Eigenedginess vs. eigenhill, eigenface and eigenedge

S. Ramesh, S. Palanivel, Sukhendu Das and B. Yegnanarayana
Department of Computer Science and Engineering
IIT Madras, Chennai - 600 036, INDIA
E-mail: mr_sriramesh@yahoo.com, {spal@cs, sdas@, yegna@}.iitm.ernet.in

ABSTRACT

Recognizing human faces, invariant to illumination and facial expressions is a difficult task. This paper proposes a method of face recognition using eigen analysis on edginess-based representation of the face. One dimensional processing of images is used to extract the edginess of the face. Experimental results show that the edginess-based representation performs better than other edge and gray level based representations under variation in illumination and facial expressions. This paper also discusses the effect of enhancing suitable edginess values and elimination of first few eigenvectors on the recognition performance.

1 Introduction

Automatic recognition of human faces is one of the challenging problems in human-computer interaction [1]-[4]. Number of methods has been used in the literature such as geometry of facial features [5], neural networks [1], template matching [2], Karhunen-Loeve transform [3], Principal Component Analysis(PCA) or eigenface [4], Fisherface [6], eigenedges [7], eigenhills [8]. Some of these methods are sensitive to variation in illumination or facial expressions. The eigenhill approach creates an artificial edginess of the face. The method proposed in this paper uses actual edginess information of the face for face recognition. The method of obtaining edginess and eigenedginess of a face are discussed in Sections 2 and 3, respectively. Performance of face recognition using transformed edginess image, effect of first few eigenvectors and variations in facial expressions are discussed in Sections 4, 5 and 6, respectively.

2 Edginess of face

To extract the edginess image of a face, computationally efficient method of one dimensional(1-D) processing of images proposed in [9] 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.

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

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

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

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

where σ_2 is the standard deviation of the Gaussian function. The smoothing filter and differential operator are shown in Fig.1. The values of σ_1 and σ_2 decides the spatial extent of these 1-D filters. In this study, the values of σ_1 and σ_2 are chosen to be 0.3 and 0.6 respectively, although these values are not critical.

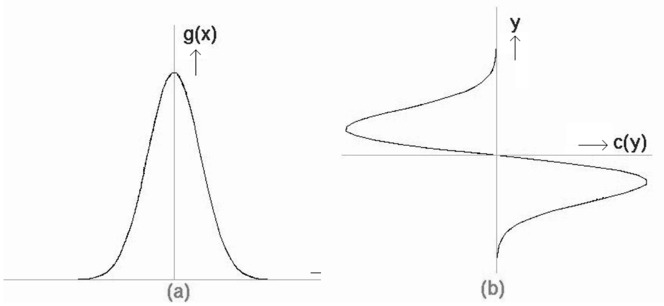

Figure 1: (a) Gaussian function in the horizontal direction (smoothing filter) and (b) first derivative of Gaussian function in the vertical direction (differential operator).

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

$$h(x, y_r) = i(x, y_r) * g(x) \quad (3)$$

where $*$ denotes the 1-D convolution operator, $g(x)$ represents the 1-D Gaussian filter, $i(x, y_r)$ represents the r^{th} row of the image i, 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 (3) 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

$$f(x_c, y) = h(x_c, y) * c(y) \quad (4)$$

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 (4) 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.

Fig.2 shows a gray level image, binary edge image and edginess image of a face. It is obvious that the edginess image carries more 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 gray 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 2: Different representations of facial image.

3 Eigenedginess

Consider a set of P sample images $I^p_{r\text{x}c}$, $p = 1, 2, \ldots, P$, with resolution $r\text{x}c$. The pixels in the image are vectorized into a N-dimensional vector $\mathbf{x}_p$, $p = 1, 2, \ldots, P$, where $N = r \text{ x } c$. 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 $\boldsymbol{\mu} = \frac{1}{P} \sum_{p=1}^{P} \mathbf{x}_p$.

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*. Images of 40 individuals were used for experimental studies, with three face samples for each individual with different illumination conditions. The face images from the database were manually cropped and scaled to a fixed size of 35x35, to exhibit only the features of the face necessary for recognition. Only one image per individual, obtained with natural lighting condition, is used in the training set. The test set consists of two images per individual, one illuminated from the left and the other from the right side. The recognition performance of eigenface, eigenedge, eigenhill and eigenedginess are shown in Table 1. It shows that the performance of eigenedginess is significantly better compared to the other three methods. From these results, it appears that edginess-based representation is more robust to variation in illumination than the other three representations of face.

Category	Faces recognized (out of 80)
Eigenface	14
Eigenedge	24
Eigenhill	21
Eigenedginess	56

Table 1: Results of eigen analysis.

4 Transformed edginess

The discriminatory information for faces is mostly present in the gradual variation in a face, which is reflected in the low edginess values of the face. The transformation function as shown in Fig.3 is used to deemphasize high values of edginess and at the same time preserve the low edginess values. The shape of the function can be determined from the histogram of the edginess images. This transformation function can be viewed as a combination of 3 parts, where the transformation of the edginess values less than x_1 is of sigmoidal type, edginess values between x_1 and x_2 is of linear type, and edginess values greater than x_2 is again of sigmoidal type. This is expressed as

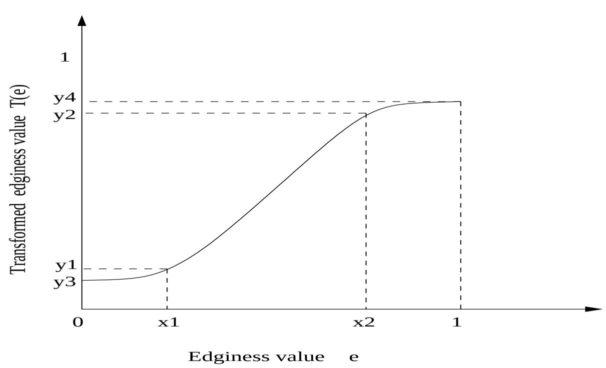

Figure 3: Transformation function used to enhance suitable edginess values.

$$T(e) = \begin{cases} \frac{1}{1+exp[-(e-\theta_1)\lambda_1]}, & 0 \leq e \leq x_1 \\ m*e+c, & x_1 < e \leq x_2 \\ \frac{1}{1+exp[-(e-\theta_2)\lambda_2]}, & x_2 < e \leq 1 \end{cases}$$

where $0 \leq e \leq 1$ is the normalized edginess value, λ_1 and λ_2 are the slope parameters, θ_1 and θ_2 are the positional parameters, and m and c are parameters for the linear part. These parameters can be derived by imposing the conditions: $T(0)=y_3$, $T(x_1)=y_1$, $T(x_2)=y_2$ and $T(1)=y_4$. Given x_1, x_2, y_1, y_2, y_3 and y_4, the parameters of $T(e)$ are given as:

$$\begin{aligned} \lambda_1 &= \frac{1}{x_1}\left[ln\left(\frac{y_1}{1-y_1}\right) - ln\left(\frac{y_3}{1-y_3}\right)\right] \\ \theta_1 &= \frac{-1}{\lambda_1} ln\left(\frac{y_3}{1-y_3}\right) \\ \lambda_2 &= \frac{1}{x_2-1}\left[ln\left(\frac{y_2}{1-y_2}\right) - ln\left(\frac{y_4}{1-y_4}\right)\right] \\ \theta_2 &= 1 + \frac{1}{\lambda_2} ln\left(\frac{1-y_4}{y_4}\right) \\ m &= \frac{y_2-y_1}{x_2-x_1} \\ c &= \frac{x_2 y_1 - y_2 x_1}{x_2-x_1} \end{aligned}$$

Experimental results show that this kind of transformation function provides improvement in the recognition performance while using fewer principal components or eigenvectors.

5 Effect of the first few eigenvectors

Pentland *et al* [10] have shown that the performance of face recognition improves when the first three eigenvectors are ignored. This is because the first three eigenvectors seem to represent variation due to illumination. The performance of face recognition on different representations after eliminating the first few eigenvectors are given in Table 2. This study is performed on 80 faces with variation in illumination. The results show that the performance of eigenedginess is significantly better than the other representations.

Eigen vectors eliminated	Eigen face	Eigen edge	Eigen hill	Eigen edginess
0	14	24	21	56
1	29	**26**	**23**	56
2	**30**	25	20	**66**
3	27	24	17	62
4	26	21	19	57
5	28	23	15	56

Table 2: Results of eigen analysis on different representations by eliminating first few eigenvectors.

6 Recognition performance due to variation in facial expressions

In this study, a separate set of faces of 100 individuals taken from the standard FERET database is considered for evaluating the recognition performance on various representations. The subset of these faces having normal facial expressions are used for training and the remaining faces having variation in facial expressions are used for testing. The recognition performance on different representations are shown in Table 3. It can be observed that the performance of the eigenedginess is better than eigenhill and eigenedge, and is comparable with that of the eigenface.

Category	%
Eigenedginess	**93**
Eigenhill	77
Eigenedge	47
Eigenface	**94**

Table 3: Results of eigen analysis on a dataset having variation only in facial expression.

7 Summary

In this paper, the concept of edginess of an image is introduced for the purpose of face recognition, which is invariant to illumination and facial expressions. Experimental results show that edginess-based representation for face recognition performs significantly better than eigenface, eigenedge and eigenhill representations. Performance of face recognition using transformed edginess image and the effect of first few eigen vectors are also discussed.

References

[1] M. Fleming and G. W. Cottrell, "Categorization of faces using unsupervised feature extraction," in *IEEE Proceedings of the International Joint Conference on Neural Networks*, pp. 65–70, San Diego, Vol. II 1990.

[2] R. Brunelli and T. Poggio, "Face recognition: features versus templates," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042–1052, Oct. 1993.

[3] 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.

[4] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuro-Science*, vol. 3, pp. 71–86, 1991.

[5] M. D. Kelly, "Visual identification of people by computer," *Tech. Rep. AI-130, Stanford AI Proj., Stanford, CA*, 1970.

[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*, vol. 19, no. 7, pp. 711–720, July 1997.

[7] B. Tacacs and H. Wescler, "Face recognition using binary image metrics," in *Proceedings of Third Int. Conf. on Automatic Face and Gesture Recognition, Nara, Japan*, pp. 294–299, April 1998.

[8] Yilmaz, Alper and M. Gokmen, "Eigenhill vs. eigenface and eigenedge," *Pattern Recognition*, vol. 34, pp. 181–184, 2001.

[9] P. Kiran Kumar, Sukhendu Das and B. Yegnanarayana, "One-Dimensional processing of images," in *Int. Conf. on Multimedia Processing Systems, Chennai, India*, pp. 181–185, Aug. 13-15, 2000.

[10] A. Pentland, T. Starner, N. Etcoff, N. Masoiu, O. Oliyide and M. Turk, "Experiments with eigenfaces," in *Proc. Looking at people Workshop Int. Joint Conf. Artificial Intelligence*, Aug. 1993.