry, iris retina etc.

The biometric is on its hand:

- Universal
- Unique
- Permanent
- Collective
- Performer
- Acceptable
- Genuine

The face recognition though being simple to human eyes is a tedious task for computational approaches. The face recognition scheme should possess sufficient parameters to recognize a face and also robust against noise. The easy scheme of face recognition is matching the pixels of test input with database image pixels in their corresponding position. If the total number of pixels matched is greater than the defined threshold percentage of matching, the face is considered to be authentic. But on a practical note, the input images captured are not always in standard position for matching. For example, the images from security cameras placed at higher altitude than the height of a person captures the images that have different face angles. In these

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 7, February 2017)

input images, the faces are tilt and could not be recognized with easy schemes.

Another problem that arises in face recognition are the quality of images that are given as input. The captured images have variation in actual size of face due to distance from where they are captured and also have varied skin color tone due to sunlight. Since the quality cannot be assured for input image, the algorithms are expected to perform with high rate of accuracy in given conditions. The solution for first consideration is a secondary step, primary step being the angle of face captured and the computational model used. Since at most of places only 2D face recognition models are installed, hence the accuracy of face recognition is subjected to angle of face or in high rotational cases is dependent of efficiency of observer. The noisy images in 2D computational model, the efficiency of mathematical algorithms are studied in terms of accuracy of detection. Though many researches claim their methods to be robust and having high efficiency, the assumptions they make are not validated in practical world.

The main objective of this research work is:

- To analyze the method of Principal Component Analysis (PCA) and its performance for face recognition.
- To propose Neural Network and Random Forest Classifier based approach for calculating the similarity score for performance evaluation.
- Accuracy, false acceptance and rejection ratio are the performance evolution parameters for this research work.

II. PROPOSED METHOD

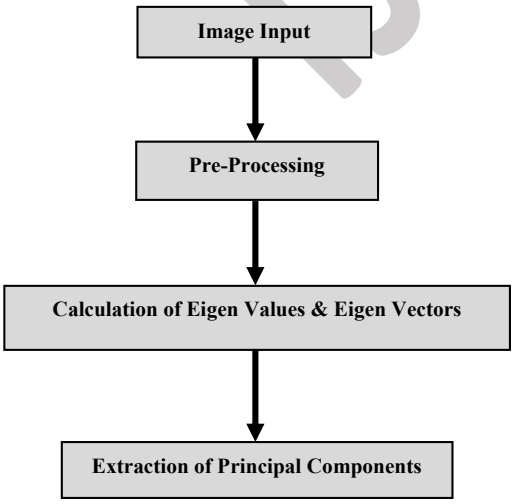

Figure 1: Flow diagram of PCA module

Mathematical Description of PCA

A. Statistics

The entire subject of statistics is based around the idea that we have this big set of data, and we want to analyze that set in terms of the relationships between the individual points in that data set.

B. Standard Deviation

To understand standard deviation, we need a data set. Statisticians are usually concerned with taking a sample of a population. To use election polls as an example, the population is all the people in the country, whereas a sample is a subset of the population that the statisticians measure. The great thing about statistics is that by only measuring (in this case by doing a phone survey or similar) a sample of the population, you can work out what is most likely to be the measurement if you used the entire population. Here's an example set:

$$X = [1\ 2\ 4\ 6\ 12\ 15\ 25\ 45\ 68\ 67\ 65\ 98\ ]$$

X refers to this entire set of numbers. The mean of the sample is given by the formula

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n} \quad (1)$$

Unfortunately, the mean doesn't tell us a lot about the data except for a sort of middle point. For example, these two data sets have exactly the same mean (10), but are obviously quite different:

$$[0\ 8\ 12\ 20]\ and\ [8\ 9\ 11\ 12]$$

It is the spread of the data that is different. The Standard Deviation (SD) of a data set is a measure of how spread out the data is. SD is given by the formula

$$s = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)}} \quad (2)$$

SD is the average distance from the mean of the data set to a point. The data set given below has a mean of 10, but its standard deviation is 0, because all the numbers are the same. None of them deviate from the mean.

$$[10\ 10\ 10\ 10] \quad (3)$$

C. Variance

Variance is another measure of the spread of data in a data set. In fact it is almost identical to the standard deviation. The formula is simply the SD squared.

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)} \quad (4)$$

D. Covariance

The last two measures we have looked at are purely 1-dimensional. However many data sets have more than one dimension, and the aim of the statistical analysis of these data sets is usually to see if there is any relationship between the dimensions.

Covariance is such a measure. Covariance is always measured between two dimensions. If you calculate the covariance between one dimension and itself, you get the variance. So, if you had a three-

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 7, February 2017)

dimensional data set (x, y, z), then you could measure the covariance between x and y dimensions, the x and z dimensions and the y and z dimensions. The formula for covariance is very similar to the formula for variance. The formula for variance could also be written like this,

$$var(x) = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(X_i-\bar{X})}{(n-1)} \quad (5)$$

The formula for covariance is given by,

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{(n-1)} \quad (6)$$

The exact value of covariance is not as important as its sign (i.e. Positive or negative). If the value is positive, then it indicates that both dimensions increase together. If the value is negative, then as one dimension increases, the other decreases. If the covariance is zero, it indicates that the two dimensions are independent of each other.

E. Covariance Matrix

Covariance (cov) is always measured between two dimensions. If we have a data set with more than two dimensions, there is more than one cov measurement that can be calculated. For example, from a three dimensional data set ($dimensions\ x, y, z$) you could calculate cov (x, y), cov (y, z), cov (y, z). In fact, for an n dimensional data set, you can calculate different covariance values.

$$\frac{n!}{(n-2)!*2} \quad (7)$$

A useful way to get all the possible cov values between all the different dimensions is to calculate them all and put them in a matrix. So, the definition for the cov matrix for a set of data with n dimensions is:

$$C^{m\times n} = \left(c_{i,j}, c_{i,j} = cov(Dim_i, Dim_i)\right) \quad (8)$$

Where, $C^{m\times n}$ is a matrix with n rows and n columns and Dim_x is the x^{th} dimension. The entry on row 2, column 3 is the cov value calculated between the 2nd dimension and the 3rd dimension.

F. Eigenvalues and Eigenvectors of Matrix

If the matrix is small, we can compute them symbolically using the characteristic polynomial. However, this is often impossible for larger matrices, in which case we must use a numerical method.

An important tool for describing eigenvalues of square matrices is the characteristic polynomial: saying that λ is an eigenvalue of A is equivalent to stating that the system of linear equations $(A - \lambda I)\, v = 0$ (where I is the identity matrix) has a non-zero solution v (an eigenvector), and so it is equivalent to the determinant

$$\det(A - \lambda I) = 0 \quad (9)$$

The function $p(\lambda) = det(A - \lambda I)$ is a polynomial in λ since determinants are defined as sums of products. This is the characteristic polynomial of A: the eigenvalues of a matrix are the zeros of its characteristic polynomial.

All the eigenvalues of a matrix A can be computed by solving the equation $pA(\lambda) = 0$. If A is a $n \times n$ matrix, then pA have degree n and A can therefore have at most n eigenvalues. Conversely, the fundamental theorem of algebra says that this equation has exactly n roots (zeroes), counted with multiplicity. All real polynomials of odd degree have a real number as a root, so for odd n, every real matrix has at least one real eigenvalue. In the case of a real matrix, for even and odd n, the non-real eigenvalues come in conjugate pairs.

Once the eigenvalues λ are known, the eigenvectors can then be found by solving:

$$(A - \lambda I)v = 0 \quad (10)$$

An example of a matrix with no real eigenvalues is the 90-degree clockwise rotation (equation 11) whose characteristic polynomial is $\lambda^2 + 1$ and so its eigenvalues are the pair of complex conjugates $i, -i$. The associated eigenvectors are also not real.

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (11)$$

Similarity score for performance evaluation will be calculated using Neural Network Classifier.

Classification using Neural Network Classifier

Back Propagation Neural Network (BPNN) generates complex decision boundaries in feature space. BPNN in specific circumstances resembles Bayesian Posterior Probabilities at its output. These conditions are essential to achieve low error performance for given set of features along with selection of parameters such as training samples, hidden layer nodes and learning rate. In else case, the performance of BPNN could not be evaluated. For W number of weights and N number of nodes, numbers of samples (m) are depicted to correctly classify future samples in following manner:

$$m \geq O\left(\frac{W}{\epsilon}\log\frac{N}{\epsilon}\right) \quad (12)$$

The theoretical computation of number of hidden nodes is not a specific process for hidden layers. Testing method is commonly entertained for selection of these followed in the constrained environment of performance [11].

Classification using Random Forest Classifier

The tree structured classifiers combine to form a random forest classifier (Figure 2) [12]. The input objects are sourced to every tree and individual unit vector is voted by every tree. The mode from individual classes with maximum number of votes is

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 7, February 2017)

given as output class by the forest. More than just prediction the random forest can be trained to achieve information on data proximities for feature generation. The variables closer to responsible variables are predicted by their importance while their relation is the function of partial independencies.

In training of N cases, the trees are trained for sampling [13]. This sampling is a random process that replaces the original data with the bootstrap sample. Out of M number of input variables, m variables are selected in random manner ($m \ll M$), and best split on these predictors split the node. During the entire operation of training, the value of m is hold for constant. Generally the value of m is selected M times smaller than M inputs. Without pruning, each tree grows to the maximum possible range. This training generates multiple numbers of trees with the maximum value decided by N_{tree}. The length of tree roots (depth of tree) is estimated via parameter node size (i.e. no. of leaf nodes) that is generally fixed to unity. In testing phase the input is fed to forest that runs to all the trees and classification from each individual is recorded as vote. The instance that gets maximum no. of votes is declared as winner or output.

About one-third of the cases are subsided or neglected in the training set of a tree when drawn by sampling with replacement. This Out-of-Bag data is employed for the test set that gets unbiased estimation of the classification errors. Henceforth the need for separate test and cross validation to obtain an unbiased minimizes estimate the test set error. The out of bag estimates are unbiased [14].

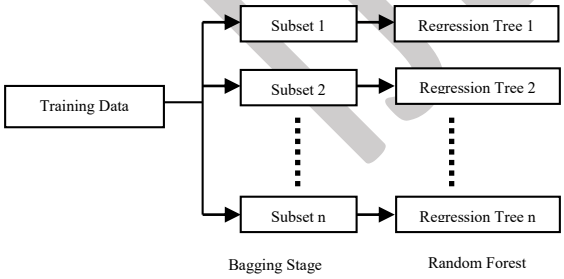

Figure 2(a): Algorithmic Sequence of Random Forest Classifier Training Phase

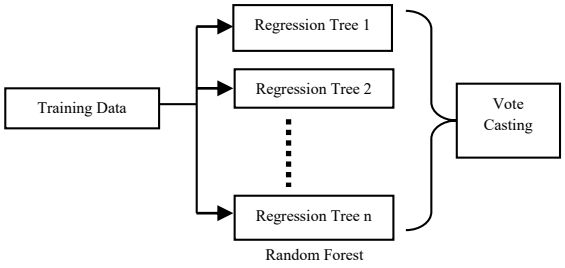

Figure 2(b): Algorithmic Sequence of Random Forest Classifier Testing Phase

III. Simulation And Results

The performance of proposed algorithms has been studied by means of MATLAB simulation.

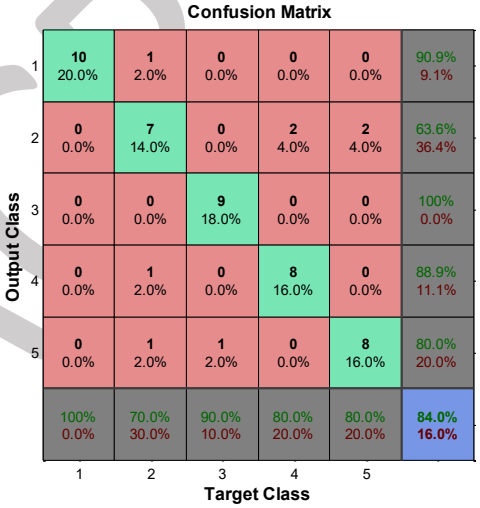

Figure 3: Confusion matrix of plot for PCA-NN approach

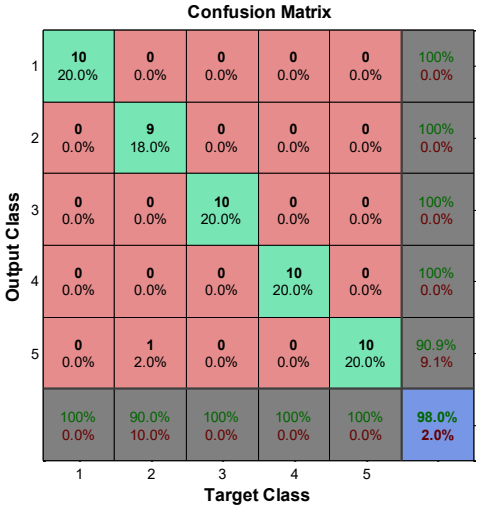

Figure 4: Confusion matrix of plot for PCA and Random Forest Classifier approach

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 7, February 2017)

IV. CONCLUSION

We have implemented a facial expression recognition system using PCA-NN and PCA-Random Forest Classifier. Principal Components Analysis is a method that reduces data dimensionality by performing a covariance analysis between factors. Confusion matrix demonstrates that the proposed research work gives more accuracy with PCA-Random Forest than the PCA-NN based approach.

REFERENCE

[1] Z. Mu-chun, "Face Recognition Based on Fast ICA and RBF Neural Networks", IEEE, 20-22 Dec. 2008.
[2] W. Wang, "Face Recognition Based On Radial Basis Function Neural Networks", IEEE, 20-20 Nov. 2008.
[3] J. Youyi and L. Xiao, "A Method for Face Recognition Based On Wavelet Neural Network", IEEE, 2010.
[4] D.N Pritha, L. Savitha and S.S. Shylaja, "Face Recognition by Feed forward Neural Network Using Laplacian of Gaussian Filter and Singular Value Decomposition", IEEE, 5-7 Aug. 2010.
[5] Hussein Rady, "Face Recognition using Principle Component Analysis with Different Distance Classifiers", IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.10, October 2011.
[6] Sukhvinder Singh, Meenakshi Sharma and Dr. N Suresh Rao, "Accurate Face Recognition Using PCA and LDA", International Conference on Emerging Trends in Computer and Image Processing (ICETCIP'2011) Bangkok Dec., 2011.
[7] V. P. Kshirsagar, M. R. Baviskar, M. E. Gaikwad, "Face recognition using Eigenfaces", IEEE 3rd International Conference on Computer Research and Development (ICCRD), Vol. 2, March 2011.
[8] M. N. Shah Zainudin., Radi H. R., S. Muniroh Abdullah., Rosman Abd. Rahim. M. Muzafar Ismail., M. Idzdihar Idris., H. A. Sulaiman., Jaafar A., "Face Recognition using Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA)", International Journal of Electrical & Computer Sciences IJECS-IJENS Vol:12 No:05, 2012.
[9] Mohammad Said El-Bashir, "Face Recognition Using Multi-Classifier", IEEE, 2012.
[10] Meher SS, Maben P. "Face recognition and facial expression identification using PCA", IEEE International Conference In Advance Computing (IACC), pp. 1093-1098, 2014.
[11] Christos Stergiou and Dimitrios Siganos, "Neural Networks", Report available at: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
[12] Ruchika Malhotra and Ankita Jain. Fault Prediction Using Statistical and Machine Learning Methods for Improving Software Quality. Journal of Information Processing Systems, Vol.8, No.2, June 2012.
[13] Breiman, L. Manual on setting up, using, and understanding random forests v3. 1. Statistics Department University of California Berkeley, CA, USA, 2002.
[14] Saida, B., Khaled, E. E. and Geol, N. Issues in Validating Object-Oriented Metrics for Early Risk Prediction. By Cistel Technology210 Colonnade Road Suite 204 Nepean, Ontario Canada K2E 7L5, 1999.