Analysis of Training Functions in a Biometric System

Anusree K
NSS College of Engineering, Palakkad
Kerala
India
anusreek08@gmail.com

Dr. Binu G S
NSS College of Engineering, Palakkad
Kerala
India
binu_g_s@rediffmail.com

Abstract**—**One of the commonly used Biometric methods is Face Classification. Face images are obtained from FEI face database. In this paper, different Training Functions of Neural Network are studied. In this research, a face recognition system is suggested based on feedforward backpropagation neural network Neural Network (FFBPNN) model. Each model is constructed separately with one input layer, 3 hidden layers and one output layer). Four ANN training algorithms (TRAINLM, TRAINBFG, TRAINGDX, and TRAINRP) are used to train each model separately. Performances using each of the training algorithms were evaluated based on mean square error and the best training algorithm is found for the face recognition.

Keywords — *Biometrics, Face Recognition, Feedforward backpropagation Neural Network*

*****

I. INTRODUCTION

Biometrics or Biometric authentication system identifies individuals by their unique physical traits. Traits can be either physiological or behavioral. Physiological trait is related to shape of the body, e.g. face, whereas behavioral is related to the behavior of the person which include voice.

A Biometric authentication system operates by the following process. First, it obtains raw biometric data from the subject. Raw biometric data may be face image, fingerprint, palm print, iris image etc. Then features are extracted from the biometric data. Next step is to compare these features extracted with those against stored template. Template is stored in the database during the enrollment. Applications of image classification are vast. Two main points among them can be pointed out as

- Security- Some areas are restricted to authorized persons only.
- Enforcement of Law

In this paper different training functions in Feedforward Backpropagation Network using neural network tool are investigated. Related works are included in section II. Section III covers the proposed work. Results are described in section IV. Conclusions are presented in section V.

II. RELATED WORK

Anil K. Jain, Arun Ross and Salil Prabhakar [2] defined biometric characteristics as any human physiological and/or behavioral characteristic that satisfies the following requirements: Universality, Distinctiveness, Permanence, and Collectability. Issues regarding a practical biometric system were explained as Performance, Acceptability and Circumvention. On the basis of application, a biometric System operates either in *verification* mode or *identification* mode. The system validates a person's identity by comparing the captured biometric data with his/her own biometric template(s) stored system database. This is the *verification* mode, whereas in the identification mode, the system identifies an individual by searching the templates of all the users in the database for a match. Accuracy of various biometrics like DNA, ear, face, fingerprint, gait, hand and finger geometry, keystroke, odor, palm print, retinal scan, iris, voice etc. were studied and compared. The applicability of a specific biometric technique depends heavily on the requirements of the application domain.

In this paper [3], R. Brunelli and Poggio explored two traditional techniques for face recognition. The first technique relied on the computation of a set of geometrical features from the picture of a face. Basic idea is to extract relative position and other parameters of distinctive features such as eye, mouth, nose and chin. In the second technique, Template Matching was experimented. Here, the image is represented as bi dimensional array of intensity values. It is then compared with the stored template. Performances of both techiniqus were compared for a specific application. Of the two techniques, Template Matching showed superior in recognition performance.

In [4], three different methods of face recognition such as geometric approach, elastic matching and neural networks were presented. In the geometric approach, 37 Points were experimentally selected from the face image. Then, distances in the feature space from a template image to every image in the database were calculated. 5 nearest face images were derived and if there were photos of the query person then the result was considered positive. In elastic matching, elastic transform was applied that will change geometry and image texture and then two images compared. Neural network approach is stated as follows: The recognition system must accept authorized people. All the other people are unauthorized should be rejected. For this, we train a system to recognize the small group of people. Multilayer Perceptron (MLP) Neural Network (NN) was studied for this task. The designed network had three layers. Input to the NN is a grayscale image and the number of input units is equal to the number of pixels in the image. Number of neurons in the hidden layer was 30 and number of output units is equal to the number of persons to be recognized. Each output unit is associated with one person. NN is trained so as to respond "+1" on output unit, for recognized person and "-1" on other outputs. After training process, highest output of NN indicates recognized person for test image.

P. Latha et al. [1] proposed a method that used Neural and PCA based algorithm for face recognition. Principal component analysis (PCA) transforms a number of possibly correlated variables into a smaller number of uncorrelated variables. These uncorrelated variables are called principal components.

Shamla Mantri and Kalpana Bapat [5] explored Self Organizing Map (SOM) method for face recognition. Here, integration of input image, Feature extraction, Training and Mapping was done. SOM is an artificial neural network. The cells of SOM become specially tuned to various input signal patterns or classes through an unsupervised learning process. In unsupervised training only the inputs are given to the neural network. Each neuron, separately decode the same input.
SOM has the following properties which makes it a good feature extractor

- Self-Organizing Maps are topologically ordered
- It is a competitive learning Artificial Neural Network
- Modifies selected nodes and its neighbors

E. Cabello et al. in [6] performed some experiments different Neural Network approaches on Face Recognition. The paper presented some results obtained using LVQ (learning vector quantization) and MLP (multilayer perceptron). Input fed to the Network for each case, were gray level images and geometrical features extracted from a set of 14 manually introduced landmarks. LVQ behaved better than MLP, showing lower error rates in the experiment. And, training times were much shorter for LVQ than for MLP. Also, MLP achieved lower error rates when dealing with geometrical features.

Omaima N. A. AL-Allaf et al. in [7], suggested a face recognition system on four Artificial Neural Network (ANN) models separately: feed forward backpropagation neural network (FFBPNN), cascade forward backpropagation neural network (CFBPNN), function fitting neural network (FitNet) and pattern recognition neural network (PatternNet). Each of the Network model was constructed separately with 7 layers i.e. One input layer, 5 hidden layers and one output layer. Each hidden layer was designed with 15 neurons. Six ANN training algorithms namely, TRAINLM, TRAINBFG, TRAINBR, TRAINCGF, TRAINGD, and TRAINGD were used to train each model separately. For each one of the four models based on 6 different training algorithms, experiments were conducted. Performance results of different models were compared based on the mean square error identifies the best ANN model. For the specified application, results showed that the PatternNet model was the best model used. Performances of different training algorithms were also performed. Comparison results showed that TrainLM was the best training algorithm for the face recognition system for the specified application.

III. ARTIFICIAL NEURAL NETWORK

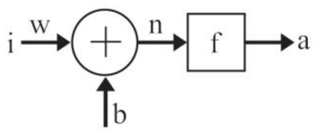

Figure 1. Simple neuron with single scalar input

'i' is the scalar input. It is transmitted through a connection that multiplies its strength by the scalar weight 'b'. The output so formed 'wp' and bias 'b', is the input to the activation function 'f' that produces the output 'a'. The bias has a constant input of 1. 'w' and 'b' are both adjustable scalar parameters of the neuron. A neural network adjusts its parameters so that the network exhibits some desired or interesting behavior. The network can be trained to do a particular job by adjusting the weight or bias parameters, or perhaps the network itself will adjust these parameters to achieve some desired end.

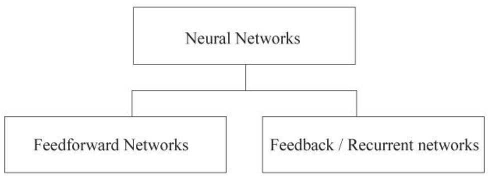

Figure 2. Taxonomy of Network Architectures

Backpropagation

The Backpropagation neural network is the most extensively used multilayered, feedforward neural network. The Backpropagation network has two stages, training and testing. During the training phase, the network is "shown" sample inputs and the correct classifications. The operations of the Backpropagation neural networks can be divided into two steps: feedforward and Backpropagation. In the feedforward step, an input pattern is applied to the input layer and its effect propagates, layer by layer, through the network until an output is produced. The network's actual output value is then compared to the expected output, and an error signal is computed for each of the output nodes. Since all the hidden nodes have, to some degree, contributed to the errors evident in the output layer, the output error signals are transmitted backwards from the output layer to each node in the hidden layer that immediately contributed to the output layer. This process is then repeated, layer by layer, until each node in the network has received an error signal that describes its relative contribution to the overall error. Once the error signal for each node has been determined, the errors are then used by the nodes to update the values for each connection weights until the network converges to a state that allows all the training patterns to be encoded. The Backpropagation algorithm looks for the minimum value of the error function in weight space using a technique called the delta rule or gradient descent. The weights that minimize the error function are then considered to be a solution to the learning problem.

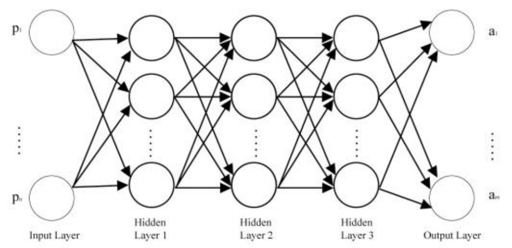

Figure 3. Feed Forward Backpropagation Neural Network with 5 layers

IV. PROPOSED WORK

Here Feedforward Backpropagation Neural Network was designed for the face recognition system constructed with 5 layers i.e. one input layer, 3 hidden layers and one output layer. Fig.1 shows 5-layer FFBPNN. Training algorithms adjusted the ANN weights and biases to minimize the performance function and to reduce errors. Mean Square Error (MSE) is the difference between the desired output and actual output. Here, MSE is used as a performance function it is minimized during ANN training.

In this research, the constructed ANN were trained with four ANN optimization training algorithms, Resilient Backpropagation training function (TRAINRP), Levenberg-Marquardt training function (TRAINLM), Variable Learning Rate Backpropagation training function (TRAINBFG) and BFGS Quasi-Newton training function (TRAINGDX) Performance of the Neural Network for classification was done. The various parameters assumed for different algorithm are as follows:

net.trainParam.epochs = 1 50,000
net.trainParam.goal = .0001
net.trainParam.lr = .888
net.trainParam.show = 200

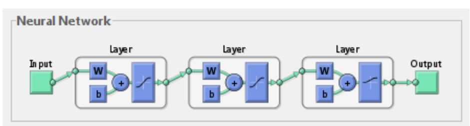

Figure 4. Feed Forward Backpropagation Neural Network with 5 Hidden layers

V. RESULTS AND DISCUSSION

A. *Resilient Backpropagation*

Multilayer networks typically use sigmoid transfer functions in the hidden layers. These functions are often called "squashing" functions, since they compress an infinite input range into a finite output range. Sigmoid functions are characterized by the fact that their slope must approach zero as the input gets large. This causes a problem when using steepest descent to train a multilayer network with sigmoid functions, since the gradient can have a very small magnitude; and therefore, cause small changes in the weights and biases, even though the weights and biases are far from their optimal valuesThe purpose of the resilient backpropagation (Rprop) training algorithm [10] is to eliminate these harmful effects of the magnitudes of the partial derivatives. Only the sign of the derivative is used to determine the direction of the weight update; the magnitude of the derivative has no effect on the weight update. The size of the weight change is determined by a separate update value. The update value for each weight and bias is increased by a factordelt_inc whenever the derivative of the performance function with respect to that weight has the same sign for two successive iterations. The update value is decreased by a factor delt_dec whenever the derivative with respect that weight changes sign from the previous iteration. If the derivative is zero, then the update value remains the same. Whenever the weights are oscillating the weight change will be reduced. If the weight continues to change in the same direction for several iterations, then the magnitude of the weight change will be increased.

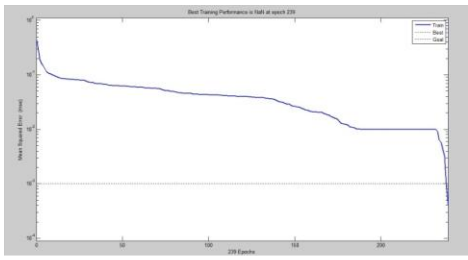

Figure 5. Performance plot using Resilient Backpropagation training function

The trainrp function is the fastest algorithm on pattern recognition problems. However, it does not perform well on function approximation problems. Its performance also degrades as the error goal is reduced. The memory requirements for this algorithm are relatively small in comparison to the other algorithms considered.

B. *Levenberg-Marquardt Training Function*

The Levenberg-Marquardt algorithm has the fastest convergence [11]. This advantage is especially noticeable if very accurate training is required. In many cases, trainlm is able to obtain lower mean square errors than any of the other algorithms tested. However, as the number of weights in the network increases, the advantage of trainlm decreases. The storage requirements of trainlm are larger than the other algorithms tested.

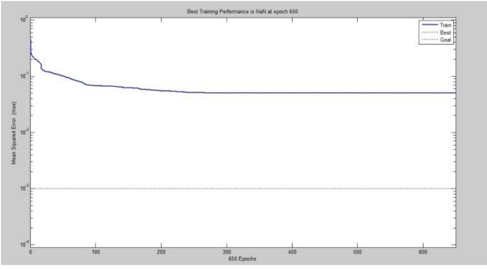

Figure 6. Performance plot using Levenberg-Marquardt training function

C. *Variable Learning Rate Backpropagation*

The variable learning rate algorithm traingdx is usually much slower than the other methods, and has about the same storage requirements as trainrp.

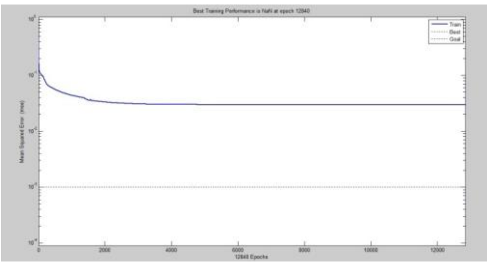

Figure 7. Performance plot using Variable Learning Rate Backpropagation training function

D. *Bfgs Quasi-Newton*

The performance of trainbfg is similar to that of trainlm. It does not require as much storage as trainlm, but the computation required does increase geometrically with the size of the network, because the equivalent of a matrix inverse must be computed for each of the iteration.

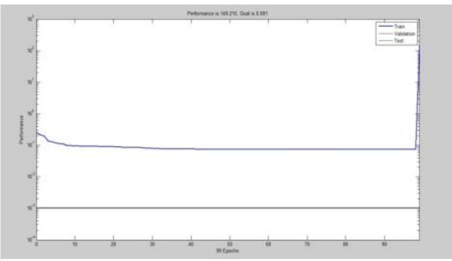

Figure 8. Performance plot using *BFGS Quasi-Newton training function*

Table I. Performance Comparison of Training Functions

No.	Training function	Acronym	Performance	Time (in sec)
a)	Levenberg-Marquardt	lm	0.000941	13
b)	Resilient Backpropagation	rp	0.0300	78
c)	BFGS Quasi-Newton	bfg	0.0500	16
d)	Variable Learning Rate Backpropagation	gdx	1611	0.04

VI. CONCLUSIONS

After analyzing the network using the different training functions following conclusions were made:

Table II. Convergence, MSE and Memory Requirement Comparison of Training Functions

Training Function	**Convergence**	**MSE**	**Memory Requirement**
lm	Fast	Low	Large
rp	Fast	Low	Low
bfg	Fast	High	Low
gdx	Slow	High	Low

In this research, four ANN optimization training algorithms (TRAINLM, TRAINRP, TRAINBFG and TRAINGDX) were used to train each of the constructed ANN models separately. The training samples of the suggested face recognition system were taken from The FEI Database of Faces [9]. As training samples100 face images (160×480). The results showed that the lowest values of MSE and number of iterations were resulted from the Levenberg Marquardt training algorithm (TRAINLM) in terms of memory, convergence and MSE for the application.

REFERENCES

[1] P.Latha, Dr. L Ganeshan, and Dr. S Annadurai,"Siganl Processing: An International Journal (SPIJ)",*Vol 3, Issue (5)*

[2] Anil K. Jain, Arun Ross and Salil Prabhakar, An Introduction to Biometric Recognition, *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics, , January 2004, Vol. 14, No. 1* [3] R. Brunelli and T. Poggio, "Face recognition: features versus templates", *IEEE Trans. on PAMI.- 1993. - Vol. 15. - P.1042-1052*

[4] V.V. Starovoitov, D.I Samal, D.V. Briliuk , "Three Approaches for Face Recognition*", 6-th International Conference on Pattern Recognition and Image Analysis, October 21-26, 2002, Velikiy Novgorod, Russia, pp. 707-711*

[5] Shamla Mantri and Kalpana Bapat, "Neural network Based Face recognition", *IJCSET, February 2011, Vol 1, Issue 1, 6-9*

[6] E. Cabello, A. Sánchez and L. Pastor. "Some experiments on Face Recognition with Neural Networks". *Face Recognition: From theory to applications. Springer-Verlag. Eds: H. Weeechsler et al. Pp: 589-599. 1998.*

[7] Omaima N. A. AL-Allaf, Abdelfatah Aref Tamimi and Mohammad A. Alia "Face Recognition System Based on Different Artificial Neural Networks Models and Training Algorithms", *(IJACSA) International Journal of Advanced Computer Science and Applications,* Vol. 4, No. 6, 2013

[8] Guoqiang Peter Zhang, "Neural Networks for Classification: A Survey", *IEEE Transactions on Systems, Man, And Cybernetics—Part C: Applications And Reviews*, Vol. 30, No. 4, November 2000

[9] http://fei.edu.br/~cet/facedatabase.html

[10] C. Igel and M. Husken, "Empirical evaluation of the improved RPROP learning algorithms", *Neurocomputing*, vol. 50, pp. 105–123, 2003.

[11] R.HariKumar, N.S.Vasanthi, M.Balasubramani "Performance Analysis of Artificial Neural Networks and Statistical Methods in Classification of Oral and Breast Cancer Stages ", *International Journal of Soft Computing and Engineering (IJSCE)* ISSN: 2231-2307, Volume 2, Issue 3, July 2012