Methodological approach for Face Recognition Using Artificial Neural Networks

K. Shirisha[1], S. Vijaya Lakshmi[2], N. Musrat Sultana[3]

[1,2,3] *Assistant Professor, Computer Science and Engineering Department, MGIT, Hyderabad INDIA*

***Abstract* — Automatic Facial Feature Detection is becoming a very important task in applications such as Model Based Video coding, Facial Image Animation, Face Recognition, Facial Emotion Recognition, Intelligent Human Computer Interaction. Most of the approaches for facial feature detection have been proposed which use independent facial feature detectors relying on hand designed filters that aim at segmenting using image properties such as edges, intensity, color, motion. In this paper, A *Hierarchical neural based facial feature detection* scheme is proposed for robustly and automatically detect a set of user-selected facial features in images that is designed to precisely locate fine features in faces of variable size and appearance.. The proposed system comprises two successive stages: Generalized Rapid Transformation (GRT) and Normalization.**

Keywords: Face Recognition, Facial Feature Detection, Face Detection, Coarse Feature Detection, Fine Feature Detection.

I. INTRODUCTION

Face Recognition [1] has become one of the most challenging tasks in the Pattern Recognition field. The Recognition of faces is very important for many applications such as: video surveillance, retrieval of an identity from a data base for criminal investigations and forensic applications. Among various solutions to the problem [2] the most successful seems to be those appearance-based approaches, which generally operate directly on images or appearances of face objects and process the image as two dimensional patterns.

These methods extract features to optimally represent faces belong to a class and separate faces from different classes. Ideally, it is desirable to use only features having high separability power while ignoring the rest. Most effort in the literature have been focused mainly on developing feature extraction methods and employing powerful classifiers such as probabilistic [3], Hidden Markov Models (HMMs) [4] neural networks (NNs) [5], [6], [7] and support vector machine (SVM) [8].

Face recognition has many applicable areas. Moreover, it can be categorized into Face Identification, Face Classification, or Sex Determination. The most useful applications contain crowd surveillance, video content indexing, personal identification (ex. driver's license), mug shots matching, entrance security, etc.

Image processing is the field of research concerned with the development of computer algorithms working on digitized images (e.g. Pratt, 1991; Gonzalez and Woods, 1992). The range of problems studied in image processing is large, encompassing everything from low-level signal enhancement to high level image understanding.

In general, image processing problems are solved by a chain of tasks. This chain, shown in Figure 1.1 outlines the possible processing needed from the initial sensor data to the outcome (e.g. a classification or a scene description).

The pipeline consists of the steps of Pre-processing, Data Reduction, Segmentation, Object Recognition and Image -Understanding. In each step, the input and output data can either be images (pixels), measurements in images (features), decisions made in previous stages of the chain (labels) or even object relation information (graphs).

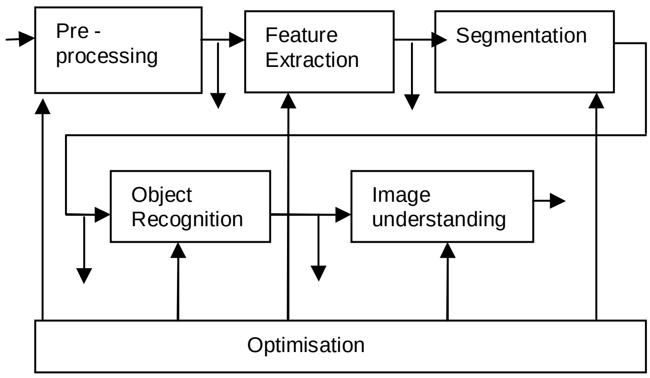


Fig.1 The Image Processing Chain

II. LITERATURE SURVEY

Many people have explored geometrical feature based methods for face recognition. Kanade [17] presented an automatic feature extraction method based on ratios of distances and reported a recognition rate of between 45-75%

with a database of 20 people. Brunelli and Poggio [10] compute a set of geometrical features such as nose width and length, mouth position, and chin shape. They report a 90% recognition rate on a database of 47 people.

However, they show that a simple template matching scheme provides 100% recognition for the same database. Cox et al. [11] have recently introduced a mixture-distance technique which achieves a recognition rate of 95% using a query database of 95 images from a total of 685 individuals. Each face is represented by 30 manually extracted distances.

Systems which employ precisely measured distances between features may be most useful for finding possible matches in a large mugshot database4. For other applications, automatic identification of these points would be required, and the resulting system would be dependent on the accuracy of the feature location algorithm.

Current algorithms for automatic location of feature points do not provide a high degree of accuracy and require considerable computational capacity .

Eigenfaces

High-level recognition tasks are typically modeled with many stages of processing as in the Marr paradigm of progressing from images to surfaces to three-dimensional models to matched models . However, Turk and Pentland [12] argue that it is likely that there is also a recognition process based on low-level, two dimensional image processing. Their argument is based on the early development and extreme rapidity of face recognition in humans, and on physiological experiments in monkey cortex which claim to have isolated neurons that respond selectively to faces. However, it is not clear that these experiments exclude the sole operation of the Marr paradigm.

Turk and Pentland [12] present a face recognition scheme in which face images are projected onto the principal components of the original set of training images. The resulting eigenfaces are classified by comparison with known individuals.

Turk and Pentland present results on a database of 16 subjects with various head orientation, scaling, and lighting. Their images appear identical otherwise with little variation in facial expression, facial details, pose, etc. For lighting, orientation, and scale variation their system achieves 96%, 85% and 64% correct classification respectively. Scale is renormalized to the eigenface size based on an estimate of the head size.

The middle of the faces is accentuated, reducing any negative affect of changing hairstyle and backgrounds. In Pentland et al. [13] good results are reported on a large database (95% recognition of 200 people from a database of 3,000). It is difficult to draw broad conclusions as many of the images of the same people look very similar, and the database has accurate registration and alignment. In Moghaddam and Pentland [14], very good results are reported with the FERET database – only one mistake was made in classifying 150 frontal view images. The system used extensive preprocessing for head location, feature detection, and normalization for the geometry of the face, translation, lighting, contrast, rotation, and scale.

Swets and Weng [15] present a method of selecting discriminant eigen features using multi-dimensional linear discriminant analysis. They present methods for determining the Most Expressive Features (MEF) and the Most Discriminatory Features (MDF).We are not currently aware of the availability of results which are comparable with those of eigenfaces (e.g. on the FERET database as in Moghaddam and Pentland [14]).

Template Matching

Template matching methods such as [10] operate by performing direct correlation of image segments. Template matching is only effective when the query images have the same scale, orientation, and illumination as the training images [11].

Graph Matching

Another approach to face recognition is the well known method of Graph Matching. In [16], Lades et al. present a Dynamic Link Architecture for distortion invariant object recognition which employs elastic graph matching to find the closest stored graph. Objects are represented with sparse graphs whose vertices are labeled with a multi-resolution description in terms of a local power spectrum, and whose edges are labeled with geometrical distances. They present good results with a database of 87 people and test images composed of different expressions and faces turned 15 degrees. The matching process is computationally expensive, taking roughly 25 seconds to compare an image with 87 stored objects when using a parallel machine with 23 transputers. Wiskott et al. use an updated version of the technique and compare 300 faces against 300 different faces of the same people taken from the FERET database. They report a recognition rate of 97.3%. The recognition time for this system was not given.

Neural Network Approaches

Much of the present literature on face recognition with neural networks presents results with only a small number of classes (often below 20). We briefly describe a couple of approaches. In the first 50 principal components of the images are extracted and reduced to 5 dimensions using an auto associative neural network. The resulting representation is classified using a standard multi-layer perceptron. Good results are reported but the database is quite simple: the pictures are manually aligned and there is no lighting variation, rotation, or tilting. There are 20 people in the database.

A hierarchical neural network which is grown automatically and not trained with gradient-descent was used for face recognition by Weng and Huang. They report good results for discrimination of ten distinctive-4 subjects.

III. SYSTEM DESIGN

In this Paper, the facial features are used for recognition process of Human. Instead of dealing with Geometrical Distances the Rapid Transformation is used to extract the features. Here ,the image is portioned in to 16 sub images of 16*16 size and for every sub portion RT Coefficients are computed by sliding the 3*3 RT Window. The Human and Nonhuman classification is done by computing global features from local feature set.

Preprocessed and resized images are used for classification.


Original Image


Resized Image

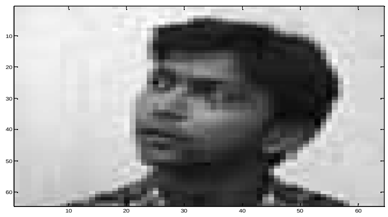

Segmented Image

Fig. 2 Images

IV. IMPLEMENTATION

Generalized Rapid Transformation (GRT)

The Generalized Rapid transform is capable of mapping the object patterns into a series of numbers which does not vary even if the object undergoes translation, scaling and rotation.

To achieve the rapid response and accuracy, the training data is obtained from a preprocessing technique “Generalized Rapid Transformation” and then ANN is trained with this data.

Input Pattern

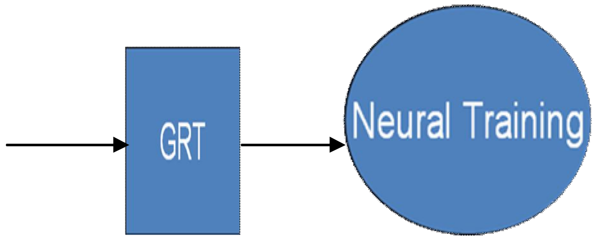


Fig. 3 GRT Algorithm

This supervised training effectively solves the problems associated with the pattern that undergo translation due to dynamism of the environment.

Rapid transformation is also known as Cyclic Shift Invariant Transform. The Rapid Transform (RT) is a fast shift invariant transform.

The RT is useful for pattern recognition, if the position of the pattern is unknown or the pattern is moving. However, the RT eliminates not only knowledge about position but also a lot of information about the original pattern itself.

The RT was introduced by Reitboeck and Brody for application in pattern recognition. The RT results from a simple modification of the Walsh-Hadamard transform. The signal flow diagram of RT is similar to that of WHT except that the absolute value of the output of the each stage of the iteration is taken before feeding it to the next stage.

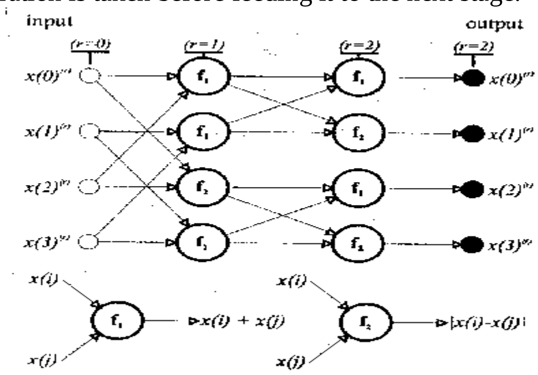


Fig. 4 Signal flow graph of the RT

This is not an orthogonal transform, as no inverse exists. RT is a member of the class CT (certain transforms fast translation invariant transforms). The transforms of the class CT can be divided according to the employed functions/s(a,b), (s=l,2). RT was used in recognition of alphanumeric characters, robotics and scene analysis. More recently the modified rapid transform (MRT) was presented to break undesired invariance of the RT, which leads to a loss of information about the original pattern. MRT was used in pattern recognition and in recognition of 3D objects.

The R-transform is a method for generating translation invariant features from a pattern that can appear anywhere in an image. Such a transform is useful in pattern recognition applications where the position of the pattern to be classified cannot be guaranteed exactly, such as a dollar bill being inserted into a money changer, or a face being presented to a camera for identification or verification. Because a slight shift in position can adversely affect blind, rigid comparison of sensed patterns with stored exemplars on a pixel by pixel basis, the R-transform offers exceptional benefits in the tolerance of position error. In addition to the overwhelming advantage of translation invariance in feature extraction and classification systems, the actual implementation of the R-transform is extremely well-suited for very large-scale integration (VLSI).

Steps in Rapid Transformation

1 Many to one Mapping: Choose the input of length power 2. k=2^n

2 Divide the input into two halves: Perform sum/absolute subtraction on selected elements.

3 Repeat the second step for k stages: Image Segmentation

Segmentation has 2 important advantages for training Neural Network.

- More the features, NN gives the more accurate output.
- It is randomizing the input, the more the randomness in data for Neural Network the more the Neural Network performance.

Each image is segmented in to 16 images of 16*16 sizes. For each segmented image RT values are computed.

Preprocessing of Images

- Originally the images are 100*100 pixels & colored.
- These images are preprocessed by resizing into 64*64 pixels and converted into gray scale.

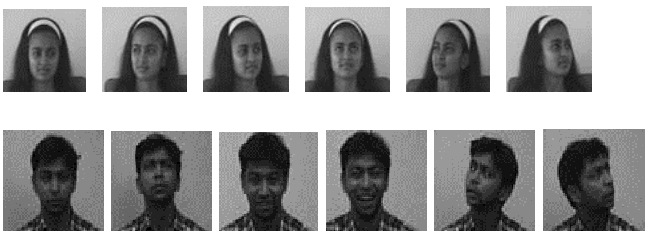


Fig. 5 **Preprocessed Sample Images**

Fig. 6 Pre Processed Non Human Images

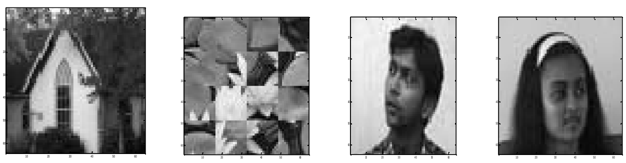


Fig. 7 Segmented Images

ACTIVE PIXELS AND CENTRE OF GRAVITY:

Active pixel is a pixel which contains the effective information of the image.

After calculating the RT values for the segmented images, we use these RT values for finding the Active Pixels in each sub image. If the RT value of four or more neighbourhood pixels is greater than CPI value, then it is considered as a Active Pixel. Like this we calculate the number of active pixels in each segmented image.

TABLE I Sub Portions of a Image

Si-1	Si-2	Si-3	Si-4
Si-5	Si-6	Si-7	Si-8
Si-9	Si-10	Si-11	Si-12
Si-13	Si-14	Si-15	Si-16

Segmented image contains 16 sub images as shown above.

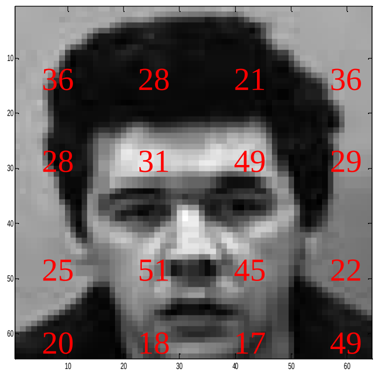


Fig.8 Segmented Image

TABLE II Corresponding Active Pixels in each sub image

36	28	21	36
28	31	49	29
25	51	45	22
20	18	17	49

For each sub image we have to find the Centre of Gravity. The CG is given by summing the all x-coordinate values of all the pixels in the sub image. This value divided by the total number of active pixels in the sub image. Similarly all the y-coordinate values are added and divided by the total number of active pixels in the sub image. We calculate the CGs for the sub images which have the active pixel count greater than zero.

$CG=(Cgx, Cgy)=(x1+x2+…xn/n, y1+y2+….yn/n)$

where 'n' is the total count of active pixels in the sub image.

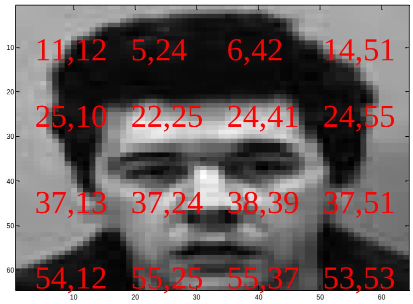


Fig.9 Segmented Image

The corresponding CG values

(11,12)	(5,24)	(6,42)	(14,51)
(25,10)	(22,25)	(24,41)	(24,55)
(37,13)	(37,24)	(38,39)	(37,51)
(54,12)	(55,25)	(55,37)	(53,53)

From these 16 sub images of a Image 3 sub images containing the maximum active pixel count are considered. Then, we consider the respective CG values of the selected 3 sub images. These 3 CG values of each image are normalized and given for the Neural Network.

Active pixel count of the image

36	28	21	36
28	31	49	29
25	51	45	22
20	18	17	49

Images with Active Pixel Count

(11,12)	(5,24)	(6,42)	(14,51)
(25,10)	(22,25)	(24,41)	(24,55)
(37,13)	(37,24)	(38,39)	(37,51)
(54,12)	(55,25)	(55,37)	(53,53)

.

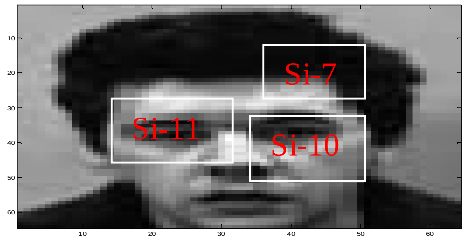


Fig. 10 Images with selected sub portions

NORMALISATION:

After finding the active pixels and CG's for the images, we normalize the CG values and these values are given for training and testing. Training set is different from the testing set data.

The normalization process is:

1. Select the maximum three active pixels from the image.
2. The respective CG values for those sub images are taken.

3 Max active pixel CG values

		X	Y
Human			
	im-1	13	25
		13	39
		53	23
	im-2	17	40
		24	24
		58	27
	im-3	14	26
		13	40
		12	25
	im-4	10	41
		25	23
		13	25
Non-Human			
	im-5	41	10
		53	11
		52	22
	im-6	38	11
		37	25
		36	40
	im-7	9	11
		56	11
		55	54
	im-8	12	37
		55	9
		56	23

Centre of Gravity values

3. The CG of x-coordinate and y-coordinate is obtained.
4. Find the maximum x-coordinate and maximum y-coordinate and minimum x-coordinate and minimum y-coordinate, from all CG values and calculate the K,L values as follows:

K= (max xvalue + min xvalue)/2,

L = (max yvalue + min yvalue)/2.

For example from the above set of values we find maxx=58,maxy=54,minx=9,miny=9. And the K,L values are given as below
K=(58+9)/2=33.5
L=(54+9)/2=31.5
5. Divide all x-coordinates in CGs of all images with K, and all y-coordinates in CGs of all images with L.
For the above set after dividing all x with K, all y with L the new set of values are as shown below

New set of Values

	x/33.5	y/31.5
	X	y
im-1	0.38806	0.793651
	0.38806	1.238095
	1.58209	0.730159
im-2	0.507463	1.269841
	0.716418	0.761905
	1.731343	0.857143
im-3	0.41791	0.825397
	0.38806	1.269841
	0.358209	0.793651
im-4	0.298507	1.301587
	0.746269	0.730159
	0.38806	0.793651
im-5	1.223881	0.31746
	1.58209	0.349206
	1.552239	0.698413
im-6	1.134328	0.349206
	1.104478	0.793651
	1.074627	1.269841
im-7	0.268657	0.349206
	1.671642	0.349206
	1.641791	1.714286
im-8	0.358209	1.174603
	1.641791	0.285714
	1.671642	0.730159

6. Now from this new set of normalized CGs ,find the maximum x-coordinate, maximum y-coordinate. Let us call these values as P,Q respectively.
From the above new set of values maximum x is P=1.731343,maximum y is Q= 1.714286.
7. Now again divide all x-coordinates values of normalized CGs with P, and all y-coordinate values of normalized CGS with Q.
After dividing the above set of values with P,Q, ,the new set of normalized values are as below.

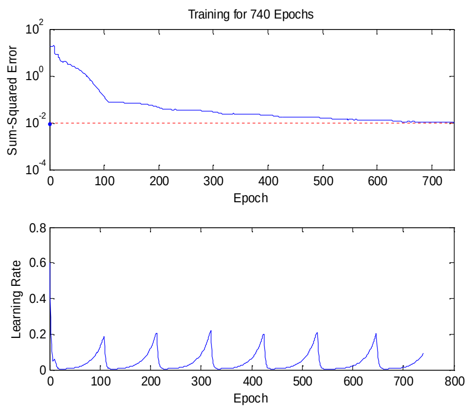
Final Normalized values

	x/1.731343	y/1.714286
	X	Y
im-1	0.224	0.463
	0.224	0.722
	0.914	0.426
im-2	0.293	0.741
	0.414	0.444
	1.000	0.500
im-3	0.241	0.481
	0.224	0.741
	0.207	0.463
im-4	0.172	0.759
	0.431	0.426
	0.224	0.463
im-5	0.707	0.185
	0.914	0.204
	0.897	0.407
im-6	0.655	0.204
	0.638	0.463
	0.621	0.741
im-7	0.155	0.204
	0.966	0.204
	0.948	1.000
im-8	0.207	0.685
	0.948	0.167
	0.966	0.426

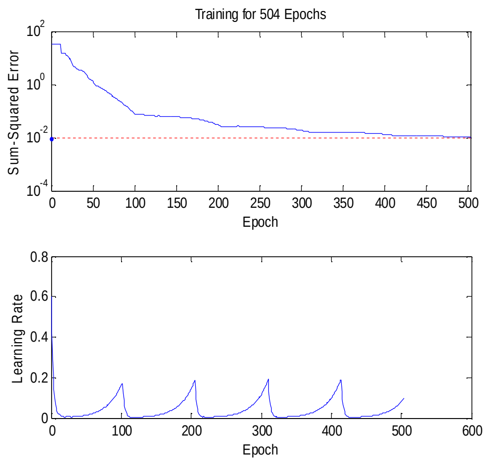
V. EXPERMENTAL RESULTS AND ANALYSIS

Selecting the TRAINING option

Training graph of the Neural Network for 1500 epochs, learning rate 0.5

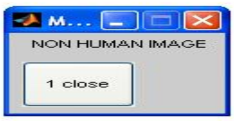


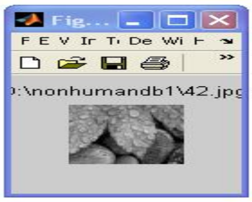
Training graph of the Neural Network for 1500 epochs, learning rate 0.6



After training is completed , the user is allowed to select a image and test the image. When the user selects the 2nd option the list of images are displayed. From this list the user will select one image. Now the user has to select the result option to see the output as follows.

As output, it displays the recognized image and a menu saying whether the selected image is human image or Non Human image. The selected image 42 is recognized as Non human Image as shown below.





If it is a Human image , the menu provides options for further recognition. The selected image and the menu are displayed as follows

VI. CONCLUSION

A methodological approach for Face Recognition provide a high degree of accuracy and require considerable computational capacity. In this paper, the facial features are used for recognition process of Human and also non Human. Instead of dealing with Geometrical Distances we have used the Generalised Rapid Transformation to extract the features. MATLAB has been used for implementation as it has many advantages such as it uses 2-D and 3-D graphics functions for visualizing data. This approach is aimed to develop automated Face Recognition System for fixed size database. The performance of the classification and recognition is 96.153%. The heterogeneous database is used for this classification process. To the most of the extent background and lighting effects are nullified but still in some cases the lighting effect influences the recognition process. The project can be extended for real time environment with the suitable modifications. Captured image is to be suitably preprocessed.

REFERENCES

[1] A. M. Martinez and A. C. Kak, “PCA versus LDA”, *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228-233, 2001.
[2] Pattern Recognition and Image Analysis by Earl Gose, Richard Johnson baugh and Steve Jost.
[3] Yuzuko Utsumi Yoshio Iwai and Hiroshi Ishiguro, “Face Tracking and Recognition Considering the Camera's Field of View", pp. 52-63.
[4] Nefian A. V., Monson, “Hidden Markov Model for Face Recognition,” *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSp98)*, Seatle (1998)2721-2724.
[5] Jun Wang Zhang Yi, Jacek M. Zurada Bao-Liang Lu Hujun Yin, “Advances in Neural Network", *Third International Symposium on Neural Networks*, May/June 2006.
[6] Adnan Khashman, “Face Recognition using Neural Networks and Pattern Averaging” , Dept of EEE, Near East University, Turkey.
[7] Belhumeur P. Hespanha, J. Kreigman, D. J.: Eigenfaces Vs Fisherfaces: “Recognition using Class Specific Linear Projection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 711-720(1997).
[8] Osuna E., Freund . R., Girosit , F, “Training Support Vector Machines: An Application to face detection", *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 130-136,1997.
[9] Ying-li Tian, Takeo Kanade, Jeffrey F. Cohn, "Recognizing action units for facial expression analysis", Pattern Analysis and Machine Intelligence.
[10] Brunelli, R. & T. Poggio, "Face Recognition: Features versus Templates", *IEEE Transactions on PAMI*, 15(10):1042-1052,1993.
[11] David Cox , Nicolas Pinto, "Beyond Simple Features: A Large-Scale Feature Search Approach to Unconstrained Face Recognition"
[12] M. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.
[13] M. Saquib Sarfraz, Olaf Hellwich and Zahid Riaz, "Feature Extraction and Representation for Face Recognition".
[14] B Moghaddam, A Pentland, "Probabilistic Visual Learning for Object Detection", Proc. Int'l Conf. Computer Vision,1995.
[15] D L Swets, J Weng, "Using discriminant eigen features for retrieval", *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1996.
[16] Steve Lawrence, C.Lee Giles, "Face Recognition: A Convolutional Neural Network Approach", *IEEE Transactions on Neural Networks*.