Neural Network Based Approach for Face Detection cum Face Recognition

Kesari Verma, Aniruddha S. Thoke, Pritam Singh

***Abstract*—**Automatic face detection is a complex problem in image processing. Many methods exist to solve this problem such as template matching, Fisher Linear Discriminate, Neural Networks, SVM, and MRC. Success has been achieved with each method to varying degrees and complexities. In proposed algorithm we used upright, frontal faces for single gray scale images with decent resolution and under good lighting condition. In the field of face recognition technique the single face is matched with single face from the training dataset. The author proposed a neural network based face detection algorithm from the photographs as well as if any test data appears it check from the online scanned training dataset. Experimental result shows that the algorithm detected up to 95% accuracy for any image.

***Keywords*—**Face Detection, Face Recognition, NN Approach, PCA Algorithm.

I. Introduction

FACE detection tasks are becoming required more frequently in the modern world. It's caused by the development of high security systems to prevent from terrorism and frauds. In addition to this algorithms are widely used in companies, video coding and for high security purpose. Face detection is initial phase of face recognition. According to Yang's survey [13], these methods can be categorized into four types: knowledge-based, feature invariant, template matching and appearance-based document. Knowledge-based methods use human-coded rules to model facial features, such as two symmetric eyes, a nose in the middle and a mouth underneath the nose.

1) Feature invariant methods try to find facial features which are invariant to pose, lighting condition or rotation. Skin colors, edges and shapes fall into this category.
2) Template matching methods calculate the correlation between a test image and pre-selected facial templates. Nilamani Bhoi et. al. [6] has proposed template based eye detection algorithm in which first the template of eye is taken in gray scale. The normalized eye template is cross correlated with overlapping region of face. The mean square error of auto correlation is calculated ad minimum MSE is stored in memory
3) Appearance-based, adopts machine learning techniques to extract discriminative features from a pre-labeled training set. The Eigenface method is the most fundamental method in this category. Recently proposed face detection algorithms such as support vector machines, neural networks, and statistical classifiers also belong to this category.

The block diagram of detection algorithm working is shown in Fig 1.

K. Verma is with the National Institute of Technology, Raipur -India (e-mail: keshriverma@gmail.com).

A.S. Thoke is working in Department of Electrical Engineering, in National Institute of Technology, Raipur- India (asthoke@yahoo.co.in).

P. Singh is Student of M. Tech, Department of Electrical Engg., National Institute of Technology, Raipur-India (email: pritamsn@gmail.com).

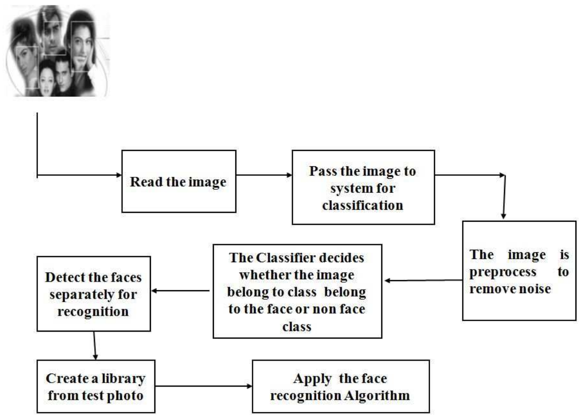

Fig. 1 Block diagram of Face detection and recognition algorithm

II. Literature Survey

In Literature we found many novel methods have been proposed to solve the problem. For example Craw, D. Tock et. al. [1] has proposed the template-matching methods in which the template of face is convolute with original image in order to detect the faces. Lanitis et. al. [2] has given flexible model appearance approach for automatic face identification; author used correlation coefficient of an input image to a standard face pattern. The feature invariant approaches are used for feature detection [3], [4] of eyes, mouth, ears, nose, etc. The appearance based algorithm by detecting eigen face was introduced by [5], [6], [7]. Neural network based face detection and information retrieval approach was proposed by [8],[9]. Currently many researchers are working on ICA based face detection algorithm in order to improve the accuracy.

The work done by various author can be summarized by Table I.

TABLE I
Literature Review

S No	Approach		Authors
1	Feature based approach	Low level analysis edges	Sakai et al. [16], Craw *et al.* [17] L. C. De Silva et. al. [19] Govindaraj [20] Herpers [21]
		Gray level Information	facial feature extraction algorithms Van Beek et. al.[22] H. P. Graf, [23] K. M. Lam [24]
2	Feature Analysis	Feature Searching	P. J. L. Van [25] J. L. Crowley [26] H. P. Graf [27]

Angela Chau, et. al. [14] have discussed the problem of variable size face on images. Some faces are very large and include necks, while other are obstructed or are very small. In order to find all the faces of different size of image the algorithm applied by author [4] is as belows.

1) Convolve the mask with a large head-and-neck shaped template. Example of several connected faces in the mask

image, even after performing erosion and dilation to separate.

2) Find the peak value resulting from the convolution, and subtract a dilated version of the template from the location of the peak value.
3) Repeat steps 1 and 2 until the peak value falls below a certain threshold
4) Repeat steps 1 to 3 with smaller and smaller head-and-neck shaped templates.

Existing bootstrapping method for face detection work as follows.

A. Boot Strapping Procedure

1) Reshape the whole training set in One Dimensional Dataset
2) Computing the average all face image
3) Calculating the deviation of each image from mean image
4) Computing the difference image for each image in the training se Ai = Ti – m
5) A = [A temp]; % Merging all centered images
6) L = A'*A; % L is the surrogate of covariance matrix C=A*A'.
7) [V D] = eig(L); % Diagonal elements of D are the eigenvalues for both L=A'*A and C=A*A'.
8) Sorting and eliminating eigenvalues. All eigenvalues of matrix L are sorted and those who are less than a specified threshold are eliminated. So the number of non-zero eigenvectors may be less than (P-1). Calculating the eigenvectors of covariance matrix 'C' Eigenvectors of covariance matrix C (or so-called "Eigenfaces") can be recovered from L's eiegnvectors. Eigenfaces = A * L_eig_vec; % A: centered image vectors
9) Eigenfaces = A * L_eig_vec; % A: centered image vectorsonvolve the mask with a large head-and-neck shaped template. Example of several connected faces in the mask image, even after performing erosion and dilation to separate.
10) Find the peak value resulting from the convolution, and subtract a dilated version of the template from the location of the peak value.

III. Neural Network Based Classifier

The proposed face detection method passes two phases first phase is preprocessing of image, second phase is train the system with Neural Network.. Images presented to the network go through a pre-processing step to improve the classification rate of the system. In order to equalize the intensity values of the pixels present in the examination window and histogram equalization is performed to expand the range of intensities, in order to compensate for differences in lighting and camera input gains. Using the following formula.

$$P_x(i) = p(x = i) = \frac{n_i}{n},\ 0 \le i \prec n \tag{1}$$

The cumulative distribution function corresponding to p_x as

$$cdf_x(i) = \sum_{j=0}^{i} p_x(j) \tag{2}$$

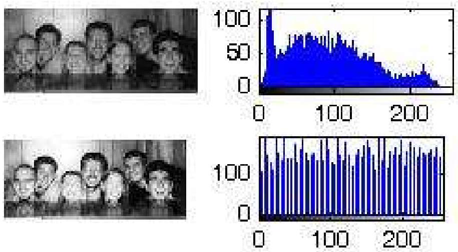

Fig. 2 Pre-processing phase for Images

The operation of the face detection through system can be broken down into three main steps.

- Initialization (design and creation of a neural network)
- Training (choice of training data, parameters, and training)
- Classification (scanning images to locate faces)
- A feed forward neural network is created which is trained using back propagation. The training set used contains examples of both face and non-face images, and the classifier is trained to output a value between 0.9 and -0.9. 0.9 represent existence of faces and -0.9 represent non existence of faces.

In training phase resize each of the image of size 27x18[16].

A. Algorithm for Detecting the Faces

1) Different mirror of the images and shifting of the images are collected and stored for faces image.
2) Features of faces images are collected through fourier transformation and inverse fourier transformation.
3) Step (1) and (2) are applied for nonfaces images also.
4) Initalize the neural network parameters.
   numInputs: 1
   numLayers: 2
   transferFcn = 'tansig';
   net.trainParam.lr = 0.4;
   net.trainParam.epochs = 1000;
   net.trainParam.show = 200;
   net.trainParam.goal = 1e-5;
5) call matlab train function(Network Input, Target input);
6) Input a Test Image
7) Compute the min and Max
8) max_ = max(max(input));
9) min_ = min(min(input));
10) $output = (\frac{(input - \min)}{(\max - \min)} - .5) * 2$
11) Compute the Minmax of scan Image, template1 image and emplate2 image.
12) Apply the convolution operator using both the templates.
13) Apply the trained net to detect the images we can obtained the point.
14) Create the window based on obtained points

The result obtained from above algorithm is saved in training dataset; whenever a test data appears it apply template based matching and principle component analysis in order to recognize the right face from the training dataset.

B. PCA Algorithm

1) Reshape the whole training set in One Dimensional Dataset
2) Computing the average all face image
3) Calculating the deviation of each image from mean image
4) 4. Computing the difference image for each image in the training se Ai = Ti – m
5) A = [A temp]; % Merging all centered images
6) L = A'*A; % L is the surrogate of covariance matrix C=A*A'.
7) [V D] = eig(L); % Diagonal elements of D are the eigenvalues for both L=A'*A and C=A*A'.
8) Sorting and eliminating eigenvaluesAll eigenvalues of matrix L are sorted and those who are less than a specified threshold, are eliminated. So the number of non-zero eigenvectors may be less than (P-1). Calculating the eigenvectors of covariance matrix 'C' Eigenvectors of covariance matrix C (or so-called "Eigenfaces") can be recovered from L's eiegnvectors. Eigenfaces = A * L_eig_vec; % A: centered image vectors
9) Eigenfaces = A * L_eig_vec; % A: centered image vectors.

C. Matching Algorithm for Images

1) temp = Eigenfaces'*A(:,i); % Projection of centered images into facespace
2) ProjectedImages = [ProjectedImages temp]; // Append all the projected Imageread the test Image
3) Reshape the test image Centered test image
4) Find out difference from test image and mean
5) Difference = double(InImage)-m;
6) ProjectedTestImage = Eigenfaces'*Difference; % Test image feature vector
7) For each image in the training set
8) Normalize the image and sqr it
9) temp = (norm(ProjectedTestImage - q))^2;
10) Find out min Equclidean Distance from temp
11) The image that is minimum euclidean distanc matching image.

IV. Experimental Results

The experiment for proposed system is performed in Windows operating system , Pentium V machine, 340 GB memory , 2.33 GHz processor, 1 GB RAM and software Matlab2010 with image processing and Neural Network toolbox. . The images are extracted in .png and .jpg format. Database contain 220 face image, 220 non face images, the snaps contain 87 images. The experimental results are shown in Table II, Table III, Table IV and Figure 3.The experimental results shows that if the number of image for test dataset is less the system elapsed time is less, when the number of faces increase in the image the scanning time increases relatively.

In experiment we also observe that number of false positive detected are very less in comparison to correct positively classified image. The comparison is represented in graph Fig 3. As the graph shows that false detection rate increase when the number of faces increases.

TABLE II
THE TRAINING TIME VS TESTING TIME

Sno	Training Time (second)	Itera-tions	Performance	Time taken (Second)
Im1	2:01	196	0.000973	27.60545
Im2				58.75241
Im3				60.25845
Im4				977.5568
Im1	1:58	181	0.000956	19.7961
Im2				46.35752
Im3				78.11271
Im4				875.343

TABLE III
ACCURACY OF ALGORITHM

S No	#No. of faces in Image	#No. of faces detecte d	Correctly detected	#false Positive	Accurac y
Img1	7	7	7	-	100%
Img2	15	14	14	1	93.3%
Img3	8	7	7	1	87.5%
Img4	45	49	43	2	95.5%
Img1	7	5	-	2	71.4%
Img2	15	12	-	3	80%
Img3	8	8	1	-	100%
Img4	45	52	40	7	86.53%

TABLE IV
FACE DETECTOR LEARNING PERFORMANCE

Number of faces example	220
Correct classification rate	85.06%
Number of incorrect Classification rate	26.40%

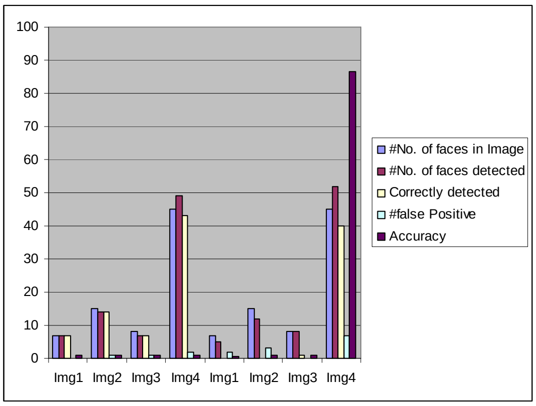

Fig. 3 Number of Correctly classified and false classified image

V. Conclusion

In this paper we elaborates neural network based algorithm to identify the faces in any images. Due to wide application of the system for security purpose, Theft detection this problem is one of the thrust area. The Neural Network algorithm works better if the training set is large if we are decreasing the size of training set the results are not very good. We extracted feature of the images using transformation method. We can optimize the different features of image to improve the accuracy of algorithm. There are various existing algorithm that works either for face detection or face recognition, but the proposed approach do the both work. For detecting the faces we got up to 85.06% of accuracy. The rate of misclassification is 26.40%. After the face detection the next phase when we give any face for testing it uses principal component analysis method with selecting Eigen value in order to recognize the faces.

For recognizing we experimented the rotated image 90,180 250, 45 degrees.

The proposed method is able to recognize the faces in different rotations and different poses. For recognition we got up to 86% accuracy. In experiment we found that for same training set it perform gives good accuracy but if change the snap shot the result is not very much effective. The future enhancement is applying this process for colored images.

This work has various applications like fraud detection in bank using CCD Cameras photographs.

TABLE V
ACCURACY TABLE ON DIFFERENT NEURON

No. of **Neurons.**	M1(7) +VE Corre	M2(15) +Ve correc	M3(8) +VE Corre	M4(57) +VE Corre	Positive Correct / Total image	Accuracy in %
100	6	15	7	51	79/87	90.8 %
200	6	13	7	52	78/87	89.6 %
300	5	13	7	52	77/87	88.5 %
400	6	13	8	48	75/87	86.2 %
500	6	13	7	48	74/87	85.0 %
600	7	12	7	47	73/87	83.9 %
700	7	12	7	47	73/87	83.9 %
800	7	12	7	47	73/87	83.9 %
900	7	12	7	48	74/87	85.0 %
1000	6	14	6	48	74/87	85.0 %

REFERENCES

[1] Craw, D. Tock, and A. Bennett, “Finding face features,” Proc. of 2nd European Conf. Computer Vision. pp. 92-96, 1992.

[2] A. Lanitis, C. J. Taylor, and T. F. Cootes, “An automatic face identification system using flexible appearance models,” Image and Vision Computing, vol.13, no.5, pp.393-401, 1995.

[3] T. K. Leung, M. C. Burl, and P. Perona, “Finding faces in cluttered scenes using random labeled graph matching,” Proc. 5th IEEE int’l Conf. Computer Vision, pp. 637-644, 1995.

[4] B. Moghaddam and A. Pentland, “Probabilistic visual learning for object recognition,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no.7. pp. 696-710, July, 1997.

[5] M. Turk and A. Pentland, “Eigenfaces for recognition,” J. of Cognitive Neuroscience, vol.3, no. 1, pp. 71-86, 1991.

[6] M. Kirby and L. Sirovich, “Application of the Karhunen-Loeve procedure for the characterization of human faces,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol.12, no.1, pp. 103-108, Jan. 1990.

[7] I. T. Jolliffe, Principal component analysis, New York: Springer-Verlag, 1986.

[8] T, Agui, Y. Kokubo, H. Nagashi, and T. Nagao,“Extraction of face recognition from monochromatic photographs using neural networks,” Proc. 2nd Int’l Conf. Automation, Robotics, and Computer Vision, vol.1, pp. 18.81-18.8.5, 1992.

[9] Krestinin, I.A., Seredin, O.S.: Excluding cascading classifier for face detection. Proc. of the 19th Int. Conf. on Computer Graphics and Vision, 380–381 (2009).

[10] Kienzle, W., Bakir, G., Franz, M., Scholkopf, B.: Face detection – efficient and rank deficient, Advan. in neural inform. process. systems 17, 2005. – P. 673–680 (2005).

[11] Nikolay Degtyarev and Oleg Seredin, Tula State University. Comparitive testing of face detection algorithm The original publication is available at www.springerlink.com.

[12] Jesorsky, O., Kirchberg, K.J., Frischholz, R.W.: Robust face detection using the hausdorff distance, Lecture Notes in Computer Science, June, 90-95 (2001).

[13] M. H. Yang and N. Ahuja. Detecting huamn faces in color images. IEEE Proc. of Int. Conf. on Image Proc. (ICIP ’98), 1998.

[14] Nilamani Bhoi, Mihir Narayan Mohanty Matching based Eye Detection in Facial Image. International Journal of Computer Applications (0975 – 8887)Volume 12– No.5, December 2010.

[15] http://www.facedetectioncode.com

[16] T. Sakai, M. Nagao, and T. Kanade, Computer analysis and classification of photographs of human faces, in Proc. First USA—Japan Computer Conference, 1972, p. 2.7.

[17] I. Craw, H. Ellis, and J. R. Lishman, Automatic extraction of face-feature, Pattern Recog. Lett. Feb. 1987, 183–187.

[18] L. C. De Silva, K. Aizawa, and M. Hatori, Detection and tracking of facial features by using a facial feature model and deformable circular template, IEICE Trans. Inform. Systems E78–D(9), 1995, 1195–1207.

[19] V. Govindaraju, Locating human faces in photographs, Int. J. Comput. Vision 19, 1996.

[20] R. Herpers, H. Kattner, H. Rodax, and G. Sommer, G aze: An attentive processing strategy to detect and analyze the prominent facial regions, in IEEE Proc. of Int. Workshop on Automatic Face- and Gesture-Recognition, Zurich, Switzerland, Jun. 1995, pp. 214–220.

[21] P. J. L. Van Beek, M. J. T. Reinders, B. Sankur, and J. C. A. Van Der Lubbe, Semantic segmentation of videophone image sequences, in Proc. of SPIE Int. Conf. on Visual Communications and Image Processing, 1992, pp. 1182–1193.

[22] H. P. Graf, E. Cosatto, D. Gibson, E. Petajan, and M. Kocheisen, Multi-modal system for locating heads and faces, in IEEE Proc. of 2nd Int. Conf. on Automatic Face and Gesture Recognition, Vermont, Oct. 1996, pp. 277–282.

[23] K. M. Lam and H. Yan, Facial feature location and extraction for computerised human face recognition, in Int. Symposium on information Theory and Its Applications, Sydney, Australia, Nov. 1994.

[24] P. J. L. Van Beek, M. J. T. Reinders, B. Sankur, and J. C. A. Van Der Lubbe, Semantic segmentation of

[25] videophone image sequences, in Proc. of SPIE Int. Conf. on Visual Communications and Image Processing,1992, pp. 1182–1193.

[26] J. L. Crowley and F. Berard, Multi-model tracking of faces for video communications, in IEEE Proc. of Int,Conf. on Computer Vision and Pattern Recognition, Puerto Rico, Jun. 1997.

[27] H. P. Graf, T. Chen, E. Petajan, and E. Cosatto, Locating faces and facial parts, in IEEE Proc. of Int.Workshop on Automatic Face-and Gesture-Recognition, Zurich, Switzerland, Jun. 1995, pp. 41–45.