AN AUTOMATIC HUMAN FACE DETECTION ALGORITHM

Irfan MUHAMMAD[1] *Ziya TELATAR*[2]

[1,2] Ankara University, Department of Electronics Engineering, Beşevler, 06100, Ankara, Turkey.

muhammad@science.anakara.edu.tr, telatar@science.ankara.edu.tr

Abstract

The paper presents an algorithm for detecting face location(s) in two-dimensional static images having complex background. The algorithm uses back-propagation neural network for detecting face patterns in the image. The network was trained by using an average face and non-face image sets, thus simplifying the training of the network. The image was sub sequentially resized to smaller dimension for detecting faces of different size in the image.

Key words: Face detection, Neural network

Face recognition

Introduction

The detection of a face is an easy task for humans, since humans have been doing this task for a very long time and start doing it right from their birth. The same job is not easy when it comes to computers. The task becomes more difficult when faces of different size have to be detected in same photograph and with complex background. Not only size has to be taken into account but also rotated and partial faces are to be detected.

In a face recognition system the prerequisite is the accurate location of the face ie detecting a face in an image, so that the latter system can use these detected face for recognition. As the means of communication are growing the information interchange is increasing at logarithmic rate, with the advent of multimedia application on internet the demand for searching of pictures or identification of a person over the internet requires an algorithm for recognition of faces and the prerequisite for this is the correct location of the face. As e-commerce becomes more and more common, the necessity for secure transaction will increase, and best way to identify anyone is his face.

Most research work to date can be regarded as only a starting point. Many aspects of human face remain to be solved. Reported works have used prior knowledge of the location of face, one face in the image or have simple background due to which the task at hand becomes easy. Many researchers have developed different algorithms for face detection. Computational model [1] uses micro templates to detect faces, hierarchical knowledge based systems [2,3] uses mosaics of the face for detection, principle component analysis method [4] divides the image into its components and then tries to detect a face, skin or lip color has also been used to detect a face, neural network [2,5,6] have also been used to detect faces in images.

The algorithm presented here uses a new approach in training the neural network and on the usage of this network, enabling the algorithm to detect face of different size with complex background. Following sections explain the theoretical background of neural network and system details of the algorithm.

Theoretical Background

In our work a modified form of backpropagation neural network algorithm developed by Osama etl. [9] was used for neural network. The algorithm is much faster and provides much better numerical stability than normal algorithm for training feed-forward networks.

The algorithm is based on backpropagation neural network. The output of the neural network is given by

$$\boldsymbol{y} = \Sigma(\boldsymbol{x} \,.\, \boldsymbol{w}^{i}) \qquad (1)$$

here, $\boldsymbol{y}$ is the output matrix, $\boldsymbol{x}$ is the input matrix presented to neural network, $\boldsymbol{w}$ is the weight matrix and $\boldsymbol{i}$ is the layer number. This output is then passed through a non-linear function given below

$$f(y) = \frac{[1 - \exp(-a \cdot y)]}{[1 + \exp(-a \cdot y)]} \qquad (2)$$

This algorithm improves the training by controlling 'k' the gain of a Kalman filter, which is estimated by

$$\boldsymbol{k}_j = \boldsymbol{S}_j(\boldsymbol{x}_j\boldsymbol{S}_j) \qquad (3)$$

$\boldsymbol{S}$ is a random matrix, updated at every iteration as follows

$$S_1 = \frac{S - (\gamma k)(xS)^T}{b} \qquad (4)$$

where γ is calculated as follows

$$\gamma = \frac{\sigma}{1 + \sqrt{\sigma}} \qquad (5)$$

where σ is calculated as follows

$$\sigma = \frac{1}{[(xS)^T (xS) + b]} \qquad (6)$$

The error $\boldsymbol{e}$ for the hidden layer and output layer is calculated as follows

$$\boldsymbol{e} = \boldsymbol{f'}\, \Sigma \boldsymbol{y}(\boldsymbol{e}^{i-1} .\, \boldsymbol{w}^{i}) \qquad (7)$$

using this error output weight matrix is updated as follows

$$w = w + k(d - y) \quad (8)$$

Here d is the desired output matrix. The hidden layer weight matrices are updated as follows

$$w = w + k \,.\, e \,.\, \mu \quad (9)$$

The algorithm uses the above relations to implement the neural network.

System details

The work presented here assumes that the faces present in the image are all upright and not rotated on any axis. All the images[1] used are 256 gray level images, any other image if presented, is first transformed into 256 gray level image and then further processing is done.

A training data set for training the neural network, which is being used for detecting face patterns, was prepared. This data set includes different face patterns from different images. Every face pattern from different images was pointed by hand and then stored. For each left and right eye their centers were pointed and then the mid point of the upper lip was pointed out manually and this information was used to extract the face pattern, these were then resize according to the pixel distances shown in figure 2. Each of these faces was then stored and resize to a new dimension of 20x20 pixel. This size puts a limit on the smallest face that can be detected by the network. Similarly a data set of non-face patterns was prepared for testing the neural network.

The face patterns thus obtained were then averaged and form that an average face (figure 3) was obtained. This face has the properties from all faces. Thus this average face represents nearly all the faces present in the data set. Another data set, which contains non-face patterns was also prepared. This data set was used to train the network for distinguishing non-face patterns from face patterns. Also contained in this data set are matrices with random values, of 20x20 size.

The neural network has one hidden layer, the input layer has 400 inputs and output layer has eight outputs. The neural network was trained to give an output of +1 for face patterns and an output of 0 for non-face patterns. Each image was scanned by a window size of 20x20, the image from the window was first histogram normalized, to carter for different light conditions, then divided into four rows and four columns, neural network was applied on each sub-matrix, output from these was summed and used as an indication of the presences or absence of the face pattern at that location. This is shown in figure 3.

Each image was scanned by a window of 20 x 20 pixel at different dimension, each dimension was resized by 10% of the original size at each new dimension. This process was repeated till the smallest dimension for scanning was reached.

[1] Most of the images for training were downloaded from http://www.cam-orl.co.uk/facedatabase.html

Probable face candidates at each dimension were then compared with each other at different location and by the process of elimination at each stage weak candidates were rejected. If a face appears at same location in more than two consecutive dimension then it was said to be a face in the image. In the end, face size and its location were computed from the data at different dimensions and a square was plotted around the face in the original image.

Experimental Results

The neural network was tested using face and non-face patterns from the testing data set. The results are shown in table 1. The network was able to detect all of the 295 faces, which were in the training set but detected 22 non-faces as faces from the non-face testing data set.

The algorithm was applied to different photographs with faces at different locations, these results are shown in figures 4, 5, 6 and 7. Figure 4 contains 2 faces and both of them have been identified without any false detection, in figure 5, out of 8 faces 5 have been detected with 7 non-faces, in figure 6 & 7 all of the faces have been detected with 2 non-faces in each of the image. When applied to photographs the overall detection rate was nearly 90%, the algorithm also pointed some non-face areas as faces, these non-faces represents the error in the algorithm.

In figure 8 there are 3 faces and out of 3 one has been detected and no false detection. The two other faced not detected are due to the fact that the image has lot of noise and is blurred. This image was used to test the response of algorithm against noise and blurred images.

Other similar works have reported detection results of 78 to 92 % [4], 75 to 93% [10] and error rate of 7.4 to 9.3 % [11].

Conclusions and future research

The network works well with upright and non-rotated faces, but with rotated faces the algorithm is not able to detect faces. In figure 5 three faces has not be detected by the algorithm, this is because the network was not trained to recognize dark tone skin colors. The non-face-data set, will be updated by including the non-face patterns pointed out as face patterns and retrain the network.

The research to follow will include the detection of faces rotated about different axis, having different skin color. Also each face may not be exactly square in nature, some faces are long and thin and others are wide and short. The algorithm will also cater for these differences in the face patterns and try to incorporate these differences.

References

1. Govindaraju V., Srihari S.N. and Sher D.B., (1990), 'A computational model for face location', in Proc. 3rd Int. Conf. on Computer Vision, p. 718-721.
2. Midorikawa, (1988), 'Face pattern identification by backpropagation learning procedure', Neural Networks, vol. 1, p. 515.
3. Yang G. and Huang T.S., (1994), 'Human face detection in a complex background', Patt. Recog. vol. 27, p. 53-63.
4. Baldi P. and Hornik K., (1989), 'Neural networks and principal components analysis: Learning from examples without local minima', Neural Networks, vol 2, p. 53-88.
5. Rowley H.A., Baluja S. and Kanade T., (1996), 'Neural Network based face detection', Proc. Computer Vision and Patt. Recog. San Francisco, California, p 203-208.
6. Sung K. K. & Poggio T., (1995), 'Learning human face detection in cluttered scenes', in Computer Analysis of Image & Patterns, p 432-439.
7. Lin & Hung Shang, (1997), 'Face Recognition/ Detection by probabilistic decision-based neural networks', IEEE Transactions on Neural Networks: v8 n1, Jan , p144-132.
8. Sanger Demas, (1997), 'Algorithm for Face Extraction based on lip detection', Journal of Imaging Sciences and Technology, v 41 n1 Jan/Feb.
9. Abdel-Wahhab Osama, Sid-Ahmed M.A., (1997), 'A new scheme ofr taining feed-forwad neural networks', Pattern Rcognition, v 30 no. 3, p 519-524.
10. Miao Jun, Yin Baocai, etl, (1999), 'A hierarchical multiscale and multiangle system for human face detection in a complex background using gravity-center template', Pattern Recgnition, v 32, p 1237-1248.
11. Matas J., Jonsson K., Kittler J., (1999), 'Fast face localisation and verification', Image and Vision Computing, v 17, p 575-581.

	Face patterns	Non-face patterns
Total number of faces	100	100
detected as faces	100	22
detected as non-faces	0	78

Table 1

No of faces in the photograph	**No of faces detected (%)**	**Non-face patterns detected as face**
2	2 (100%)	0
2	2 (100%)	2
8	5(63%)	7
3	3(100%)	2

Table 2

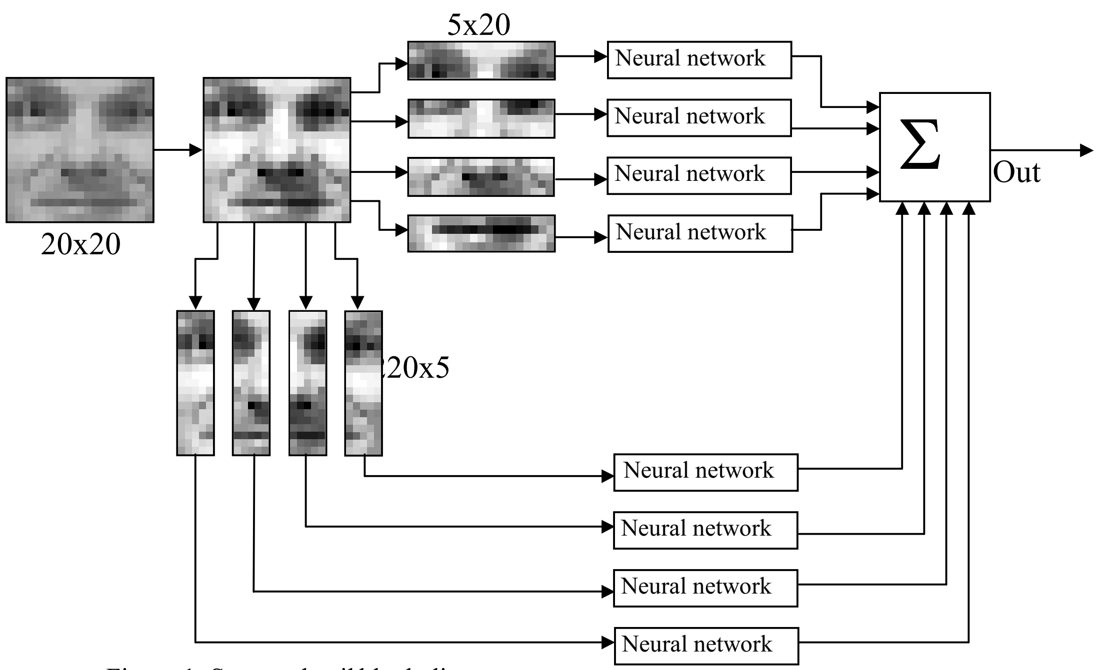

Figure 1: System detail block diagram

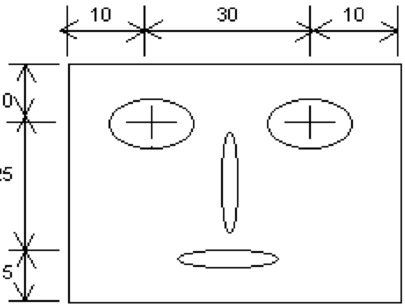

Figure 2 Pixel distance for face

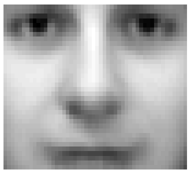
Figure 3 : Average face

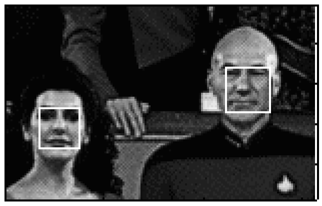
Figure 4: Faces located by algorithm

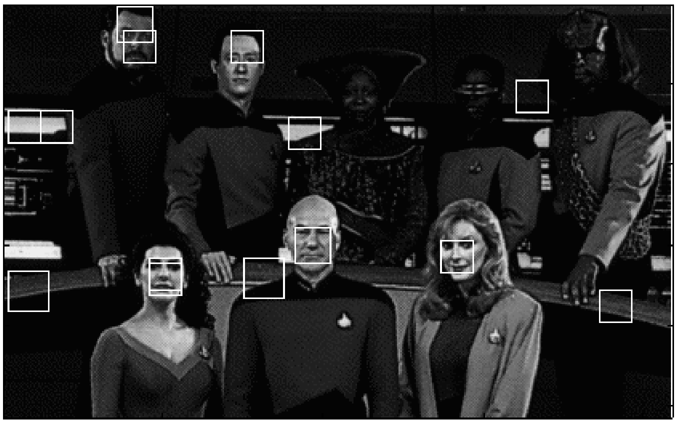
Figure 5: Faces located by algorithm

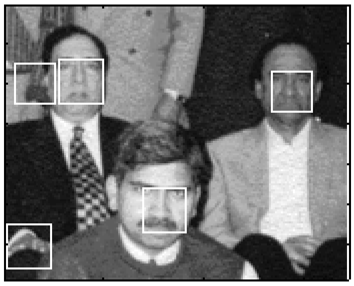
Figure 6: Faces located by algorithm

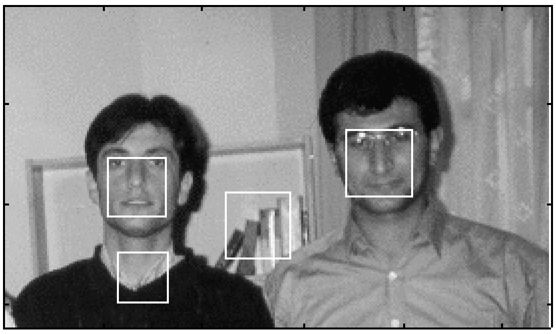
Figure 7: Faces located by algorithm

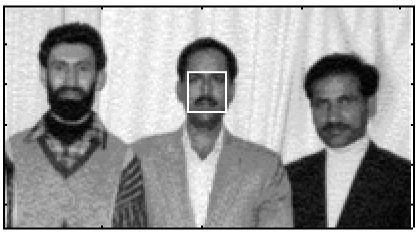
Figure 8: Faces located by algorithm in noisy and blurred image