Face Detection and Recognition in an Image Sequence using Eigenedginess

B S Venkatesh, S Palanivel and B Yegnanarayana
Department of Computer Science and Engineering.
Indian Institute of Technology, Madras
Chennai - 600 036
{venkatesh,spal,yegna}@cs.iitm.ernet.in

Abstract

This paper describes a system for face detection and recognition in an image sequence. Motion information is used to find the moving regions, and probable eye region blobs are extracted by thresholding the image. These blobs reduce the search space for face verification, which is done by template matching. Eigen analysis of edginess representation of face is used for face recognition. One dimensional processing is used to extract the edginess image of face. Experimental results for face detection show good performance even across orientation and pose variation to a certain extent. The face recognition is carried out by cumulatively summing up the Euclidean distance between the test face images and the stored database, which shows good discrimination for true and false subjects.

1. Introduction

Face detection and recognition are challenging tasks due to variation in illumination, variability in scale, location, orientation (up-right, rotated) and pose (frontal, profile). Facial expression, occlusion and lighting conditions also change the overall appearance of face. Face detection and recognition has many real world applications, like human/computer interface, surveillance, authentication and video indexing.

Face detection using artificial neural networks was done by Rowley [7]. It is robust but computationally expensive as the whole image has to be scanned at different scales and orientations. Feature-based (eyes, nose, and mouth) face detection is done by Yow *et al.*[15]. Statistical model of mutual distance between facial features are used to locate face in the image [4]. Markov Random Fields have been used to model the spatial distribution of the grey level intensities of face images [1]. Some of the eye location technique use infrared lighting to detect eye pupil [2]. Eye location using genetic algorithm has been proposed by Wechsler [3] . Skin color is used extensively to segment the image, and localize the search for face [13, 12]. The detection of face using skin color fails when the source of lighting is not natural. In this paper, motion information is used to reduce the search space for face detection. It is known that eye regions are usually darker than other facial parts, therefore probable eye pair regions are extracted by thresholding the image. The eyes pair region gives the scale and orientation of face, and reduces the search space for face detection across different scales and orientations. Correlation between averaged face template and the test pattern is used to verify whether it is a face or not.

Recognition of human face is also challenging in human-computer interaction [6, 10, 11, 14]. The proposed system for face recognition is based on eigen analysis of edginess representation of face, which is invariant to illumination to certain extent [8, 9]. The paper is organized as follows: Section 2 describes the face detection process. The method of obtaining edginess image and eigenedginess of a faces are discussed in Sections 3 and 4, respectively. Experimental results are presented in Section 5.

2. Face Detection

In an image sequence the position of the head is not stationary, as there is always some motion. Therefore the regions having significant motion are extracted by subtracting consecutive frames and thresholding it. Figure 1 shows two consecutive frames from a video sequence. Let I_t represent the image at time t. The difference image D is given by

$$D(i,j) = \begin{cases} 1, & if\ (I_t(i,j) - I_{t-1}(i,j)) > \lambda \\ 0, & otherwise \end{cases} \quad (1)$$

where (i,j) are the (row,column) indices of the image, and λ is the threshold, which is set such that D is zero when there is no significant motion. The contour in the difference image D is traced to find an approximate bounding box. The corresponding region in image I_t is referred as E. Figure 2 shows the thresholded difference image D with the bounding box, and the corresponding grey level image E. Since objects other than face can also be in motion, a decision has to be made whether it is a face or not. Correlation between the averaged face template and the test pattern is computed, and the test pattern is accepted as face if the correlation exceeds a certain threshold. To localize the face in the image E, the face template has to be passed over the image with different scales and orientations. To speed up the process, first the possible region of the eye pair is extracted. It is known that the eye region is usually darker than other facial parts such as nose and mouth. To extract these dark regions, a threshold γ is calculated as

$$\gamma = \mu_E - \sigma_E \tag{2}$$

where μ_E is the mean and σ_E is the standard deviation of the image E. Let B be the binary image given by

$$B(i,j) = \begin{cases} 1, & if\ E(i,j) < \gamma \\ 0, & otherwise \end{cases} \tag{3}$$

where (i, j) are the (row,column) indices of the image E. Figure 3(a) shows the binary image B. Morphological operators are used to close small discontinuities in the image B. This results in blobs as shown in Figure 3(b). Along with the possible eye pair region there may be other dark regions which are extracted. The size of the eye region is approximately known, and larger regions than this size are filtered out. Figure 3(c) shows the image after region filtering. The pair of blob gives the orientation of face, which is used to normalize the test pattern to the size of the face template. Correlation between the face template and test pattern is computed. If the correlation exceeds certain threshold level, it is accepted as face. This eliminates the search for face in all scales and orientations. Figure 3(d) shows the detected face resized to 50x50 pixels after template matching. Figure 4 shows the detected faces for 5 different subjects.

Figure 1: Two consecutive frames from a video.

3. Edginess Image of Face

To extract the edginess image of a face, computationally efficient method of one dimensional (1-D) processing of images proposed in [5] is used. In this method, the image is smoothed using a 1-D Gaussian filter along the horizontal (or vertical) scan lines to reduce noise. A differential operator (first derivative of 1-D Gaussian function) is then used in the orthogonal direction, i.e., along the vertical (or horizontal) scan lines to detect the edges. This method differs from the traditional approaches based on 2-D operators in the sense that smoothing is done along one direction and the differential operator is applied along the orthogonal direction. The traditional 2-D operators smooth the image in all directions, thus resulting in smearing of the edge information.

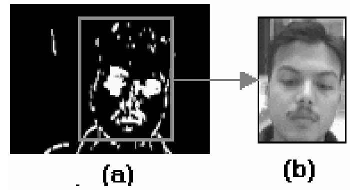

Figure 2: (a) Thresholded difference image D and (b) corresponding grey level image E.

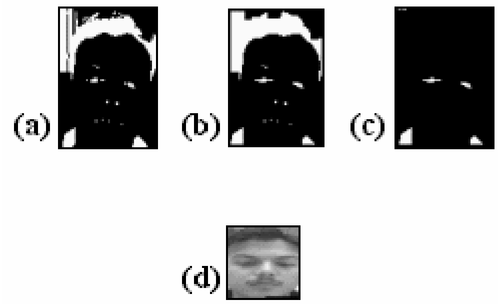

Figure 3: (a) Thresholded image B.(b) Image after morphological operator applied to B.(c) Image after region filtering.(d) Detected face resized to 50x50 pixels after template matching.

The smoothing filter is a 1-D Gaussian filter, and the differential operator is the first order derivative of the 1-D Gaussian function. The 1-D Gaussian filter is given by

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{x^2}{2\sigma_1^2}} \tag{4}$$

where σ_1 is the standard deviation of the Gaussian function. The first order derivative of 1-D Gaussian is given by

$$c(y) = \frac{-y}{\sqrt{2\pi}\sigma_2^3} e^{-\frac{y^2}{2\sigma_2^2}} \tag{5}$$

where σ_2 is the standard deviation of the Gaussian function. The smoothing filter and the differential operator are shown in Figure 5. The values of σ_1 and σ_2 decide the spatial extent of these 1-D filters.
The response of the 1-D Gaussian filter applied along a particular scan line of an image in one direction (say, along the horizontal scan line y_r of pixels) can be expressed as

$$h(x, y_r) = k(x, y_r) * g(x) \tag{6}$$

Figure 4: Result of face detection (faces resized to 50x50 pixels).

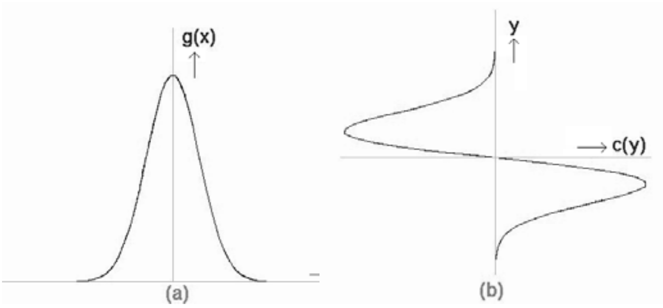

Figure 5: (a) Gaussian function in the horizontal direction (smoothing filter) and (b) first derivative of Gaussian function in the vertical direction (differential operator).

where $*$ denotes the 1-D convolution operator, $g(x)$ represents the 1-D Gaussian filter, $k(x, y_r)$ represents the r^{th} row of the image k, and $h(x, y_r)$ is the corresponding filter response. The response is computed for all rows in the image to obtain $h(x, y)$.
For the 1-D Gaussian filter output $h(x, y)$, obtained using Equation 6 for all the rows, the differential operator is applied along each column x_c to extract the edges oriented along the horizontal lines of the pixels. The result is given by

$$f(x_c, y) = h(x_c, y) * c(y) \tag{7}$$

where $c(y)$ is 1-D differential operator, and $h(x_c, y)$ denotes the c^{th} column in the 1-D Gaussian filtered image $h(x, y)$. The resulting image $f(x, y)$, obtained by applying Equation 7 for all columns, produces the horizontal components of edginess (strength of an edge) in the image. Similarly, the vertical components of edginess are derived by applying the 1-D smoothing operator along all the vertical scan lines of the image and further processing with the 1-D differential operator along the horizontal scan lines of pixels. Finally, the partial edge information obtained in both the horizontal and vertical directions are added to extract the edginess map of the original image. Figure 6 shows a grey level image, binary edge image and edginess image of a face. It is obvious that the edginess image carries additional information which is missing in the binary edge image. The edginess of a pixel in an image is identical to the magnitude of the gradient of the grey level function, which corresponds to the amount of change across the edge. Hence capturing directly the gradual variation present in a facial image is better and accurate than constructing the edginess image artificially from the edge image of the face.

Grey level image   Edge image   Edginess image

Figure 6: Different representations of facial image.

4. Eigenedginess

Consider a set of P sample images $I^p_{r\text{x}c}$, p = 1, 2, ..., P, with resolution $r\text{x}c$. The pixels in the image are vectorized into a N-dimensional vector $\mathbf{x}_p$, p = 1, 2, ..., P, where $N = r$ x c. The vectors obtained in this manner from all the P sample images can be denoted as $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots\ldots, \mathbf{x}_P\}$.
For a given set of N-dimensional vector representation of faces, principal component analysis (PCA) can be used to find the subspace whose basis vectors correspond to the directions of maximum variance in the original space. Let W represent the linear transformation that maps the original N-dimension space onto a M-dimension feature subspace, where $M << N$. This yields a set of projection vectors, $\mathbf{y}_p \,\epsilon\, \mathcal{R}^{\mathcal{M}}$, where $\mathbf{y}_p = W^T\mathbf{x}_p$, $p = 1, \ldots, P$. The columns of W are the M eigenvectors $\mathbf{e}_i$ corresponding to the first M eigenvalues obtained by solving the eigen equation, $C\,\mathbf{e}_i = \lambda_i\mathbf{e}_i$, where $C = \sum_{p=1}^{P}(\mathbf{x}_p - \boldsymbol{\mu})(\mathbf{x}_p - \boldsymbol{\mu})^T$ is the covariance matrix, λ_i is the eigenvalue associated with the eigenvector $\mathbf{e}_i$, and $\boldsymbol{\mu} = \frac{1}{P}\sum_{p=1}^{P}\mathbf{x}_p$.

The reduced dimension representation of the edginess image of a face is determined using the PCA technique. Eigenvectors of the covariance matrix of the edginess images are referred as $eigenedginess$.
In an image sequence, number of face images are captured for training. For each of the training face vector $\mathbf{x}_{l,u}$, the projection vector is given by

$$\mathbf{y}_{l,u} = W^T\mathbf{x}_{l,u} \quad 1 \le l \le L,\ 1 \le u \le U \tag{8}$$

where L is the number of subjects, and U is the number of face images per subject. The identity of the person (ψ)

from Q test face vectors $\mathbf{x}_q$, $q = 1, \ldots, Q$, is calculated as follows:

$$\mathbf{y}_q = W^T \mathbf{x}_q \tag{9}$$

$$d_{l,q} = \min_u ||\mathbf{y}_q - \mathbf{y}_{l,u}||^2 \tag{10}$$

$$\psi = arg(\min_l (\sum_{q=1}^{Q} d_{l,q})) \tag{11}$$

where $\mathbf{y}_q$ is the projection vector of the q^{th} test face vector, and $d_{l,q}$ is the minimum Euclidean distance of test face image q from the l^{th} subject.

5. Experimental Results

The experiment is conducted with 5 subjects. For each subject 30 faces are captured to form the training set. Similarly for the testing data set, 30 faces per subject were collected on a different day. Each face image is resized to 50x50 pixels. The edginess image of the face is calculated as described in Section 3. To reduce the dimension of the vector, the first 20 eigenvectors of the edginess images, are used. The test face pattern is classified by taking the minimum Euclidean distance between the stored pattern and the test pattern in the eigenedginess space. Face recognition results for a total of 150 test face patterns for 5 subjects (30 faces per subject) is shown in Figure 7. The graph shows the number of faces classified into a particular class for each subject. For test face patterns of subject 1 the graph shows that 23 faces were recognized correctly as subject 1, none were recognized as subject 2, 4 faces were recognized as subject 3, 2 faces were recognized as subject 4, and 1 face was recognized as subject 5. The performance is 86% for 150 faces. Figure 8 shows the minimum Euclidean distance plot for the test face patterns (subject 4) against all subjects. Figure 9 shows the cumulative Euclidean distance for the test face patterns of subject 4. The cumulative sum of Euclidean distance for the test face patterns gives better discrimination than from a single test face pattern.

6. Conclusions

In this paper we presented a face detection and localization technique in a video. To speed up the process of face detection, motion information is used, and probable eye pair regions are extracted, which guides the template matching for face verification. With this approach, scanning the image for different scales and orientation is avoided. In our method eigen analysis of edginess representation of face is used for recognition. For each subject, 30 face images are captured from the video, and the face is recognized based on minimum cumulative sum of the Euclidean distances,

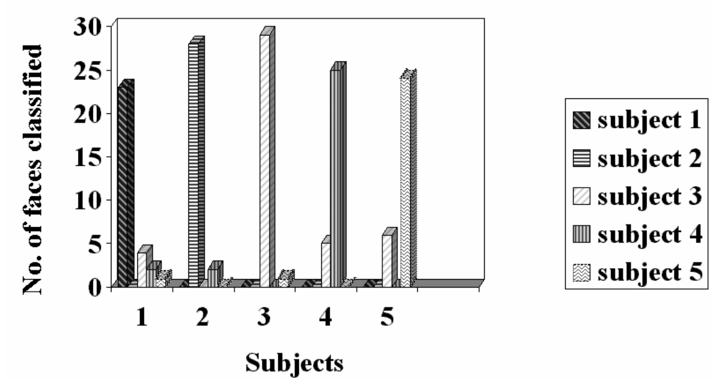

Figure 7: Face recognition performance. The bar graph shows the number of faces classified out of 30 test face patterns for each subject

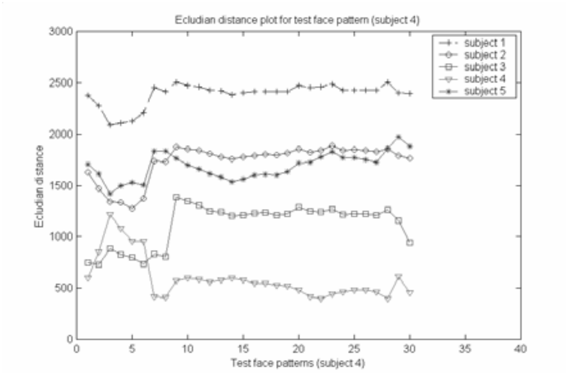

Figure 8: Minimum Euclidean distance plot for test face pattern (subject 4) against 5 different subjects

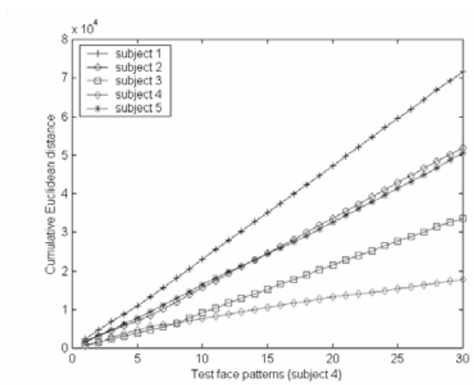

Figure 9: Cumulative Euclidean distance plot for test face pattern (subject 4) against 5 different subjects

which gives better performance than the distance from a single face image.

References

[1] S. C. Dass and A. K. Jain. Markov face models. In *Proceedings,Eighth IEEE International Conference on Computer Vision (ICCV)*, pages 680–687, July 2001.

[2] C.-C. Han, H.-Y. M. Liao, G.-J. Yu, and L.-H. Chen. Fast face detection via morphology-based pre-processing. In *Proceedings, Ninth International Conference on Image analysis and Processing (ICIAP)*, volume 2, pages 469–476, 1998.

[3] J. Huang and H. Wechsler. Eye location using genetic algorithm. In *Proceedings, Second International Conference on Audio and Video-Based Biometric Person Authentication*, pages 130–135, March 1999.

[4] T. Leung, M. Burl, and P. Perona. Finding faces in cluttered scenes using labeled random graph matching. In *Proceedings, Fifth International Conference on Computer Vision*, pages 637–644, June 1995.

[5] P. Kiran Kumar, Sukhendu Das and B. Yegnanarayana. One-Dimensional processing of images. In *International Conference on Multimedia Processing Systems, Chennai, India*, pages 181–185, August 13-15, 2000.

[6] P. N. Belhumeour, J. P. Hespanha and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):711–720, July 1997.

[7] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(1):23–38, January 1998.

[8] S. Ramesh, S.Palanivel, Sukhendu Das and B. Yegnanarayana. Eigenedginess vs. eigenhill, eigenface, and eigenedge. In *Proceedings, Eleventh European Signal Processing Conference*, pages 559–562, September 3-6, 2002.

[9] K. Suchendar. *Online face recognition system*. M.Tech Project Report, IIT, Madras, January 2002.

[10] B. Tacacs and H. Wechsler. Face recognition using binary image metrics. In *Proceedings, Third International Conference on Automatic Face and Gesture Recognition*, pages 294–299, April 1998.

[11] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *Proceedings, Eleventh International Conference on Pattern Recognition*, pages 586–591, 1991.

[12] J. Yang and A. Waibel. A real-time face tracker. In *Proceedings, Third IEEE Workshop on Applications of Computer Vision*, pages 142–147, 1996.

[13] M. H. Yang and N. Ahuja. Detecting human face in color images. In *Proceedings, IEEE International Conference on Image Processing*, volume 1, pages 127–130, 1998.

[14] Yilmaz, Alper and M. Gokmen. Eigenhill vs. eigenface and eigenedge. *Pattern Recognition*, 34:181–184, 2001.

[15] K. C. Yow and R. Cipolla. Feature-based human face detection. *Image and Vision Computing*, 15(9):713–735, 1997.