A Study on Neural Network Training Algorithm for Multiface Detection in Static Images

Zulhadi Zakaria, Nor Ashidi Mat Isa and Shahrel A. Suandi, Member IEEE

School of Electrical & Electronic Engineering
Universiti Sains Malaysia, 14300 Nibong Tebal, Seberang Prai Selatan
Pulau Pinang, Malaysia
e-mail: n4smy@yahoo.com

***Abstract*—** This paper reports the study results on neural network training algorithm of numerical optimization techniques multiface detection in static images. The training algorithms involved are scale gradient conjugate backpropagation, conjugate gradient backpropagation with Polak-Riebre updates, conjugate gradient backpropagation with Fletcher-Reeves updates, one secant backpropagation and resilent backpropagation. The final result of each training algorithms for multiface detection application will also be discussed and compared.

***Keywords*—**training algorithm, multiface, static image, neural network

I. Introduction

THIS paper is part of a case study to apply artificial neural networks (ANN) for face detection in static images. The Combining Skin Color and Neural Network for Multiface Detection in Static Images was explain in our previous paper [1]. Now the performance (speed processing and high accuracy result) of training algorithm that been used in that Detection system is our research target. Fig. 1 shows the system architecture of Multiface Detection System for Static Image.

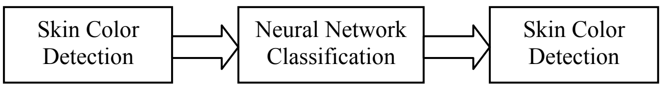

Fig. 1 Multiface System Detection Block Diagram

Manuscript received October 10, 2009. This work was supported in part by the Universiti Sains Malaysia Short Term Research Grant 304/PELECT/6035270.

Zulhadi Zakaria is with the School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, Nibong Tebal, 14300 Pulau Pinang, Malaysia. (corresponding author phone: +6012-4489740; fax: +604-5389266; e-mail: n4smy@yahoo.com).

Nor Ashidi Mat Isa is with the School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, Nibong Tebal, 14300 Pulau Pinang, Malaysia. (email: ashidi@eng.usm.my).

Shahrel A. Suandi is with the School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, Nibong Tebal, 14300 Pulau Pinang, Malaysia. (e-mail: shahrel@eng.usm.my).

II. System Architecture

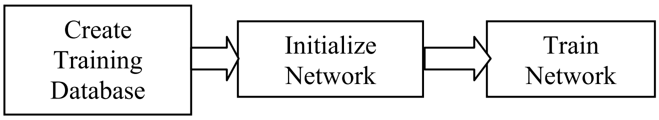

Fig. 2 System Flow Neural Network Classification Model

Referring to Fig. 2 above, system implementation begins with creating training database. Then, neural network model such as training function, architecture and parameter were initialized.

A. Training Database

75 images as face and 55 images as non-images is used as training images. Fig. 3 shows the faces images database and Fig. 4 shows non-faces images.

Fig. 3 Face training dataset

Fig. 4 Non-Face training dataset

B. Initialize Neural Network Layer

In our neural network model, 3 layers (2160 input, 100 hidden, 1 output). Fig. 5 below shows the neural network architecture of feed forward neural network with one hundred neurons in the hidden layer and one neuron in the output layer for classification task.

C. Train Network : Neural Network Training algorithm

Backpropagation is widely used to solve many classification problems by using the concept of Multilayer Perceptron (MLP) training and testing. However, the major disadvantages of BP are its convergence rate relatively slow [2] and being trapped at the local minima. But there are many solutions proposed by many neural network researchers to overcome the slow converge rate problem.

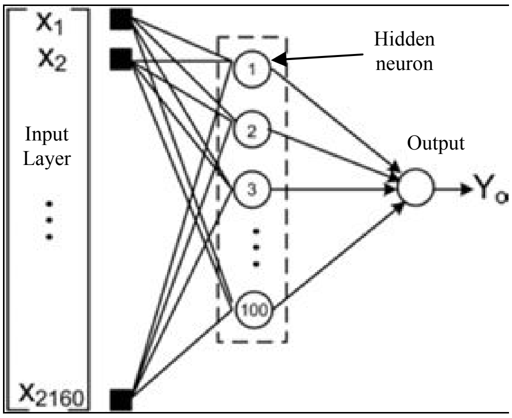

Fig. 5 Neural network architecture

Therefore, many powerful optimization algorithms have been devised, most of which have been based on simple gradient descent algorithm as explain by C.M. Bishop [3] such as conjugate gradient decent, scaled conjugate gradient descent, quasi-Newton BFGS and Levenberg-Marquardt methods.

Next section, the training algorithm that is use in Matlab to implement in Multiface Detection System is described.

D. Training Algorithms in MATLAB

In the Neural Network model by using MATLAB, there were several training algorithms which have a variety of different computation and storage requirements. However, no one algorithm is best suited to all application. In our works, we try to implement our system by using a few Conjugate Gradient Algorithms, Quasi-Newton Algorithms and Heuristics Algorithms.

Numerical optimization techniques for neural network training include three types such as Conjugate gradient, Quasi-Newton and Levenberg-Marquardt. However in this paper, the conjugate gradient and Quasi-Newton only is used.

III. System Parameters

System was developed by using Matlab software and SCG training parameters are determines such like epochs, show, goal, time, min_grad, max_fail, sigma, and lambda. The parameter for learning rate is between 0.3 and 0.6, the epochs is 400, the error goal is about $1e^{-3}$. Sigma parameter is used to determine changes in weight for second derivative approach. The SCG routine requires more iteration to converge than the other conjugate gradient algorithms, but the number of computations in each iteration is significantly reduced because no line search is performed [4].

To determine changes in weight and bias, learning rate lr multiplied by the negative slope If the level of learning is too large, then the algorithm will become unstable while on the contrary, algorithms take a long time to Converge [4].

Based on the Multifaces detection problems in Matlab, learning rate between 0.3 to 0.5 is used to determine the length of the weight update (step size) which is adjusted at each iteration. A search is made along the conjugate gradient direction to determine the step size, which minimizes the performance function along that line.

IV. Conjugate Gradient Algorithms

Conjugate gradient is the most popular iterative method for solving large systems of linear equations [5]. In the first iteration usually the conjugate gradient algorithm will find the steep descent direction. This paper will describe 3 types of conjugate Gradient Algorithms Such as Scaled Gradient Conjugate Backpropogation(SCG), conjugate gradient backpropagation with Polak-Riebre Updates(CGP), conjugate gradient backpropagation with Fletcher-Reeves updates(CGF).

Approximate solution, x_k for conjugate gradient iteration is described as formulas below [6]:

$$x_k = x_{k-1} + \alpha_k d_{k-1} \quad (1)$$

k will always be the iteration index, αk is the length of the step preformed at iteration k, dk is search direction, rk is residual vector and βk is improvement.Formula (2),(3),(4),(5) shows the relative component of approximate solutionfor conjugate gradient.

$$\alpha k = (r^T{}_{k-1} r_{k-1}) / (d^T{}_{k-1} A d_{k-1}) \quad (2)$$

$$d_k = r_k + \beta_k d_{k-1} \quad (3)$$

$$r_k = r_{k-1} - \alpha_k A d_{k-1} \quad (4)$$

$$\beta_k = (r^T{}_k r_k) / (r^T{}_k - 1 r_{k-1}) \quad (5)$$

A. Scaled Gradient Conjugate Backpropogation(SCG)

SCG is a second order Conjugate Gradient Algorithm that help minimize goal functions of several variables. This theoretical foundations was prove by Moller[6] which remains first order techniques in first derivatives like standard backpropogation and find the better way to a local minimum in second order techniques in second derivatives.

SCG use a step size scaling mechanism avoids a time consuming line-search per learning iteration, which makes the algorithm faster than other second order algorithms recently proposed. Base on the Moller[6], SCG methods shows superlinear convergence on most problems.

B. Conjugate Gradient Backpropagation with Fletcher-Reeves Updates (CGF)

Second version of the conjugate gradient algorithm was proposed by Fletcher-Reeves. As with the Polak and Ribiére algorithm, the search direction at each iteration is computed by equation (6) below.

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \qquad (6)$$

C. Conjugate Gradient Backpropagation with Polak-Riebre Updates (CGP)

Another version of the conjugate gradient algorithm was proposed by Polak and Ribiére. The search direction at each iteration is same like SCG search direction equation. However for the Polak-Ribiére update, the constant beta, βk is computed by equation (7) below.

$$\beta_k = \frac{\Delta g_{k-1}^T g_k}{g_{k-1}^T g_{k-1}} \qquad (7)$$

V. Other Algorithms

A. Quasi-Newton Algorithms (One-Step Secant Backpropagation (OSS))

An alternative way to speed up the conjugate gradient optimization is Newton's method. The basic step of Newton's method shows in equation (8) below.

$$x_{k-1} = x_k - A_k^{-1} g_k \qquad (8)$$

A_k is the Hessian matrix of the performance index at the current values of the weights and biases. Refer to [4] Newton's method converges faster than conjugate gradient methods.

BFGS algorithm requires much more time for storage and computation compared with OSS algorithms. However, conjugate gradient algorithms require less storage and computation per epoch compared with the OSS algorithm. It can be considered a compromise between full quasi-Newton algorithms and conjugate gradient algorithms [4].

B. Heuristics Algorithms (Resilent Backpropagation(RP))

The purpose of the resilient backpropagation training algorithm is to eliminate these harmful effects of the magnitudes of the partial derivatives [4].

VI. Experiments Result

Depending on the number of active neurons, the models usually took 3 to 100 seconds to be fitted using dual 1MHz PC-compatible computers. MATLAB software is used to perform calculations. In the five training algorithms experiments, 40 random test images from Caltech [7], Georgia [8], UCD Color Database [9] and Yahoo database are tested and the results are shown in Table 1 below. The best performance from this experiment suggests the appropriate training algorithm.

TABLE I SUMMARY OF USING FIVE DIFFERENT TRAINING ALGORITHMS IN NEURAL NETWORK [1]

Training Algorithm	Criteria	Total of images: Single 20	2 Faces 10	3 Faces 10
Conjugate gradient backpropagation with Fletcher-Reeves updates	Accuracy	90%	70%	50%
	Processing Time(s)	924.477	202.814	339.249
	False-positive answer	0%	0%	0%
	False-negative answer	0%	0%	0%
Conjugate gradient backpropagation with Polak-Ribiere updates	Accuracy	90%	75%	60%
	Processing Time(s)	938.140	229.966	354.798
	False-positive answer	0%	0%	0%
	False-negative answer	0%	0%	0%
Resilient backpropagation	Accuracy	95%	70%	53%
	Processing Time(s)	910.778	198.281	317.415
	False-positive answer	0%	0%	0%
	False-negative answer	10%	0%	0%
One step secant backpropagation	Accuracy	85%	70%	63%
	Processing Time(s)	927.735	227.984	372.480
	False-positive answer	0%	0%	0%
	False-negative answer	10%	0%	0%
Scaled conjugate gradient backpropagation	Accuracy	100%	75%	63%
	Processing Time(s)	889.671	200.109	345.404
	False-positive answer	0%	0%	0%
	False-negative answer	0%	0%	0%

From Table 1, scaled conjugate gradient backpropagation is found to be the best performance compare with other algorithms in accuracy and processing time aspects. As a result, scaled conjugated gradient backpropagation is used as the training algorithm for proposed system.

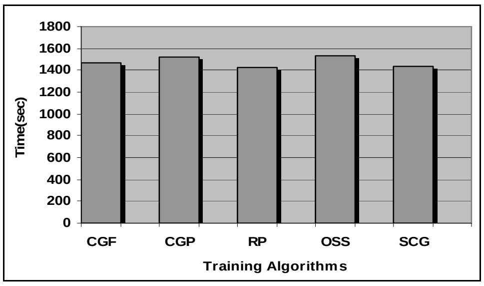

Fig. 6 Comparative of training algorithms processing time(s) classification

Fig. 6 shows, that RP and SCG has better results on processing time. However, OSS converges and reduces the error with maximum iterations.

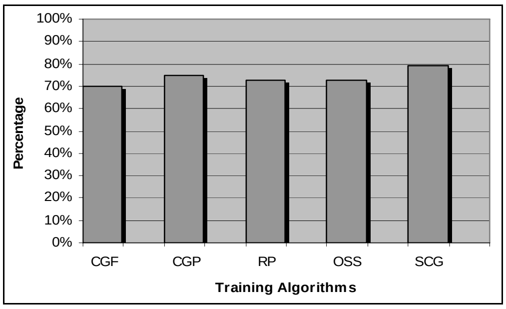

Fig. 7 Comparative of training algorithms accuracy(s) classification

Fig. 7 shows, that SCG has better results correct classification percentage. SCG converges in a short time with high correct classification percentage. But in terms of convergence time, it shows that RP is better than SCG, and RP significantly reduces the error with minimum iterations.

For overall performance, the experiments show that SCG produces feasible results in terms of convergence time and classification percentage.

Based on the Fig.8 below, the SCG routine can require more iteration to converge than the other conjugate gradient algorithms, but the number of computations in each iteration is significantly reduced because no line search is performed.

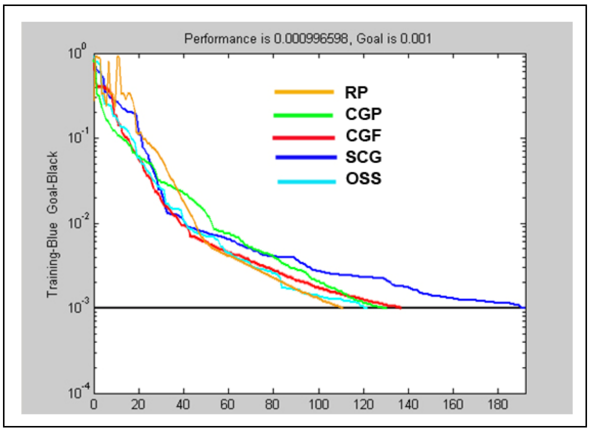

Fig. 8 Comparison of Convergence between five training algorithm

VII. Data analysis

All tested Artificial Neural Network (ANN) were feed-forward networks in which the neuronal signals were processed by a nonlinear hidden layer of units, which used the so-called tan-sigmoid output function [10] that fed into a linear output layer that predicted the position signals. There was one output for ANNs used for estimating position of multiface in static images.

Several different algorithms for training the networks were evaluated : classical backpropagation with variable learning rates, scale gradient conjugate, conjugate gradient with Polak-Riebre Updates, conjugate gradient with Fletcher-Reeves updates, one secant and resilent were tested as well, but were found to require too much RAM memory to allow PC-compatible computers to be used efficiently. Empirically, it was found that the fastest fitting of the models and the best predictions of face position were obtained using one hidden layer with 100 units. The Gradient Descent with Adaptive Learning rate as out training algorithm was test but the data was over-fitting. However Levenberg-Marquardt training algorithm cannot be done because the network is very large and run out of memory after we run using MATLAB.

VIII. Discussion and Conclusions

The conjugate gradient algorithms, in particular trainscg, seem to perform well over a wide variety of problems, particularly for networks with a large number of weights. The SCG algorithm is almost as fast as the LM algorithm on function approximation problems (faster for large networks) and is almost as fast as RP on pattern recognition problems. Its performance does not degrade as quickly as RP performance does when the error is reduced. The conjugate gradient algorithms have relatively modest memory requirements [4].

The SCG Method avoids a time consuming line-search per learning iteration, which takes the algorithm faster than other second order Conjugate Gradient algorithms, Quasi-Newton algorithms and Heuristics algorithms for multiface detection system in static images. This work has shown good results, with the SCG technique, using neural network model architecture. However, in a future work, more samples to train the classifiers are considered to use.

The RP 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.

Acknowledgement

This work is fully supported by Universiti Sains Malaysia Short Term Grant (304/PELECT/6035270).

References

[1] Zulhadi Zakaria, Nor Ashidi Mat Isa, and Shahrel A. Suandi, "Combining Skin Color and Neural Network for Multiface Detection in Static Images," Symposium on Progress in Information & Communication Technology, Kuala Lumpur, pp. 147-154, 2009.

[2] Zweiri, Y.H., Whidborne, J.F. and Sceviratne, L.D. A Three-term Backpropagation Algorithm. Neurocomputing. 50: 305-318. 2002.

[3] 10. C.M. Bishop. Neural Networks for Pattern Recognition. Chapter 7, pp.253-294. Oxford University Press. 1995.

[4] Mathworks Online. Available: http://www. www.mathworks.com, 2009.

[5] Jonathan Richard Shewchuk, An Introduction to the Conjugate Gradient Method Without the Agonizing Pain, August 1994.

[6] Martin Fodslette Moller. A Scaled Conjugate Gradient Algorithm For Fast Supervised Learning. Neural Networks, 6:525-533, 1993.

[7] Caltech Image Database. California Institute of Technology. Available: http://www.vision.caltech.edu/Image_Datasets/faces/ faces.tar, 2007.

[8] Georgia Face Image Database. Georgia Institute of Technology. Available: http://www.anefian.com/gt_db.zip, 2007.

[9] UCD Color Face Image Database. College of Engineering, Mathematics and Physical Sciences UCD, Dublin. Available: http://ee.ucd.ie/~prag/. March, 2007.

[10] Demuth, H., Beale, M. Neural Network Toolbox (Mathworks, Natick, Massachusetts, 1998).