Proceedings of the 7th WSEAS International Conference on Neural Networks, Cavtat, Croatia, June 12-14, 2006 (pp152-157)

A New Approach for Fast Face Detection

Hazem M. El-Bakry

Faculty of Computer Science & Information Systems,
Mansoura University, EGYPT

Nikos Mastorakis

Department of Computer Science,
Military Institutions of University Education
(MIUE) -Hellenic Naval Academy, Greece

Abstract- In this paper, a new approach to reduce the computation time taken by neural networks for the searching process is introduced. Both fast and cooperative modular neural networks are combined to enhance the performance of the detection process. Such approach is applied to identify human faces automatically in cluttered scenes. In the detection phase, neural networks are used to test whether a window of 20x20 pixels contains a face or not. The major difficulty in the learning process comes from the large database required for face / nonface images. A simple design for cooperative modular neural networks is presented to solve this problem by dividing these data into three groups. Such division results in reduction of computational complexity and thus decreasing the time and memory needed during the test of an image. Simulation results for the proposed algorithm on Bio database show a good performance.

Keywords: Fast Neural Networks, Face Detection, Cross Correlation in the Frequency Domain

1. Introduction

The goal of this paper is to solve the problem of requiring large database to build an automatic system in order to detect the location of faces in scenes. This paper explores the use of modular neural network (MNN) classifiers. Non-modular classifiers tend to introduce high internal interference because of the strong coupling among their hidden layer weights [3,5]. As a result of this, slow learning or over fitting can occur during the learning process. Sometimes, the network could not be learned for complex tasks. Such tasks tend to introduce a wide range of overlap which, in turn, causes a wide range of deviations from efficient learning in the different regions of input space [3,5]. High coupling among hidden nodes will then, result in over and under learning at different regions [8]. Enlarging the network, increasing the number and quality of training samples, and techniques for avoiding local minina, will not stretch the learning capabilities of the NN classifier beyond a certain limit as long as hidden nodes are tightly coupled, and hence cross talking during learning [7]. A MNN classifier attempts to reduce the effect of these problems via a divide and conquer approach. It, generally, decomposes the large size / high complexity task into several sub-tasks; each one is handled by a simple, fast, and efficient module. Then, sub-solutions are integrated via a multi-module decision-making strategy. Hence, MNN classifiers, generally, proved to be more efficient than non-modular alternatives [2,5,6]. In section II, a method for detection of human faces in photo images is presented. Also, an algorithm during the searching procedure is described. A fast searching algorithm for face detection which reduces the computational complexity of neural networks is presented in section III.

II. Human Face Detection Based on Neural Networks

The human face is a complex pattern. Finding human faces automatically in a scene is a difficult yet significant problem. It is the first step in fully automatic human face recognition system. Face detection is the fundamental step before the face recognition or identification procedure. Its reliability and time response have a major influence on the performance and usability of the whole face recognition system. Training a neural network for the face detection task is challenging because of the difficulty in characterizing prototypical "nonface" images [1]. Unlike face recognition, in which the classes to be discriminated are different faces, the two classes to be discriminated in the face detection are "image containing faces" and "image not containing faces". It is easy to get a representative sample of images that contain faces, but much harder to get a representative sample of those which do not. Feature information needs to be stored in the database for the purpose of retrieval. Information retrieval can be done by using a neural network approach which has the potential to embody both numerical and structural face data. However, neural network approaches have been demonstrated only on limited database. The use of huge samples of face/nonface images makes the learning process very difficult for the neural network.

A) A Proposed Algorithm For Face Detection Using MNNs

First, in an attempt to equalize the intensity values of the face image, the image histogram is equalized. This not only reduce the variability of generated by illumination conditions, and enhance the image contrast but also increases the number of correct pixels that can be actually encountered [3]. The next component of our system is a classifier that receives an input of 20x20 pixel region of gray scale image and generates an output region ranging from 1 to -1, signifying the presence or absence of a face, respectively. This classification must have some invariance to position, rotation, and scale. To detect faces anywhere in the input, the classifier is applied at every location in the image. To detect faces larger than the window size, the input image is repeatedly reduced in size. The classifier is applied at every pixel position in the image and scale the image down by a factor of 1.2 for each step as shown in Fig. 1. So, the classifier is invariant to translation and scaling. To have rotation invariant, the neural network is trained for images rotated from 0° to 345° by a step of 15°. In order to train neural networks used in this stage, a large number of face and nonface images are needed. A sample of nonface images, which are collected from the world wide web, is shown in Fig. 2. So, conventional neural networks are not capable of realizing such a searching problem. As a result of this, MNNs are used for detecting the presence or absence of human faces for a given image. Images (face and nonface) in the database are divided into three groups which result in three neural networks. More divisions can occur without any restrictions in case of adding more samples to the database. Each group consists of 600 patterns (300 for face and 300 for nonface). Each group is used to train one neural network. Each network consists of hidden layer containing 30 neurons, and an output layer which contains only one neuron. Here, two models of MNNs are used. The first is the ensemble majority voting which gives a result of 82% detection rate for the Bio-Database. The other is the average voting which gives a better result of 89% detection rate.

B) Enhancement of Recognition Performance

To enhance the detection decision, the detection results of neighboring windows can be used to confirm the decision at a given location. This will reduce false detection as neighboring windows may reveal the nonface characteristics of the data. For each location the number of detections within a specified neighborhood of that location can be counted. If the number is above a threshold, then that location is classified as a face. Among a number of windows, the location with the higher number of detections in range of one pixel is preserved, and locations with fewer detections are eliminated. In our case, a threshold of 4 is chosen. Such strategy improves the detection rate for the Bio-database to 96% (average voting), as a result of reducing the false detections. It is clear that, the use of MNNs and this enhancement has improved the performance over our previous results in [9], where non-modular neural networks are used, in which the best result on the same samples was 61% [9].

III. Fast Neural Networks For Face Detection

In subsection 2.1, modular neural network for object detection is presented using a sliding window to test a given input image. In this section, a fast algorithm for object detection (used with each of the neural networks presented in section 2.1) based on two dimensional cross correlations that take place between the tested image and the sliding window. Such window is represented by the neural network weights situated between the input unit and the hidden layer. The convolution theorem in mathematical analysis says that a convolution of f with h is identical to the result of the following steps: let F and H be the results of the Fourier transformation of f and h in the frequency domain. Multiply F and H in the frequency domain point by point and then transform this product into spatial domain via the inverse Fourier transform. As a result, these cross correlations can be represented by a product in the frequency domain. Thus, by using cross correlation in the frequency domain a speed up in an order of magnitude can be achieved during the detection process [1-3]. In the detection phase, a sub image I of size mxn (sliding window) is extracted from the tested image, which has a size PxT, and fed to the neural network. Let X_i be the vector of weights between the input sub image and the hidden layer. This vector has a size of mxn and can be represented as mxn matrix. The output of hidden neurons h(i) can be calculated as follows:

$$h_i = g\left(\sum_{j=1}^{m}\sum_{k=1}^{n} X_i(j,k)I(j,k) + b_i\right) \tag{1}$$

where g is the activation function and b(i) is the bias of each hidden neuron (i). Eq. 1 represents the output of each hidden neuron for a particular sub-image I. It can be obtained to the whole image Z as follows:

$$h_i(u,v) = g\left(\sum_{j=-m/2}^{m/2}\sum_{k=-n/2}^{n/2} X_i(j,k)\,Z(u+j,v+k)+b_i\right) \quad (2)$$

Eq. 2 represents a cross correlation operation. Given any two functions f and d, their cross correlation can be obtained by [12-14]:

$$f(x,y)\otimes d(x,y)=\left(\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} f(x+m,y+n)d(m,n)\right) \quad (3)$$

Therefore, Eq. 2 may be written as follows [12-14]:

$$h_i = g\left(Z\otimes X_i + b_i\right) \quad (4)$$

where h_i is the output of the hidden neuron (i) and h_i (u,v) is the activity of the hidden unit (i) when the sliding window is located at position (u,v) and (u,v) $\in$ [P-m+1,T-n+1].

Now, the above given cross correlation can be expressed in terms of Fourier Transform:

$$Z\otimes X_i = F^{-1}\left(F(Z)\bullet F^*\left(X_i\right)\right) \quad (5)$$

Hence, by evaluating this cross correlation, a speed up ratio can be obtained compared to conventional neural networks. Also, the final output of the neural network can be evaluated as follows:

$$O(u,v) = g\left(\sum_{i=1}^{q} w_o(i)\,h_i(u,v) + b_o\right) \quad (6)$$

O(u,v) is the output of the neural network when the sliding window located at the position (u,v) in the input image Z.

For a tested image of NxN pixels, the 2D-FFT requires $(5N^2\log_2N^2)$ real computation steps [11]. The same number of computation steps is required for the weight matrix at each neuron in the hidden layer. The inverse 2D-FFT of the resulted dot product must be computed at each neuron in the hidden layer. As a result, q backward and (q+1) forward transforms have to be computed. Therefore, for a tested image, the total number of the 2D-FFT to compute is $((2q+1)(5N^2\log_2N^2))$. Moreover, the input image and the weights should be multiplied in the frequency domain. Therefore, real computation steps of $(6qN^2)$ should be added. Finally, a total of $O((q+1)(5N^2\log_2N^2)+6qN^2)$ computation steps must be evaluated for fast the neural algorithm. , for the weight matrix to have the same size as the input image, a number of zeros = (N^2-n^2) must be added to the weight matrix. This requires a total real number of real computation steps = $q(N^2-n^2)$ for all neurons. Moreover, after computing the 2D-FFT for the weight matrix, the conjugate of this matrix must be obtained. So, a real number of computation steps $=qN^2$ should be added in order to obtain the conjugate of the weight matrix for all neurons. Also, a number of real computation steps equal to N is required to create butterflies complex numbers $(e^{-jk(2\Pi n/N)})$, where 0<K<L. These (N/2) complex numbers are multiplied by the elements of the input image or by previous complex numbers during the computation of 2D-FFT. To create a complex number requires two real floating point operations. Thus, the total number of computation steps required for fast neural networks becomes [11]:

$$\sigma=((2q+1)(5N^2\log_2N^2) + q(8N^2-n^2) + N) \quad (7)$$

Using sliding window of size nxn, for the same image of NxN pixels, $(q(2n^2-1)(N-n+1)^2)$ computation steps are required when using traditional neural networks for the face detection process. The theoretical speed up factor η can be evaluated as follows [11-14]:

$$\eta = \frac{q(2n^2-1)(N-n+1)^2}{(2q+1)(5N^2\log_2N^2)+q(8N^2-n^2)+N} \quad (8)$$

The speed up factor introduced in [10] for object detection which is given by:

$$K = \frac{qn^2}{(q+1)\log^2 N} \quad (9)$$

is not correct for the reasons given in [11]. The relation between the image size and speed up ratio is shown in Fig. 3. Practical speed up ratio is shown in Fig.4 using MATLAB ver 5.3 and 700 MHz processor. A comparison between the classic and fast neural networks for different window size is illustrated in Fig. 5.

IV. Conclusion

A fast modular neural network approach has been introduced to identify frontal views of human faces. Such approach can manipulate gray scale images of resolution 20x20 up to 500x500 pixels. The technical problem associated with large database (face/nonface) required for training neural networks has been solved using MNNs. A simple algorithm for fast face detection based on neural network and FFT is presented in order to speed up the execution time. Simulation results on Bio-database have shown that our algorithm is an efficient method for finding locations of faces when the size of the face is unknown as well as mirrored, noised, and occluded faces are detected correctly.

References

[1] H. A. Rowley, S. Baluja, and T. Kanade, "Neural Network - Based Face Detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, pp. 23-38, 1998.

[2] R. Jacobs, M. Jordan, and A. Barto, "Task Decomposition Through Competition in a Modular Connectionist Architecture: The what and where vision tasks," Neural Computation 3, pp. 79-87, 1991.

[3] Hazem M. El-Bakry, "Automatic Human Face Recognition Using Modular Neural Networks," Machine Graphics & Vision Journal (MG&V), vol. 10, no. 1, 2001, pp. 47-73.

[4] R. Klette, and Zamperon, "Handbook of image processing operators," John Wiley & Sonsltd, 1996.

[5] Hazem M. El-Bakry, "Human Iris Detection Using Fast Cooperative Modular Neural Networks and Image Decomposition," Machine Graphics & Vision Journal (MG&V), vol. 11, no. 4, 2002, pp. 498-512.

[6] E. Alpaydin, " Multiple Networks for Function Learning, " Int. Conf. on Neural Networks, Vol.1 CA, USA, pp. 9-14, 1993.

[7] A. Waibel, "Modular Construction of Time Delay Neural Networks for Speech Recognition," Neural Computing 1, pp. 39-46, 1989.

[8] K. Joe, Y. Mori, and S. Miyake, "Construction of a large scale neural network: Simulation of handwritten Japanese Character Recognition," on NCUBE Concurrency 2 (2), pp. 79-107, 1990.

[9] Hazem M. El-bakry, and M. A. Aboelsoud, "Automatic Personal Identification Using Neural Networks," The 24th international Conf. on Statistics computer Science, and its applications, pp. 405-416, cairo, Egypt, 1999.

[10] S. Ben-Yacoub, "Fast Object Detection using MLP and FFT," IDIAP-RR 11, IDIAP, 1997.

[11] H. M. El-Bakry, "Comments on Using MLP and FFT for Fast Object/Face Detection," Proc. of IEEE IJCNN'03, Portland, Oregon, pp. 1284-1288, July, 20-24, 2003.

[12] Hazem M. El-Bakry, "Fast Pattern Detection Using Parallel Neural Processors and Image Decomposition," Proc. of the 19th European Conference on Modeling and Simulation (ESM2005), Simulation of Intelligent Systems, 1-4 June 2005, Riga, Latvia.

[13] Hazem M. El-Bakry, "A Fast Neural Algorithm for Patten Detection Using Cross Correlation in the Frequency Domain," Proc. of the 19th European Conference on Modeling and Simulation (ESM2005), The 2005 High Performance Computing & Simulation (HPC&S) Conference, 1-4 June 2005, Riga, Latvia.

[14] Hazem M. El-Bakry, "Fast Sub-Image Segmentation Using Parallel Neural Processors and Image Decomposition," Proc. of the SPIE Vol. 5960, the International Symposium on Visual Communication and Image Processing (VCIP), 12-15 July, 2005, Beijing, China, pp. 1712-1722.

Fig. 1. Image resizing by a factor of 1.2 during face detection.

Fig. 2. Examples of nonface images.

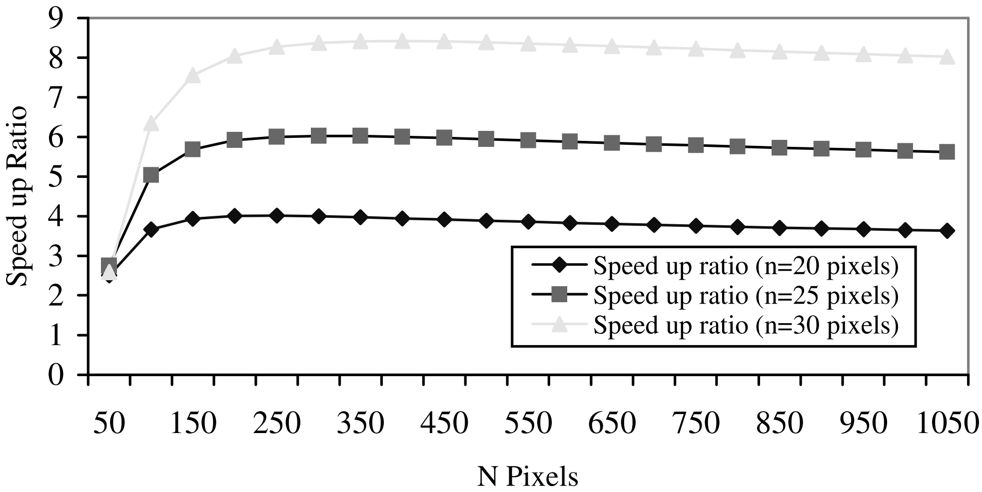

Fig. 3. The relation between the size of the image under test and the speed up ratio.

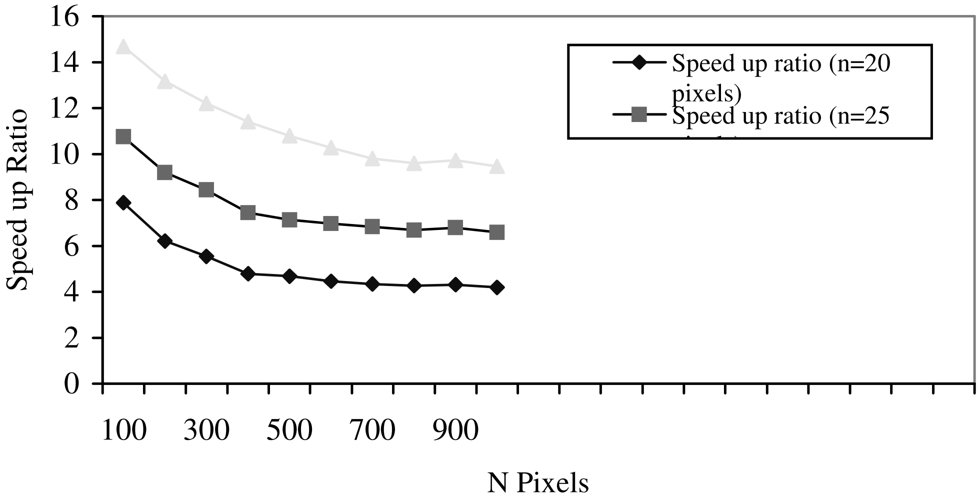

Fig. 4. Practical speed up ratio for images with different size.

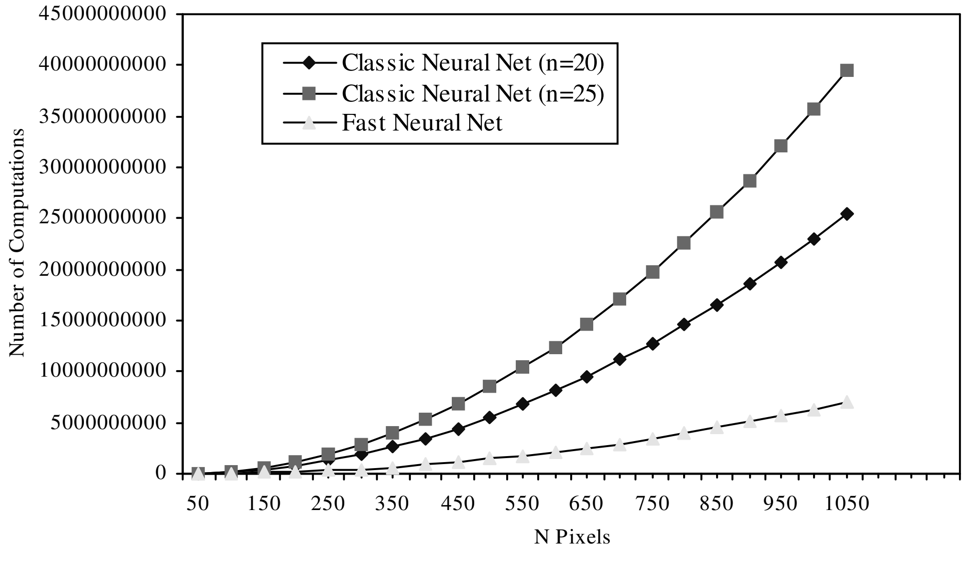

Fig. 5. A comparison between the number of computations taken by classic and fast neural networks for face detection.