FACE DETECTION AND RECOGNITION USING BACK PROPAGATION NEURAL NETWORK (BPNN)

[*1]Ms. Vijayalakshmi. T , [*2] Mrs. Ganga T. K

[*1]*M.phil Research Scholar, Department of computer Science Muthurangam Government Arts College (Autonomous), Vellore, Tamilnadu, India.*

[*2]*Assistant Prof, Department of Computer Science Muthurangam Government Arts College (Autonomous), Vellore.*

---***---

Abstract - *Face Recognition is one of the most important and fastest growing biometric areas during the last several years and become the most successful application in image processing and broadly used in security systems. A real-time system for recognizing faces using mobile device or webcam was implemented. Face detection is the first basic step of any face recognition system. Viola-Jones method is used to detect and crop face area from the image. Feature extraction considered as a main challenge in any face recognition system. Principal Component Analysis (PCA) is efficient and used for feature extraction and dimension reduction. Back Propagation Neural Network (BPNN) and Radial Basis Function (RBF) are used for classification process. RBF is considered the result of BPNN output layer as input. The system is tested and achieves high recognition rates. Information about individuals was stored in a database.*

Key Words: *Face Detection, Face Recognition, Feature Extraction, Biometrics, Neural Network, PCA, BPNN, RBF*

I. INTRODUCTION

Face recognition is very important for our daily life. It can be used for remote identification services for security in areas such as banking, transportation, law enforcement, and electrical industries, etc. For this security access project is aimed at demonstrating facial recognition techniques that could antiquate, substitute, or otherwise, supplement, conventional key, and can be used as an alternative to existing fingerprint biometrics method. A computerized system equipped with a digital camera can identify the face of a person and determine if the person is authorized to start the vehicle. This integrated system would be able to authorize a user before switching on the vehicle with a key. Whilst facial recognition systems are by now readily available in the market, the vast majority of them are installed at large open spaces, such as in airport halls. The focus of this project is, thus, to compare the extracted feature with face image database for the recognition analysis using Neural Network.

Face recognition is a visual pattern recognition problem. In detail, a face recognition system with the input of an arbitrary image will search in database to output

people's identification in the input image. A face recognition system generally consists of four modules as depicted in Figure 1: detection, alignment, feature extraction, and matching, where localization and normalization (face detection and alignment) are processing steps before face recognition (facial feature extraction and matching) is performed [1].
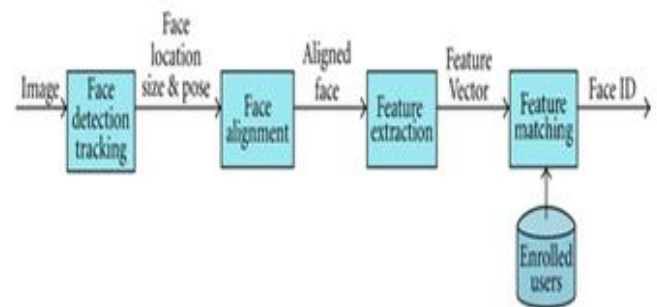

Figure 1: Structure of a face recognition system

Face detection segments the face areas from the background. In the case of video, the detected faces may need to be tracked using a face tracking component. Face alignment aims at achieving more accurate localization and at normalizing faces thereby, whereas face detection provides coarse estimates of the location and scale of each detected face. Facial components, such as eyes, nose, and mouth and facial outline, are located; based on the location points, the input face image is normalized with respect to geometrical properties, such as size and pose, using geometrical transforms or morphing. The face is usually further normalized with respect to photometrical properties such illumination and gray scale. After a face is normalized geometrically and photometrically, feature extraction is performed to provide effective information that is useful for distinguishing between faces of different persons and stable with respect to the geometrical and photometrical variations. For face matching, the extracted feature vector of the input face is matched against those of enrolled faces in the database; it outputs the identity of the face when a match is found with sufficient confidence or indicates an unknown face otherwise.

II. RELATED WORK

ARTIFICIAL NEURAL NETWORK

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process.

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if" questions.

Types of neural networks:

Feed forward neural network

The feed forward neural networks are the first and arguably simplest type of artificial neural networks devised. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network.

Single-layer perceptron

The earliest kind of neural network is a single-layer perceptron network, which consists of a single layer of output nodes; the inputs are fed directly to the outputs via a series of weights. In this way it can be considered the simplest kind of feed-forward network. The sum of the products of the weights and the inputs is calculated in each node, and if the value is above some threshold (typically 0) the neuron fires and takes the activated value (typically 1); otherwise it takes the deactivated value (typically -1). Neurons with this kind of activation function are also called McCulloch-Pitts neurons or threshold neurons.

Multilayer perceptron

The MLP neural network consists of an input layer, one or more hidden layers, and an output layer. Each layer is made up of units. The inputs to the network correspond to the attributes measured for each training tuple. The inputs are fed simultaneously into the units making up the input layer. These inputs pass through the input layer and are then weighted and fed simultaneously to a second layer of "neuron like" units, known as a hidden layer. The outputs of the hidden layer units can be input to another hidden layer, and so on. The number of hidden layers is arbitrary, although in practice, usually only one is used.

Back propagation:

Back propagation is a common method of training artificial neural networks so as to minimize the objective function. It is a supervised learning method, and is a generalization of the delta rule. It requires a dataset of the desired output for many inputs, making up the training set. It is most useful for feed-forward networks (networks that have no feedback, or simply, that have no connections that loop). The term is an abbreviation for "backward propagation of errors".

III. PREVIOUS IMPLEMENTATIONS

Face Recognition is a multi-class classification problem in which face is classified as belonging to any subject. Face recognition is substantially different from classical pattern recognition problems, such as object recognition. The shapes of the objects are usually different in an object recognition task, while in face recognition one always identifies objects with the same basic shape.

Henry Schneider man, A statistical method for 3D objects detection. We represent the statistics of both object appearance and "non-object" appearance using a product of histograms. Each histogram represents the joint statistics of a subset of wavelet coefficients and their position on the object [3]. N. Revathy, T.Guhan Face recognition is to find the best match of an unknown image against a database of face models or to determine whether it does not match any of them well. In this method, we use back propagation neural network for implementation. It is an information processing system that has been developed as a generalization of the mathematical model of human recognition. The function of a neural network is to produce an output pattern when presented with an input pattern [2]. Zdravko Liposcak, Sven Loncaric A grey-level profile image is thresholder to produce a binary image, representing the face region. After normalizing the area and orientation of this shape using basic morphological operations, dilation and erosion, we simulate hair growth and haircut and produce two new profile silhouettes. From this three profile shapes feature vectors are obtained using distances between outline curve points and shape centroid [6].

IV. SYSTEM IMPLEMETNATION

A machine learning technique that uses Bayesian inference to obtain parsimonious solutions for regression and classification it has an identical functional form to the support vector machine, but provides probabilistic classification. It is actually equivalent to a Gaussian process model with covariance function:

$$k(x,x') = \sum_{j=1}^{n} \frac{1}{\alpha j} \partial(x, xj)\partial(x', xj)$$

Where φ is the kernel function (usually Gaussian), and *x1,...,xN* are the input vectors of the training set. Multilayer Perceptron (MLP) network is the most widely used neural network classifier. MLPs are universal approximates. MLPs are valuable tools in problems when one has little or no knowledge about the form of the relationship between input vectors and their corresponding outputs.

Feed-Forward Backpropagation

A Feed-Forward network consists of a series of layers. The first layer has a connection from the network input. Each subsequent layer has a connection from the previous layer. The final layer produces the network's output. Feed-forward networks can be used for any kind of input to output mapping. Specialized versions of the feed-forward network include fitting (fitnet) and pattern recognition (patternnet) networks. Feed-forward backpropagation network is simply the application of backpropagation procedure into the feed-forward networks such that every time the output vector is presented, it is compared with the desired value and the error is computed. The error value tells us how far the network is from the desired value for a particular input and the backpropagation procedure is to minimize the sum of error for all the training samples.

The error is computed by,

Error = (desired value – actual value)²

The syntax of Feed-Forward Backpropagation takes the following arguments:

net = feedforwardnet (hidden-Sizes, training-function)

where,
hidden-sizes – Row vector of one or more hidden layer sizes (Default = 10)
training-function – Training function (Default = 'trainlm')
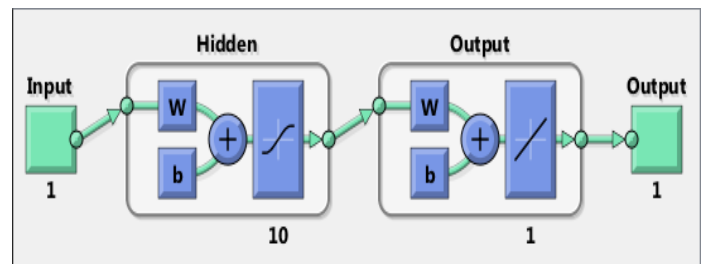The functions return a newfeed-forward backpropagation network.

Fig: Feed-Forward Network

This example shows how to use feedforward neural network to solve a simple problem.

```
[x,t] = simplefit_dataset;
      net = feedforwardnet(10);
      net = train(net,x,t);
view(net)
      y = net(x);
perf = perform(net,y,t)
```

Cascade-Forward Backpropagation

Cascade-Forward networks are similar to feed-forward networks, but include a connection from the input and every previous layer to following layers. As with feed-forward networks, two-or more layer cascade-network can learn any finite input-output relationship arbitrarily well given enough hidden neurons.

The syntax of Cascade-Forward Backpropagation takes the following arguments:

net = cascadeforwardnet (hidden-Sizes, training-function)
where,
hidden-sizes – Row vector of one or more hidden layer sizes (Default = 10)
training-function – Training function (Default = 'trainlm')
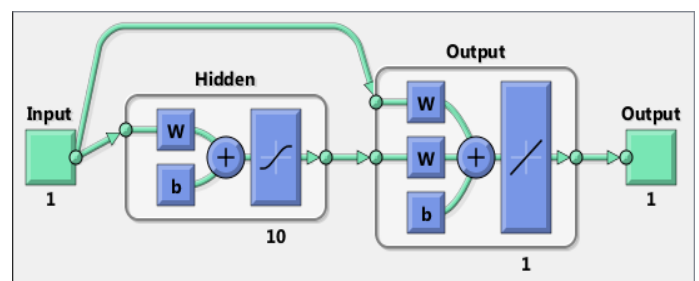The functions return a new Cascade-Forward Backpropagation network.

Fig: Cascade-Forward Backpropagation

Certain points are noteworthy while developing either a feed-forward or cascade-forward networks as follows:

- The transfer functions can be any differentiable transfer function such as tansig, logsig or purelin.
- The training function can be any of the backpropagation training functions such as trainlm, trainbfg, trainrp, traingd, traingdx etc.
- The learning function can be either of the following functions such as learngd or learngdm

Cascade-forward networks are similar to feed-forward networks, but include a connection from the input and every previous layer to following layers. As with feed-forward networks, a two-or more layer cascade-network can learn any finite input-output relationship arbitrarily well given enough hidden neurons.

Here a cascade network is created and trained on a simple fitting problem.

```
[x,t] = simplefit_dataset;
        net = cascadeforwardnet(10);
        net = train(net,x,t);
view(net)
        y = net(x);
        perf = perform(net,y,t)
```

Perceptron

Perceptrons are simple single-layer binary classifiers, which divide the input space with a linear decision boundary. Perceptrons can learn to solve a narrow range of classification problems. They were one of the first neural networks to reliably solve a given class of problem and their advantage is a simple learning rule.

The syntax of a perceptron takes the following arguments:

Perceptron(hardlimitTF, perceptronLF)

Where,
- hardlimitTF- hard limit Transfer function (Default = 'hardlim')
- perceptronLF- perceptron Learning rule (Default = 'learnp')
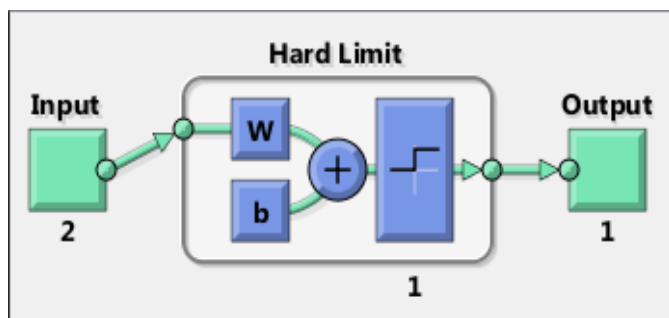
The functions return a new perceptron network.

Fig: Perceptron Network

Algorithm: Generate Data Set

Input: Training Data, Testing Data

Output: Decision Value

Method:

Step 1:   Load Dataset
Step 2: Classify Features (Attributes) based on class labels
Step 3:   Estimate Candidate Support Value
            While (instances! =null)
            Do
Step 4: Support Value=Similarity between each instance in the attribute Find Total Error Value
Step 5:   If any instance < 0
            Estimate
            Decision value = Support Value/Total Error
            Repeat for all points until it will empty
End If

Classification Tree Algorithm

Algorithm: Generate a Classification from the training tuples of data partition D.

Input:

- Data partition D, which is a set of training tuples and their associated class labels;

- Attribute list, the set of candidate attributes;

- Attribute selection method, a procedure to determine the splitting criterion that "best" Partitions the data tuples into individual classes. These criterions consist of a splitting Attribute and, possibly, either a split point or splitting subset.

Output: A decision tree

Method:

1. Create a node N;

2. If tuples in D are all of the same class, C then

3. Return N as a leaf node labeled with the class C

4. If attribute list is empty then

5. Return N as a leaf node labele

6. d with the majority class in D

7. Apply Attribute selection method (D, attribute list) to find the "best" splitting criterion

8. Label node N with splitting criterion

9. If splitting attribute is discrete-valued and multiway splits allowed then

10. Attribute list ← attribute list – splitting attribute

11. For each outcome j of splitting criterion

12. Let Dj be the set of data tuples in D satisfying outcome j

13. If Dj is empty then

14. Attach a leaf labeled with the majority class in D to node N

15. Else attach the node returned by Generate decision tree (Dj, attribute list) to node N

16. End for

17. Return N

V EVALUATION RESULT

The Regression plot from the UCI Machine Learning Repository is used to differentiate benign (non-Face Detection and Recognition ous) from malignant (Face Detection and Recognition ous) samples. To evaluate the effectiveness of our method, experiments on WDBC is conducted. This database was obtained from the university of Wisconsin hospital, Madison from Dr. William H. Wolberg. This is publicly available dataset in the Internet. Table 1 shows a brief description of the dataset that is being considered.

The dataset for this study are collected from the Wisconsin breast Face Detection and Recognition diagnosis database available in the UCI repository. All the data that have been collected are the results of diagnosis made through Biopsy procedures. There are 569 instances and 10 attributes which also contains patient's ID number as a separate attribute. All the values are encoded with four significant digits. The ten real-valued attributes are as follows:

EXPERIMENTAL RESULT

The experiment was carried out in MATLAB workspace (version number –MATLAB 2016a).MATLAB's Neural Network Toolbox (NNTool) provided various features to carry out the implementation part of the three algorithms chosen as mentioned in the previous sections of this document. Initially all the input, sample and the target data have been imported into the Matlab workspace to create and train the networks using the NNTool. The discussion is as follows:

The network for feed-forward backpropagation and cascade-forward backpropagation were created by setting the training function TRAINGDX and learning function LEARNGDM. The errors are computed using the Mean Squared Error (MSE) function. The number of neurons and layers in the network are initialized at random and the transfer function used is LOGSIG. For

perceptron, network is created using HARDLIM training function and LEARNP learning function. The network is thus created and trained number of times until satisfactory performance is met. The following figure shows the working analysis of feed-forward network using the tool.

(a) and (b) shows the plot regression analysis of a feed-forward network and a cascade-forward network with respect to the data computed for the analysis of the breast Face Detection and Recognition diagnosis.
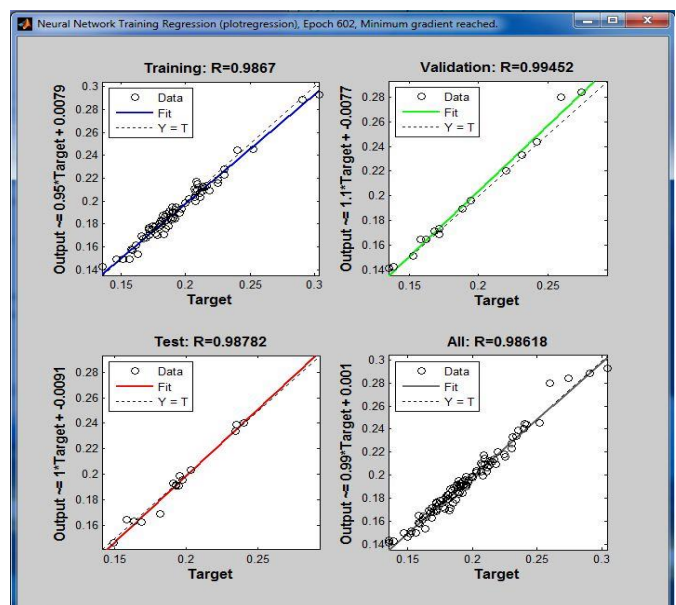

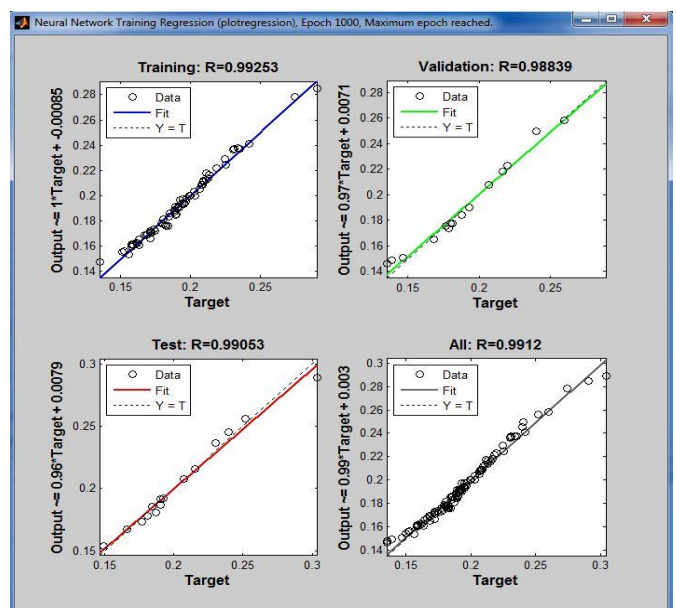**Fig: (a) Regression Plot of Feed-Forward Backpropagation**

Fig: (b) Regression Plot of Cascade-Forward Backpropagation

Figure(c) shows the Performance of a perceptron network. The Regression plot displays three divisions of output namely Training, Validation and Testing. The Regression R values measure the correlation between outputs and targets. An R value of 1 means a close relationship and 0 a random relationship.
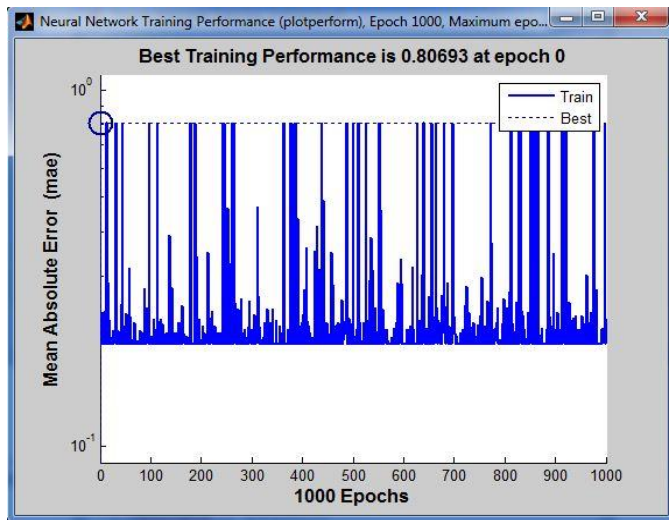

Fig: (c) Performance Plot of Perceptron Network

Based on the experimental analysis of the algorithms on the breast Face Detection and Recognition diagnosis data, it is shown that the cascade-forward backpropagation algorithm works better than the feed-forward backpropagation and perceptron networks by producing the network output close to its target, supplying the value 0.99 which has a linear close relationship with that of the target data. The performance graph of the cascade-forward backpropagation network is given as follows:
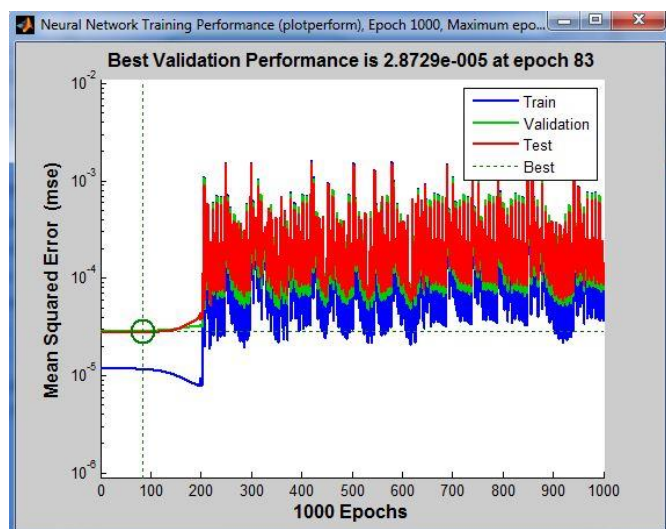

Fig: Performance graph of Cascade-Forward Backpropagation Network

Clustering is one of the most important unsupervised learning problems in that it deals with finding a structure in a collection of unlabeled data. In other words, clustering is the process of organizing objects into groups whose members are similar in some way.

CONCLUSION

In this work, a real-time face recognition system is implemented. Images are captured from a mobile device or webcam. Face detection is the first step in face recognition system. Viola-Jones method is used for face detection. BPA is used for feature extraction. Classification process is done using BPNN and RBF-NN. The neural networks aimed to provide artificial intelligence to the system. Neural networks using Back Propagation and Radial Basis Function is presented for face recognition. The BPNN method is preferred over other neural network methods because of its unique ability to minimize errors. The use of BPNN shows acceptable results. The recognition rate is increased when combine BPNN with RBF-NN. From these results, it can be concluded that this method has the acceptance ratio is more than 90 % and execution time of only few seconds. The results of the experiments indicate that the RBF-NN achieves the best performance while allowing less neuron in the hidden layer. The RBF-NN method shows also to be less sensitive to the choice of the training set. This method can be suitably extended for moving images and the images with varying background.

REFERENCES:

1. Tej Pal Singh, "Face Recognition by using Feed Forward Back Propagation Neural Network", International Journal of Innovative Research in Technology & Science, vol.1, no.1,

2. N.Revathy, T.Guhan, "Face recognition system using back propagation artificial neural networks", International Journal of Advanced Engineering Technology, vol.3, no. 1, 2012.

3. H.Schneiderman and T. Kanade, "A statistical method for 3D object detection applied to faces and cars," IEEE conference on Computer vision and pattern recognition, 13-15 June 2000, Hilton Head Island, pp. 746–751.

4. Ling-Zhi Liao, Si-Wei Luo, and Mei Tian "Whitened faces Recognition With PCA and ICA" IEEE Signal Processing Letters, vol. 14, no. 12, pp1008-1011, Dec. 2007.

5. Kwan-Ho Lin, Kin-Man Lam, and Wan-Chi Siu. "A New Approach using ModiGied Hausdorff Distances with EigenFace for Human Face Recognition" IEEE Seventh international

Conference on Control, Automation, Robotics and Vision , Singapore, 2-5 Dec 2002,pp 980-984.

6. Zdravko Liposcak, Sven Loncaric, "Face Recognition From Profiles Using Morphological Operations", IEEE Conference on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 1999.

7. Simone Ceolin, William A.P Smith, Edwin Hancock, "Facial Shape Spaces from Surface Normals and Geodesic Distance", Digital Image Computing Techniques and Applications, 9th Biennial Conference of Australian Pattern Recognition Society IEEE 3-5 Dec., 2007s, Glenelg, pp-416-423.

8. Ki-Chung Chung, Seok Cheol Kee ,and Sang Ryong Kim, "Face Recognition using Principal Component Analysis of Gabor Filter Responses", Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 1999,Proceedings. International Workshop IEEE 26-27 September, 1999, Corfu, pp-53-57.

9. Ms. Varsha Gupta, Mr. Dipesh Sharma, "A Study of Various Face Detection Methods", International Journal of Advanced Research in Computer and Communication Engineering), vol.3, no. 5, May 2014.

10. Irene Kotsia, Iaonnis Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines", IEEE transaction paper on image processing, vol. 16, no.1, pp-172-187, January 2007.

11. R.Rojas," The back propagation algorithm", Springer-Verlag, Neural networks, pp 149-182 1996.

12. Hosseien Lari-Najaffi, Mohammad Naseerudin and Tariq Samad,"Effects of initial weights on back propagation and its variations", Systems, Man and Cybernetics, Conference Proceedings, IEEE International Conference, 14-17 Nov, 1989,Cambridge, pp-218-219.