Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 70 (2014) 24413-24416

ISSN: 2229-712X

Comparative Analysis of Ann (Artificial Neural Network) To Identify Olive Ridley Sea Turtle

S. Santhosh Baboo and A.R.J.Vigneswari
D.G.Vaishnav College, Arumbakam, Chennai, India.

ARTICLE INFO

Article history:
Received: 17 April 2014;
Received in revised form:
16 May 2014;
Accepted: 26 May 2014;

Keywords
Artificial Neural network ,
Feedforwrad network ,
Perceptron.

ABSTRACT

The present study deals with the performance and comparison between two architecture in neural networks namely perceptron neural networks and feed forward neural networks. This networks help to identify the particular species of olive ridley sea turtle. The existing algorithm in training the images faces many difficulties such as time delay in training the images and also improper learning rate in the ANN. To overcome this problems the new algorithm is developed were training is made with both perceptron and feed forward neural networks. Initially 8 images were trained and these images consisting of four main features which includes length, breadth, color and shape. These trained images are processed for performance analysis to make effective identification of olive ridley sea turtle.

Introduction

The main objective of the study is to make an analysis to conserve the olive ridley sea turtle which is one among the endangered species all over the world. This image classification is done through ANN training as seen in "Performance comparison on face recognition system using different variants of back propagation algorithm with radial basis function neural networks" [1] were two different neural networks is used for training the same dataset of face recognition in this algorithm has its own difficulties, where the poor dataset of images will not be trained in these kind of methodology. Also for training a dataset with two other classifier are used for the image classification for better classification and object recognition of images is observed in" Based on perceptron object classification algorithms for processing of agricultural fields images" [2] where multilayer perceptron classification is done to classify the agricultural fields to deduct the disease in the plants, where the deduction and land cover classification is difficult in this algorithm.

In this paper new algorithm is implemented where the feature extracted image of olive ridley is taken for training in ANN. The training is done with different neural networks namely percepton and fast feed forward neural network in which comparative analysis is made to measure the performance of the training among the networks. The result analysis curve of the perceptron neural network clearly shows the improved training of images as compared with the training graph of the fast feed forward neural network.

Methodology

The algorithm used is to make a comparative analysis is made between two training neural networks namely perceptron and feed forward neural networks where eight images is trained and best training network is taken in to consideration for identifying the olive ridley sea turtle images. The process of the neural network training are enlighten below.

The figure 1 represents the image extraction of olive ridley sea turtle using ANN networks were the input images of olive ridley sea turtle is taken for training where initially eight images are trained in which feature is extracted from these images. Then these images are given input in both neural networks for training and the performance is measured among these networks this technique is implemented to make best training performance to identify the sea turtles.

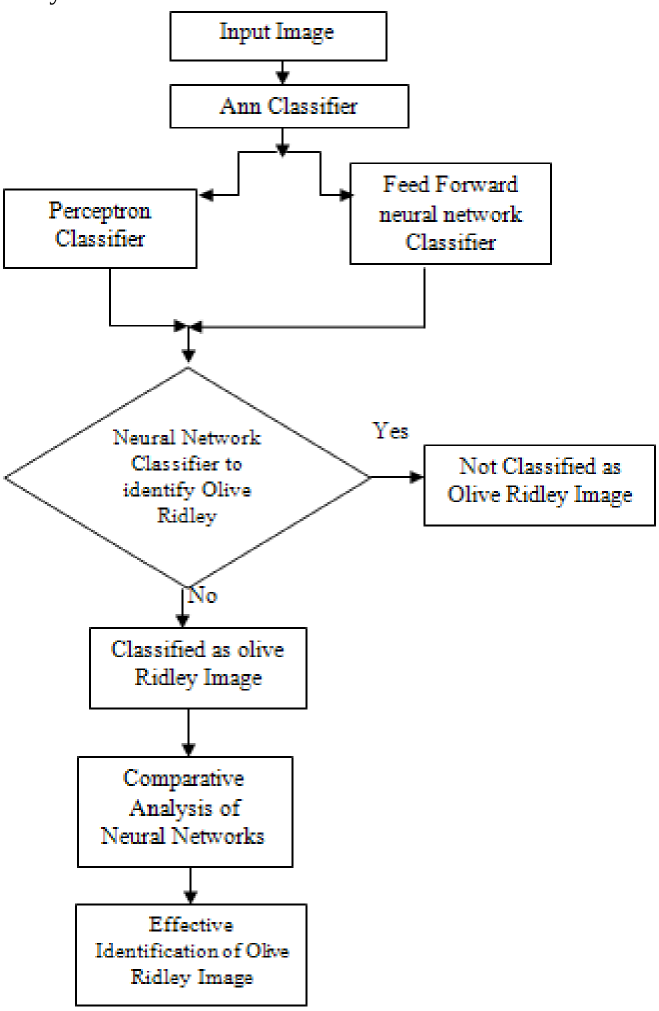

Fig 1: Image Extraction of olive ridley image using ANN networks

Tele:
E-mail addresses: vigneswarijothi@yahoo.com

ANN Classifier

The artificial neural network used for training of images, where the ANN classifier follows some features which are given below

- The signal that are received as an input for processing the elements where the elements sums with the weighted inputs to receive the efficient output.
- The weight at the output end of the neural network has the capacity to modify the incoming elements.
- The neuron can able to transmit the output when enough input is acquired.
- The output which is produced from the neural network can be transmitted to other neurons.
- The weights that are given as input neurons can be changed at any time.
- The entire artificial neural network must fallow the fault tolerance.

Perceptron algorithm for training the images

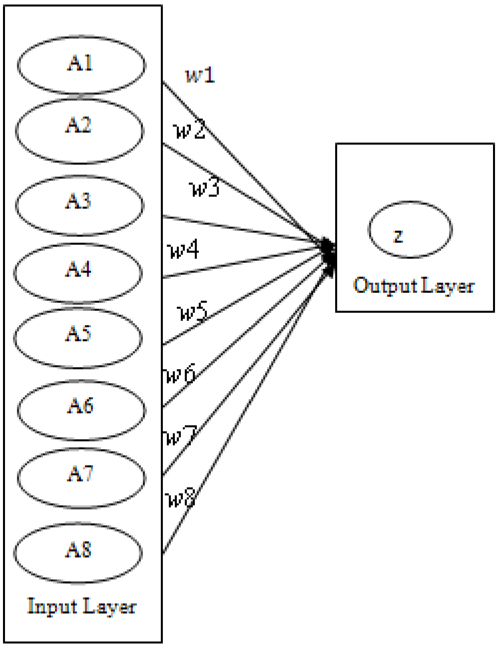

Fig 2: Architecture of single layer of the perceptron

The perceprton image training, algorithm initially weights and then the bias are set as zero. Then The threshold value range must be set for learning rate parameter these ranges has been listed in the table1 and the truth table for the images ranges is also explained in table 2.In this training the net input is calculated by multiplying the weights with the inputs and adding the result with the bias entity. Once the net input is calculated, by applying the activation function the output of the network is obtained. This output is compared with the target, where if any difference occurs, then the training has to go for the weight updating based on perceprton learning rule, else the network training is stopped. This training uses the bipolar target with fixed threshold and adjustable bias.

The figure 2 shows the basic structure of perceptron neural network where the A1,A2,A3 ,A4,A5,A6,A7 and A8 are the threshold values of the olive ridley images that are briefly explained in table1 and table2 , these valued must be given as input for the perceptron layer. The weights w1, w2, w3, w4, w5, w6, w7 and w8 are the feature that is extracted from the input images that must be initialized in the training of the networks. The Z in the structure represents the output trained images of the olive ridley images.

Table 1: Threshold values of Images

Images	Color RGB color from	Length and Breadth	Shape
Image 1	0.608,0.60,0.608	+1.0412 , 1.0085 1.6847 , +1.5429	Hexagonal
Image 2	0.608,0.60,0.608	+1.0059 , +1.01070.99526 , +1.0038	Hexagonal
Image 3	0.608,0.60,0.608	+1.0965 , 1.0255 0.44992 , 0.46143	Hexagonal
Image 4	0.608,0.60,0.608	0.99625 ,1.01290.77043 , 0.67395	Hexagonal
Image 5	0.608,0.60,0.608	1.0059 , 1.01070.99526 , 1.0038	Hexagonal
Image 6	0.608,0.60,0.608	1.0059, 1.01070.99526 , 1.0038	Hexagonal
Image 7	0.608,0.60,0.608	+1.0965 , 1.0255 0.44992 , 0.46143	Hexagonal
Image 8	0.608,0.60,0.608	1.0 412 , 1.0085 1.6847 , +1.5429	Hexagonal

The table 1 which explains the threshold values of the images ranges of olive ridley. Initially eight images are trained and when the images ranges obtain the threshold values it is considered to be an olive ridley

Table 2: Truth Tables for Images

Images	Color	Length	Breath	Shape
Image 1	1	0	0	1
Image 2	1	1	1	1
Image 3	1	0	1	1
Image 4	1	1	0	1
Image 5	1	0	0	1
Image 6	1	1	0	1
Image 7	1	1	1	1
Image 8	1	1	1	1

The table 2 is a truth table values for the perceptron neural networks. The dataset value will be saved as a variable in matlab and with these combinations the images will be trained and the result of the trained in the perceptron neural network and the result curve is compared with the feed forward neural networks. That is when the o's value given in color parameter which means that given image does not reach the threshold value and 1's value in truth table which represent the threshold value of the image reaches the range given in the above table

Step1: Initialize the weights and bias (initially it can be zero). Set learning rate (0 to 1)

Step 2: while stopping condition is false do steps 3 and 7

Step 3: For each training pair s: t does steps 4-6

Step 4: Set activations of input units

$$x_j = s_j \; for \; i = 1 to n$$

Step 5: compute the input unit response

$$y_{in} = b + \sum_i x_i + w_i$$

The activation function used is,

$$y = f(y_{in}) = \begin{cases} 1, if \\ 0, if \; -\sigma \le y_{-in} \le \sigma \\ -1, if \; y_{-in} \end{cases}$$

Step 6: The weights and bias are updated if the target is not equal to the output response.

If $t \neq y$ and the value of x_i is not zero

$$w_{i(new)} = w_{i(old)} + \alpha t \, x_i$$

$$b_{(new)} = b_{(old)}$$

Step 7: Test for stopping condition.
Fig 3: Perceptron Training Algorithm

The figure 3 which explains the steps that must be followed in the perceptron structure, where initialize the weights bias and α (training state) which determines the magnitude of the weights. Initialize the specific tolerance error by activation of the input units in which the x_i is the of the liner neurons that must be equal to the s_j output layer of the perceptron that ranges from 1 to n. Then compute the input unit response where y_i implements the directive of activation function this is to specify the tolerance error, it is calculated by summing up the bias with the activated input units and the weights that is specified in the network. In the same step when the activations are applied over the net input to calculate the output response. When the weights and bias are updated then it will check the condition whether the target is equal to the output response, so when the condition is not satisfied then the make the adjustment in the bias and weights for just J=1 to m and i= 1 to n. If the heights weight change that occurred during training is smaller than the specified tolerance then the stop the training else continue the process.

Feed Forward Neural network

The feed forward networks are the single layer of weights where the input images that is directly connected with the output layers with the intervening sets of hidden units. These networks use the hidden units to create input patterns. The table 1 and table two which are described above where the input threshold values is given as an input in network where eight images are trained to make the performance analysis between the networks.

Comparative analysis of neural networks

The comparative analysis is made in between perceptron and feed forward neural networks to measure the performance in training the input images also the training time is measured between the networks to make an effective identification of olive ridley sea turtle images.

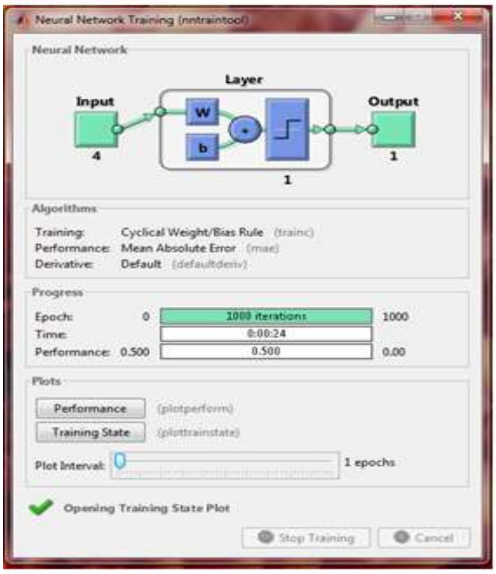

Fig 4: Training and Testing of Perceptron

The figure 4 and figure 5 is the testing and training of perceptron and fast feed forward neural network where the eight images are trained in both the networks in matlab.

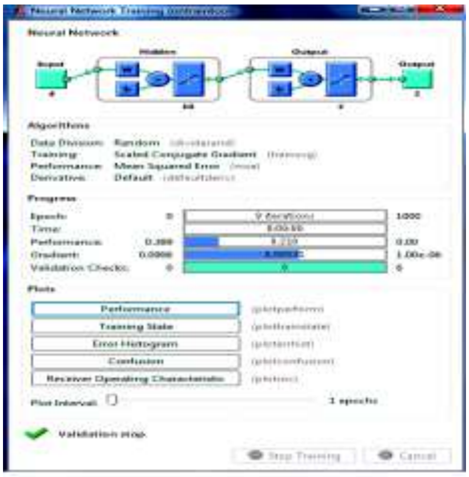

Fig 5: Fast feed forward testing and Training

In this training 4 input value is inserted in which feature extracted threshold values of olive ridley sea turtle image is inserted in the input neurons. Thus the performance iterations will be 1000 and time taken for the epochs training will be 0.00.24 in persptron and 0.500 will be the performance for the perceptron neural networks. But in the fast feed forward neural network there will be only 9 iterations for 1000 epochs and performance and time taken for training is comparatively high as compared (0.389,0.00) with the perceptron training

Fig 6: perceptron training comparisons curve

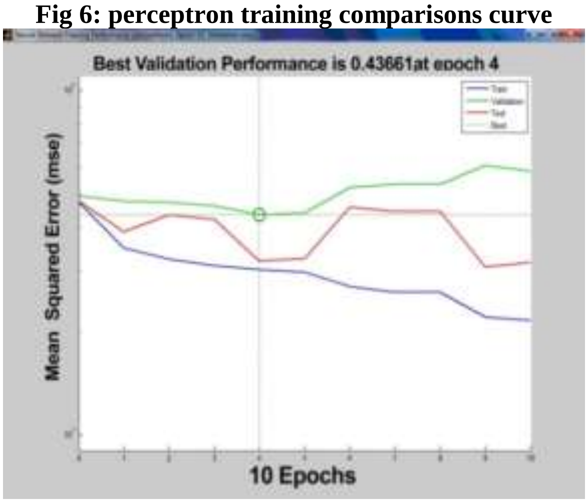

Fig 7: Mean square error Vs Epochs Graph

The figure 6 and figure 7 shows the training comparative curve which clearly states that the perceptron model for 10 epochs the mean square error is reduced as compared with feed forward artificial neural network from 0.6 and in perceptron 6 epochs is reduced to zero error rate as compared with the feed forward network. From this comparison it clearly states that the perceptron model is more efficient in training the threshold values where the error rate and the time are reduced.

Figure 8: Best Training Performance of Perceptron (MAE Vs Epochs)

The figure 8 the best training performance of perceptron is plotted as graph and from this analysis it shows that mean absolute error is reduced at the level of 10^{-6} with 0.6 performance rate and the error rate is reduced zero . From this analysis it clearly stated that the performance training of the perceptron is high as compared with the feed forward model neural network. This validation among the both perceptron model and the feed forward neural network model is explained in the table 3.

Table 3: Comparative analysis of Persptron and Feed Forward model

ANN Training Function	Learning Performance Rate	Training Time
Perceptron Model	0.005	0.0024
Feed Forward Model	0.389	0.1

Table 3 which represent the performance rate of both perceptron and feed forward neural network function for identifying olive ridley sea turtle in which it proves that perceptron is faster in learning the images as compared with feedforward networks.

Conclusion

In this paper the identification of olive ridley sea turtle for which 8 feature extracted images are trained using two different training methods of neural network namely perceptron neural networks and feed forward neural network in which the validation and testing is made among the networks to evaluate the performance and time measurement in training the images. In this validation perceptron gives the better generalization performance in comparison with the feed forward method. Both these networks depend upon the bias and the weight of the neurons. However this system can be further developed by training the images in fuzzy logic so the identification of specific images can be made more efficiently as compared with other neural networks.

References

[1] kiran arya and Virendra , P. Vishwakarma "Performance comparison on face recognition system using different variants of back propagation algorithm with radial basis function neural networks "Vol. 3, Issue 5, Sep-Oct 2013, pp.1588-1595,Department of computer science and engineering , Inderprastha Engineering College , Ghaziabad 201010, UP, India.

[2] Valentin Ganchenko, Alexander Doudkin , Tadeusz Pawlowski , Albert Petrovsky , Rauf Sadykhov , "Based on perceptron object classification algorithms for processing of agricultural fields images", United Institute of Informatics Problems of National Academy of Science of Belarus, Minsk, Belarus, ganchenko, Industrial Institute of Agricultural Engineering, Poznan, Poland, tadek, Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus

[3] L. Zhou *, X. Yang , "Use of Neural Networks for Land Cover Classification from Remotely Sensed imagery" Department of Geography, Florida State University, 113 Collegiate Loop, Tallahassee, FL 32306, USA.

[4] N. W. Campbwll, B. T. Thomas "Automatic Segmentation and Classification of Outdoor Images Using Neural Networks" Advanced Computing Research Centre, University of Bristol, UK.

[5] Xavier Glorot, Yoshua Bengio ,"Understanding the difficulty of training deep feedforward neural networks"DIRO, Universit´e de Montr´eal, Montr´eal, Qu´ebec, Canada.

[6] Robert P.W. Duin, Dick de Ridder , "Neural network experiences between perceptrons and support vectors "Pattern Recognition Group, Faculty of Applied Physics, Delft University of Technology, The Netherlands