FACE DETECTION AND RECOGNITION USING MORPHOLOGICAL SHARED-WEIGHT NEURAL NETWORK

Miss.Triveni K. Patil[1], Prof. Balasaheb T. Salokhe[2]

[1]Master of Electronics, Department of Electronics Tatyasaheb Kore Institute of Engineering & Technology, Warananagar, M.S., (India)

[2]Asso. Prof. Department of Electronics, Tatyasaheb Kore Institute of Engineering& Technology, Warananaga, M.S., (India)

ABSTRACT

An algorithm based on the morphological shared-weight neural network is introduced here. Being nonlinear and translation-invariant, the MSNN can be used to create better generalization during face recognition. As electronic recording system generates securities in many sectors with the help of biological parameters present in human body like face, Finger, Hand etc. In this project MATLAB based algorithm is implemented for human face recognition system using two biological parameters first method includes detection and recognition using whole face while other method includes eye level based detection and Recognition.

Keywords— Face Recognition, Hidden Layer, Neural Networks, Morphological Operations, Structuring Element.

I. INTRODUCTION

Face detection and localization is usually the first step in many biometric applications like, face recognition, video surveillance, human computer interface etc. Face detection's research has rapidly expanded by engineers and also neuroscientists, since it has found that it have many applications in computer vision communication and automatic access control system. Biometric-based technologies include identification based on physiological characteristics (such as face, fingerprints, finger geometry, hand geometry, hand veins, palm,iris, retina, ear and voice) and behavioral traits (such as gait, signature and eye stroke dynamics). Face recognition appears to offer several advantages over other biometric methods, a few of which are outlined here: Almost all these technologies require some voluntary action by the user, i.e., the user needs to place his hand on a hand-rest for fingerprinting or hand geometry detection and has to stand in a fixed position in front of a camera for iris or retina identification. However, face recognition can be done passively without any explicit action or participation on the part of the user since face images can be acquired from a distance by a camera. This is particularly beneficial for security and surveillance purposes. The system we are describing here uses a shared-weight morphological neural network (MSNN) in its face detection phase [2]. MSNNs have been used in many automatic target recognition applications and have shown to be un-affected by variations in lighting level and robust enough to handle some variation in target size and/or orientation. The main characteristics of neural networks are that they have the ability to learn complex nonlinear input-output relationships, use sequential training procedures, and adapt themselves to the data.

II. NEURAL NETWORK

Neural network [6] techniques are powerful tools that prove their efficiency in real-world applications, where problems are badly defined or difficult to formalize. Some applications, especially those involving images, require a huge number of operations and an enormous reduction of the dataflow from input to output data. For sorting objects by vision or image analysis, for example, inputs are images at video rate; outputs are correcting values, names of objects, or locations of specific objects. The main characteristics of neural networks are that they have the ability to learn complex nonlinear input-output relationships, use sequential training procedures, and adapt themselves to the data. The most commonly used family of neural networks for pattern classification tasks is the feed-forward network, which includes multilayer perceptron networks. Another popular network is the Self-Organizing Map (SOM), or Kohonen-Network, which is mainly used for data clustering and feature mapping. The learning process involves updating network architecture and connection weights so that a network can efficiently perform a specific classification/clustering task. A neural network is a massively parallel distributed processor that has a natural prosperity for storing experimental knowledge. It resembles the brain in two respects: [7]

i. Knowledge is acquired by the network through a learning process;

ii. Inter-connected connection strengths known as synaptic weights are used to store the knowledge; each neuron has an internal state called its threshold or activation function (or transfer function) used for classifying vectors.

Neural classification generally comprises of four steps:

i. Pre-processing, e.g., atmospheric correction, noise suppression, band rationing, Principal Component Analysis, etc;
ii. Training - selection of the particular features which best describe the pattern;
iii. Decision - choice of suitable method for comparing the image patterns with the target patterns;
iv. Assessing the accuracy of the classification.

III. SYSTEMV DEVELOPMENT

MSNN stands for morphological shared weight neural network. This type of architecture was proposed by Won [1]. It consists of two networks which are joined to each other. The first phase of the network is used for feature extraction from the image and next network is used for classification. During training, input images given to the MSNN. The feature extraction is achieved with morphological hit miss transform. Then these features are applied to the neural network. The weights of neural network are saved. During testing, the testing image again goes through MSNN. The output of MSNN is depending on training. During training, input images given to the MSNN. The feature extraction is achieved with morphological hit miss transform. Then these features are applied to the neural network. The weights of neural network are saved. During testing, the testing image again goes through MSNN. The output of MSNN is depending on training.

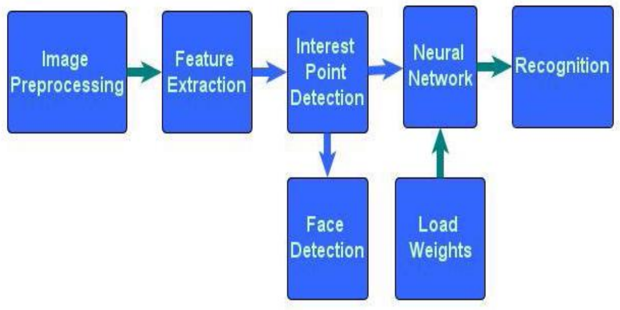

Figure 1: Face Detection and Recognition System Overview.

2.1 Neural Network Modeling

Training a network [10] is not an easy task. Each choice of any parameter will affect the other. An appropriate choice of learning parameters should guarantee that a good quality solution is found within a reasonable period of computing time. Generally, it is important to pay attention to a following few aspects:

a. Weight Initialization [11]

Initialization strongly affects the classifying solution. There are now many new methods of doing it using statistics. It is necessary to reset the weights if an unsuccessful training occurs. A good convergence is also determined by the values of weights that are initialized: that is, whether they are randomly initialized or zero-initialized. The choice of random or zero weights for the hidden and output layers affects the network's performance.

b. Weight Adaptation

Cumulative weight adjustment [11] refers to the implementation of weight adjustments at the conclusion of a complete learning cycle. During incremental training, the weights continue to be modified as each error is computed. If the network is capable and the learning rate is set correctly, the error will eventually be driven to zero. In batch mode, the weights and biases of the network are updated only after the entire training set has been applied to the network.

c. Learning Constant

Its optimum value depends on the problem to be solved and is normally chosen experimentally (with values ranging from 0.05 to 0.35). Only small learning rate guarantees a true smooth gradient descent. Too large a value leads to fast convergence but poor stability. Too small a value results in slow convergence. An adaptive rate may be more suited for exploratory work.

d. The Hidden Layer

The best number of hidden units depends on the numbers of input and output units; the number of training examples; the complexity of the function or classification to be learned; the architecture; the type of hidden unit activation function; the training algorithm

IV. IMPLEMENTATION RESULTS

a. Training Parameters and Weight Initialization-

Error-epoch graphs are used to evaluate training quality. The graph in Fig. 2 shows consistent and smooth convergence during training. The first set of faces even fails to converge during training. The rest did not converge smoothly. It is seen that, the descent is too fast. The graph was also used to compare the learning quality of the validation set with that of the training set. The weights that produce the smallest training error between the two are saved.

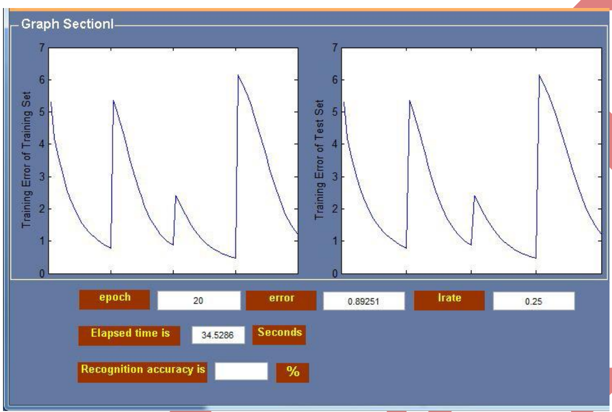

Figure 2: An error-epoch graph showing stability and smooth convergence.

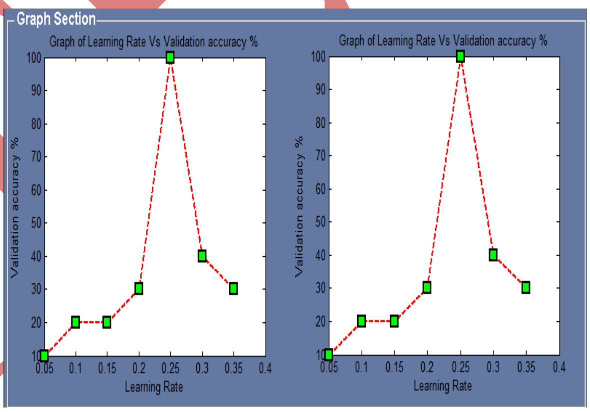

Figure 3: Results for Learning Rate Vs accuracy evaluation.

b. Learning Rate-

The performance of the MSNN is very sensitive to the proper setting of the learning rate. It cannot be set too high; otherwise, the network may oscillate and become unstable. If the learning rate is too small, the algorithm will take a long time to converge. Several trainings should be performed using a variety of learning rates before determining the optimum η. Here experiments are conducted with learning rates ranging from 0.05 to 0.4, each time increasing by 0.05. Recognition accuracy increases with the learning constant until it reaches full recognition at η = 0.25, after which the performance starts to deteriorate. The recognition rate finally drops to 40% at η = 0.35. The observations are plotted out in the graph below.

c. Number of Hidden Neurons-

To determine the number of hidden neurons is always a discussion topic in neural networks. In this experiment, MSNN is trained with different numbers of hidden units (5 to 40) and recorded their recognition rates. For a network with 10 to 20 hidden neurons, the recognition accuracy is 100%. Any number of hidden units beyond 20 will experience a gradual decrease in performance, eventually hitting zero recognition at 40 neurons. As the graph in Fig. 6.24 indicates, efficiency increases as the number of neurons increases from 5 to 10; it remains constant between 10 to 15 neurons, and drops drastically after that. Although it is true that increasing the number of hidden neurons can extract more implicit information, an excessive number may cause over fitting and high

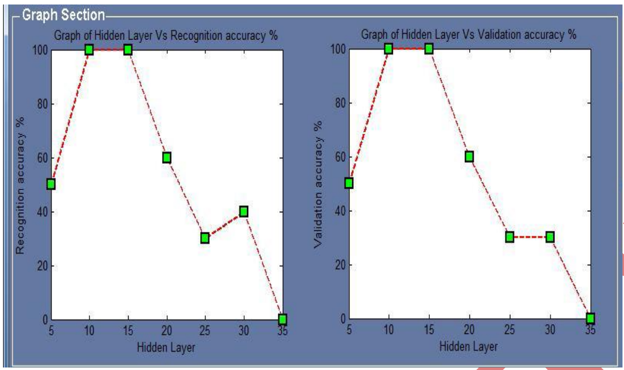

Fig. 6 Graph of recognition Accuracy vs. NO of Hidden Cells

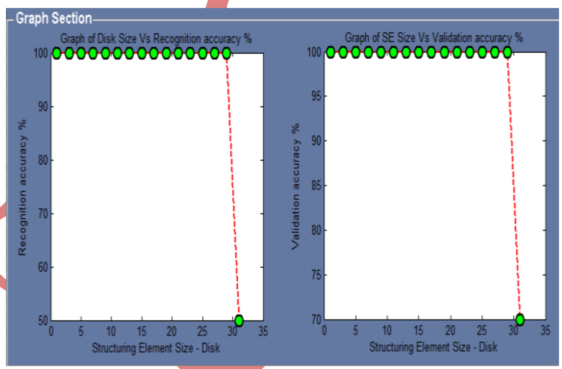

Figure 4: Results for Disk Type Structuring Element size Vs Accuracy Evaluation.

d. Evaluation of Sizes and Shapes of the Structuring Element-

Here the size and shape of the structuring element is varied. The size was increased progressively from 1×1 to 31×31 pixels. Separate tests were conducted for the "disk" and the "diamond" structuring element. Results showed that the MSNN is not very sensitive to structuring element size and shape (Fig. 4, Fig. 5). For the network that uses a "disk" structuring element, recognition accuracy remains constant at 100% until it drops abruptly at the size of 31×31 pixels; the fail size is 29×29 pixels for the network that uses a "diamond" structuring element.

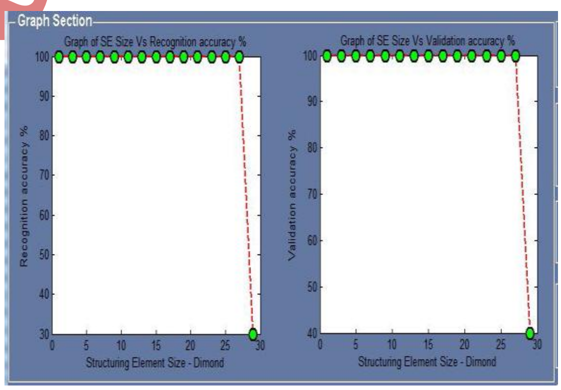

Figure 5: Results for Diamond Type Structuring Element Size Vs Accuracy Evaluation.

V. CONCLUSION

From the evaluation and analysis data it is concluded that with increased size of the structuring element validation and recognition accuracy reduces so use small size of structuring element. From the Training time and Test time analysis data it is concluded that with increase in size of the structuring element processing time for training MSNN increases. Structuring element type is decided by conducting analysis for two types Disk type and Diamond type from the analysis it is concluded that the results for disk type of structuring element gives better validation and recognition accuracy over defined range. From the learning rate analysis result it is concluded that validation and recognition accuracy is reach to 100 % for 0.25. Further analysis for deciding hidden layer for MSNN is also conducted from the analysis data it is concluded that validation and recognition accuracy reaches to 100 % between the ranges of 10 to 15 so finally above parameters are chosen for carrying out recognition test.

REFERENCES

[1] Y. Won et. al., Morphological Shared-Weight Networks with Applications to Automatic Target Recognition, Electronics and Telecommunications Research Institute, Daejon, South Korea, 1995.

[2] Y. Won and P. Gader Morphological Shared-Weight Neural Network for Pattern Classification and Automatic Target Detection, University of Missouri-Columbia, 1995.

[3] D. Haun, K. Hummel, and M. Skubic, Morphological Neural Network Vision Processing for Mobile Robots, University of Missouri-Columbia, 1997.

[4] U. Uludag and A. Jain, Biometrics, International Conference on Pattern Recognition, Department of Computer Science and Engineering, Michigan State University, 1999.

[5] D. Samal, V.V. Starovoitov and D.V. Briliuk ‘’ Image Enhancement for Face Recognition’’International Conference on Iconics, 2003, St.Petersburg, Russia.

[6] G. Scott and M. Skubic, Face Recognition Using Morphological Shared-Weight Neural Networks, University of Missouri-Columbia, 1995.

[7] Mohamed A. Khabou and Laura F. Solari “A Morphological Neural Network-Based System for Face Detection and Recognition” IEEE 2006.

[8] L. Aryananda, Online and Unsupervised Face Recognition for Humanoid Robot: Toward Relationship with People, A. I. Lab, MIT, 2001.

[9] Y. Lee, S. Oh, and M. Kim, The Effect of Initial Weights on Premature Saturation in Backpropagation Learning, International Joint Conference on Neural Nets, Seattle, 1991.

[10] Y. Won, P. Gader, and P. Coffield, “Morphological Shared-Weight Neural Networks with Applications to Automatic Target Recognition,” *IEEE Transactions Neural Networks*,Vol. 8, pp. 1195–1203, 1997.

[11] Y. Lee, S. Oh, and M. Kim, The Effect of Initial Weights on Premature Saturation in Backpropagation Learning, International Joint Conference on Neural Nets, Seattle, 1991.