A BIOMETRIC SYSTEM FOR PERSONAL IDENTIFICATION USING MODULAR NEURAL NETS

نظام بايومترى للتعرف على الأشخاص بإستخدام الشبكات العصبية المعدلة

Hazem M. El-Bakry
Faculty of Computer Science &Information System
Mansoura University - Egypt
helbakry1@hotmail.com

Mohy A. Abo-Elsoud
Electronics & Elec. Comm. Dept
Faculty of Engineering
Mansoura University – Egypt
mohyldin@mum.mans.eun.eg

Mohamed S. Kamel
Systems Design Eng. Dept
University of Waterloo
Canada
mkamel@watfast.uwaterloo ca

ملخص البحث: يهدف هذا البحث إلى دراسة إمكانية إستخدام وجوه الأشخاص كوسيلة بديلة لحماية المعلومات بدلاً من أساليب الحماية التقليدية والتى من الممكن أن تتلف أو 'تسرق أو 'تفقد مثل الكروت أو 'تنسى أو يحدث لها تخمين مثل كلمات السر. وتعتبر وجوه الأشخاص هى الطريقة الوحيدة التى يمكن من خلالها إجراء عملية التعرف بطريقة خفية دون شعور الشخص المراقب. وتتم عملية التحقق من الشخصية على مرحلتين. أولاً: من الممكن أن تحتوى الصورة المدخلة على وجه لشخص أو لا وبالتالى يتم أولاً إختبار ما إذا كانت الصورة تحتوى على وجه من عدمه. وحيث أنه من الممكن أن تكون الصورة المدخلة كلها صورة لوجه أو جزء من الصورة فقط هو الذى يحتوى على الوجه لذلك فى حالة إحتواء الصورة على الوجه يتم إستخلاص هذا الجزء من الصورة ككل ويعتبر هذا الجزء هو الدخل للمرحلة الثانية. وقد تم تطوير أسلوب البحث بإستخدام الشبكات العصبية حتى يمكن الكشف عن وجود وجه من عدمه فى زمن أقل. وقد لوحظ من نتائج الأبحاث السابقة أن المشكله الأساسية هى أن عدد الصور المطلوبة فى أثناء عملية التعليم خاصة بالنسبة للصور التى لا تحتوى على الوجوه ينبغى أن تكون كبيرة. وفى هذا البحث تم حل هذه المشكلة عن طريق إستخدام الشبكات العصبية المعدلة. ثانياً: يتم التمييز بين الأشخاص وبعضهم البعض عن طريق مقارنة الوجه المكتشف فى المرحلة السابقة بمجموعة الوجوه المختزنة فى قاعدة البيانات. لذلك تم دمج معاملات فوريير لصورة الوجه مع معاملات الموجات للإستفادة من مميزاتهما معاً. وقد نتج عن ذلك تحسين كفاءة عملية التعرف مع تقليل الزمن اللازم لإتمام هذه العملية.

ABSTRACT

In this paper, a fast biometric system for face recognition is introduced. We combine both fast and cooperative modular neural nets (MNNs) to enhance the performance of the detection process Such approach is applied to identify frontal views of human faces automatically in cluttered scenes. In the detection phase, neural nets are used to test whether a window of 20x20 pixels contains a face or not The large number of examples required for face and nonface images makes the convergence process very difficult during the learning process. A simple design for cooperative modular neural nets is presented to solve this problem by dividing these data into three groups Such division

Accepted March 28, 2000

results in reduction of computational complexity and thus decreasing the time and memory needed during the test of an image. For the recognition phase, feature measurements are made through Fourier descriptors which are insensitive to rotation, translation and scaling Such feature is modified to reduce the number of neurons in the hidden layer. From these features, wavelet coefficients are extracted which have been shown to provide advantages in terms of better representation for a given data to be compressed Finally, the resulted vector is fed to a neural net for face classification. Simulation results for the proposed algorithm show a good performance.

1. INTRODUCTION

Face recognition refers to the automatic identification of an individual based on the facial information contained in a digital gray scale image On the other hand, there are many biometric techniques which are automated methods for recognizing a person based on his/her physiological or behavioral characteristics. Various types of biometric algorithms are being used for real-time identification, the most popular are based on face recognition and fingerprint matching. However, there are other biometric techniques that are widely used or under investigation such as iris, retinal scan, speech, handwritten signature, hand vein, facial thermogram, and hand geometry. Among all biometric identification methods, face recognition has the potential to be most user friendly and has, therefore, attracted much attention in recent years [13] Furthermore, while every other biometric requires some voluntary action, face recognition can be used passively. This has advantages both for ease of use and for covert use such as police surveillance. Moreover, it is more accurate than fingerprint recognition [14].

Face identification is one of the most challenging tasks for machine recognition. Over the last couple of years, face recognition has become an active area of research in computer vision It is of interest because of its possible applications in areas such as security and personal checking systems. Although humans seem to recognize faces in cluttered scenes with relative ease, machine recognition is a much more daunting task. Psychological studies of human face recognition suggest that virtually every type of available information is used [21] Progress has advanced to the point that face-recognition systems will soon be demonstrated in real world settings [20]. The rapid development of face recognition is due to a combination of factors: active development of algorithms, availability of a large database of facial images, and a method for evaluating the performance of face recognition algorithms [19]. A general statement of the problem can be formulated as follows: given images of a scene, identify one or more persons in the scene using a stored database of faces. The solution of the problem involving segmentation of faces from cluttered scenes, extraction of features from the face region, identification, and matching [22].

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. In section 2, we present a method for detecting frontal views of human faces in photo images. Also, an algorithm for the searching procedure is described. A fast searching algorithm for face detection, which reduces the computational complexity of neural nets, is presented in section 3. Our approach to the face recognition problem is to combine both Fourier and wavelet transforms with neural nets in order to enhance the recognition performance and reduce the response time during the test phase. In section 4, invariant Fourier descriptors are discussed and a modified algorithm is presented. This modification allows to reduce the total number of neurons required to classify the feature vectors and the time required during the learning / testing phase In order to enhance the recognition performance as

well as reduce the computation time, a combination between the FFT and DWT features is discussed in section 5

2. HUMAN FACE DETECTION BASED ON MODULAR 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.This paper explores the use of MNN classifiers Non-modular classifiers tend to introduce high internal interference because of the strong coupling among their hidden layer weights [2] As a result of this interference, 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]. The rate of convergence of neural network output is very low when training feedforward neural networks for multiclass problems using the backpropagation algorithm While backpropagation will reduce the Euclidean distance between the actual and desired output vectors, the differences between some of components of these vectors increase in the first iteration Furthermore, the magnitudes of subsequent weight changes in each iteration are very small, so that many iterations are required to compensate for the increased error in some components in the initial iterations 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. Experimentally, speeds up of magnitudes can be obtained and in some cases convergence was impossible using the modular approach but not using a non-modular network [10]· Using modular technique, the training time can be reduced in several ways

1 The total number of iterations needed to train the individual nets in the modular approaches is less than the number of iterations required to train a non-modular network for the same task

2. The modules in a modular network are smaller and simple in architecture than the comparable non-modular network So, one training iteration of a modular network requires less time than one iteration of the comparable non-modular network
3. If the total number of patterns in all of the modules exceeds the number of patterns in the non-modular network, the modular approach yields faster convergence, and generalizes well.
4. The modules can be trained independently and in parallel on a cluster of high speed workstations.

Using MNNs, the modules are trained independently in parallel. Similarly, the modules operate in parallel when classifying Arbitration among these networks can be done by majority voting, average voting, anding or oring the outputs of different modules. In [10], each module computes one of the output vector. Hence, MNN classifiers, generally, proved to be more efficient than non-modular alternatives [5,6]

2.1 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 [1]. The second 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. 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. We apply the classifier at every pixel position in the image and scale the image down by a factor of 1.2 for each step. Therefore, this classifier is invariant to changes in scale and position. To train neural networks used in this stage, a large number of face and nonface images are needed. So, conventional neural nets are not capable of realizing such a searching problem. As a result of this, we use MNNs 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 use of three neural networks for training. More divisions can occur without any restrictions in case of adding more samples to the database. Each group consists of 600 patterns (300 for faces and 300 for nonfaces). A sample of nonface images, which are collected from the world wide web. 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, we use two models of MNNs. The first is the ensemble majority voting which gives a result of 77% detection rate The other is the average voting which gives a better result of 82% detection rate. In order to have better results, we try to direct the input pattern to one of the three networks which has been trained for the nearest pattern to it. This requires another network to direct the input data to one of the three networks according to some features taken from such data This network consists of two layers, the hidden layer contains 50 neurons, while the output one contains only one neuron From the Fourier descriptors of the input data inside each window, we may obtain four features. These features are the mean, standard deviation. absolute summation and the range between maximum and minimum value for the Fourier vector This makes the overall detection rate 87%

2.2 Enhancement of Recognition Performance

To enhance the detection decision, we can use the detection results of neighboring windows 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, we preserve the location with the higher number of detections in range of one pixel, and eliminate locations with fewer detections. In our case, we choose a threshold of 4 Such strategy improves the detection rate to 94%, as a result of reducing the false detections It is clear that, the use of MNNs and this enhancement have improved the performance over our previous results in [9], where we used non-modular neural networks, in which the best result on the same samples was 71% .

3. FAST NEURAL NETS FOR FACE DETECTION

In section 2, we presented the neural network for face detection using a sliding window to parse a given input image. In this section, a fast algorithm for face detection (used with each of the neural nets described 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 net weights situated between the input unit and the hidden layer. Using the convolution theorem, these cross correlations between input image and weights can be represented by a product in frequency domain [4]. As a result of this, speed up in order of magnitude can be gained during the test phase

During the detection step, a sub image I of size mxn (sliding window) is extracted from the tested image of size SxT and fed to a neural network. Let w_i be the vector of weights needed to compute the activity of hidden neuron This vector of size mn can be represented as mxn matrix ϕ_i. The output of hidden neurons h(i) can be given by

$$h(i) = g\left(\sum_{j=1}^{m} \sum_{k=1}^{n} \varphi_i(j,k) I(j,k) + b(i) \right) \quad (1)$$

where g is the activation function and b(i) is the bias of each hidden neuron (i). The expression is obtained for a particular sub-image I. Equ. (1) can be extended to the global image ψ as follows.

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

This operation denotes a cross correlation operation For any two functions f and d, their cross correlation is given by:

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

We may reformulate equ (2) as

$$h_i = g\left(\varphi_i \otimes \psi + b_i \right) \quad (4)$$

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

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

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

Evaluating this cross correlation using FFT, an important speed up can be gained in comparison to a classic neural network based on sliding windows Similarly the final

output of the neural network can be expressed by linear combination of the hidden unit activity:

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

where, O(u,v) is the output of the observed window located at the position (u,v) in the input image ψ.

The 2D FFT of a NxN test image requires $O(N^2(\log_2 N)^2)$ computation steps. The 2D FFT of the weight matrix ϕ_i can be computed off line since these are constant parameters of the network independent of the test image A 2D FFT of the test image has to be computed, therefore the total number of FFT to compute is q+1 (one forward and q backward transforms). In addition, we have to multiply the transforms of the weights and the input image in the frequency domain adding a further (qN^2) computation steps. This yields a total of $O((q+1)N^2(\log_2 N)^2+qN^2)$ computation steps for the fast neural networks.

For a classic neural network, $O((N-n+1)^2 n^2 q)$ computation steps are required, when one considers the activity of q neurons in the hidden layer and a square test image of size NxN and a sliding window of size nxn. Therefore, the theoretical speed up factor η is:

$$\eta = O\left(\frac{q(N-n+1)^2 n^2}{(q+1)N^2 \log_2^2 N + qN^2}\right) \quad (7)$$

For N=500, n=20, q=30, we may achieve a speed up factor of 4.46. The relation between the image size and speed up ratio is shown in Fig. 1. A comparison between the classic and fast neural nets for different window size is illustrated in Fig 2

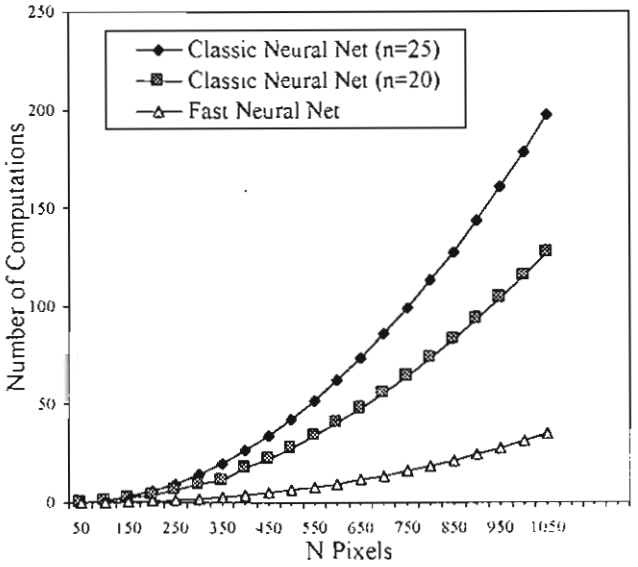

Figure 1. The relation between the length of image under test and the speed up ratio

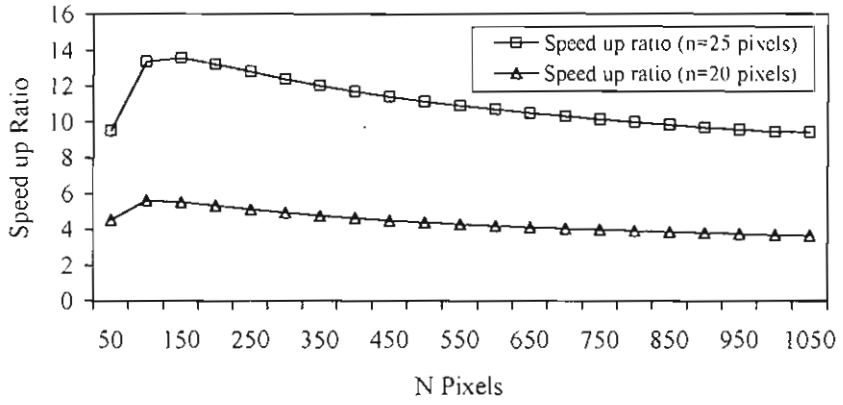

Figure 2 A comparison between the number of computations taken by classic and fast neural nets for face detection

4. FACE RECOGNITION USING INVARIANT FEATURES

Image – invariance schemes assume that there are certain spatial image relationships common and possibly unique to all face patterns, even under different imaging conditions [11]. Fourier descriptors are well suited for recognizing objects using the characteristics of the global shape. By using the Fourier coefficients for an object, we obtain a set of Fourier descriptors that have the invariant property with respect to rotation, translation, and size [15-18] By computing the Fast Fourier Transform of the data, x[m], $1 \le m \le L$, we obtain the Fourier coefficients for $0 \le m \le L-1$ as follows.

$$a[k] = \sum_{m=1}^{L} x[m]\, e^{-jk(2\pi)(L)/m} \qquad (8)$$

For face recognition, we need a certain value, namely, a feature vector which represents the face How to make a feature vector from a given graphic image data is an important issue in pattern recognition. As a feature, we utilize the Fourier descriptor because it gives a unique value irrespective of image rotation, translation, and size scaling Another advantage is that we can compress the data size significantly by eliminating one half of the Fourier components The Fourier descriptors of the input face (20x20 pixels) are used to make the face image invariant to rotation or translation as follows:

1-Discard one half of the data and the dc component a[0] since it depends only on the position of the image center

2-Obtain the absolute value of the complex number a[k] which will make it invariant under rotation or translation as follows

$$r[k] = \sqrt{Re(a[k])^2 + Im(a[k])^2} \qquad (9)$$

3-Obtain the maximum component of the absolute values

4-Dividing r[k] by the maximum absolute value of the other components allowing r[k] to be invariant under scaling

Here, a backpropagation neural network is chosen for classification of feature vectors since it is known to have inherent robustness in classification and flexibility in its application We construct a neural net for solving this classification problem A network

of three layers is learned to classify 120 patterns corresponding to ten human faces. A total number of 12 patterns are taken for each face through rotations from 0° to 165° by a step of 15°. The feature vector consists of 200 elements. Hence, it is common to select the input layer to have 200 nodes. The input nodes do not have any processing element (neuron) The second layer contains 30 neurons while the output layer has 10 neurons to distinguish between 10 human faces. The network is tested for 200 (20 for each face) images containing faces with different scales and orientations. Results show that the network can recognize 98% from these patterns correctly.

In order to reduce the number of neurons in the hidden layer, we make a modification during the learning process. Instead of training the neural net from 0° to 165°, we train it on images rotated from 0° to 75°. This is done as follows:

1. First, the input (face) image is resized 20x20.
2. Second, rotate the input (face) image by an angle of 90°.
3. Obtain FFT for the input (face) image and its rotated version.
4 Calculate the difference between the two vectors computed in the previous step
5 Repeat the previous steps for the input image but with angles up to 75° by a step of 15°.
6. Do the above steps for all the faces to be stored in the database.
7. Train the neural networks with all of these patterns.

Using this procedure the neural network could be learned using only 14 neurons in the hidden layer instead of 30. For some examples of mirrored, noised, and occluded faces, experimental results have shown that the modified algorithm could recognize them correctly

5. COMBINING FOURIER AND WAVELET TRANSFORMS WITH NEURAL NETS FOR FACE RECOGNITION

Recently, wavelet transforms have been shown to provide certain advantages in terms of better data compaction for a given signal. But they are not invariant to rotation, translation or scaling [12]. Using wavelet transform, we can analyze our signal in time for its frequency content. Unlike Fourier analysis, in which we analyze signals using sins and cosines, now we use wavelet function. Dilations and translation of the "mother function" or analyzing wavelet $\phi(x)$, defines an orthogonal basis, our wavelet basis:

$$\phi(\lambda) = 2^{\frac{s}{2}} \phi(2^{-s} x - z) \qquad (10)$$

The variables s and z are integers that scale and dilate the mother function ϕ to generate wavelets, such as a Daubechies wavelet family. The scale index s indicates the wavelet's width, and the location z gives its position. The mother functions are rescaled, or dilated by powers of two, and translated by integers. What makes wavelet bases especially interesting is the self-similarity caused by the scales and dilations Once we know about the mother functions, we know everything about the basis. Compared with Fourier descriptor that uses global sinusoids as the basis functions, the wavelet descriptors is more efficient in representing and detecting local features due to the spatial and frequency localization property of wavelet bases [9]. In this section we make a combination between Fourier descriptors which are invariant to changes in scale, rotation or translation as shown in previous section and wavelet features to recognize the same faces as in previous section Our approach is introduced to overcome the problems of rotation and translation variability of wavelet transforms while retaining their other advantages. The output vector in last section is then compressed using daubechies (s=3) wavelet. As a result of this, the

new vector has a length of only 100 points Thus, the number of neurons in the hidden layer is reduced to 10 neurons So, over a sample of 30 images, the response time is reduced compared to using only Fourier coefficients as shown in Fig 3. Fig. 4 shows some examples for mirrored, noised, occluded, and horizontally/vertically skewed faces which has been recognized correctly using this combination with neural nets. Also, for a threshold value of 0 65 this combination is efficient in rejecting unknown faces

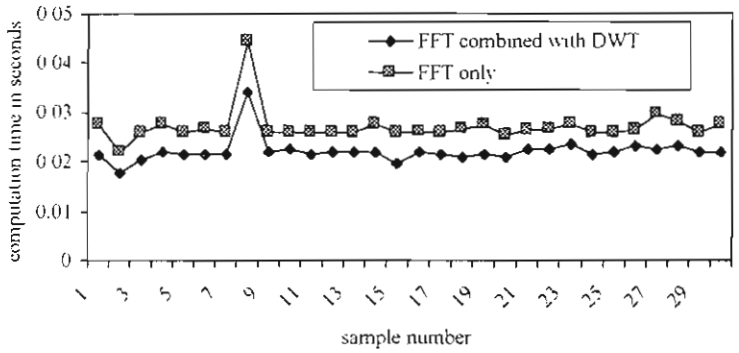

Figure 3. A comparison between the elapsed time taken by FFT only and FFT combined with DWT

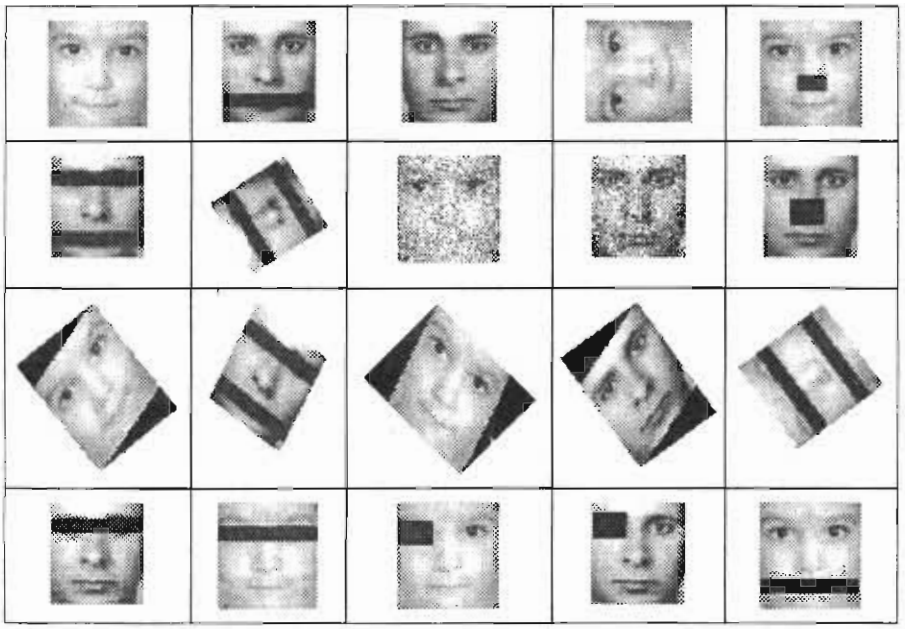

Figure 4 Some examples of mirrored, rotated, noised, deformed, and occluded faces that could be recognized correctly using a combination of FFT and DWT

6. CONCLUSION

A 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