EVALUATION OF SIMILARITY AMONG HUMAN FACES

Vasif Nabiyev[1], Yasemin Bekiroğlu[2]

Karadeniz Technical University, Trabzon, Turkey
[1]*vasif@ktu.edu.tr*, [2]*yaseminbekiroglu@gmail.com*

Abstract

This paper contains the examination of determining similarity between human faces. Detection of similarity can be defined as listing faces similar to a given face and to be able to achieve it, a system should imitate the talent with which humans recognize faces despite a lot of factors. Defining a way of representing faces taking possible changes into account and classifying a given face using the chosen way are the required operations.

1. Introduction

Automatic recognition of people is a challenging problem considering different factors such as illumination, pose, emotional state, and aging. It has become a popular area of research and utilized in different fields such as human-computer interaction, security applications or video indexing. In such systems, to be able to increase the efficiency, it is significant to detect and evaluate the similar faces properly. Detection of similarity involves identifying different images belonging to one person and also detecting similar images belonging to different people as illustrated in Figure 1.

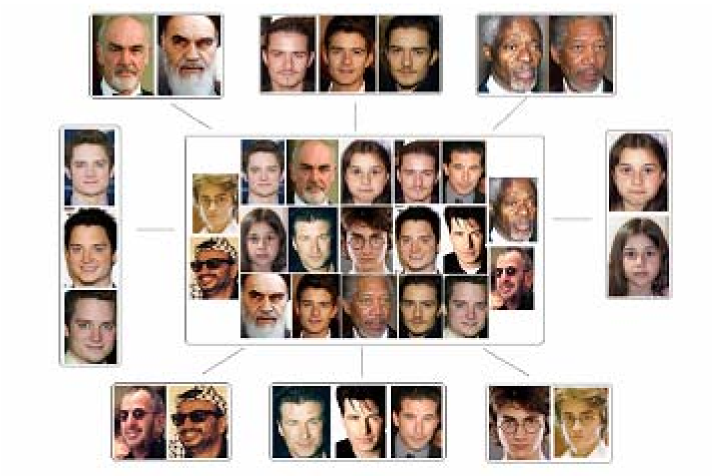

Figure 1. Similarity detection

2. Description of the System

To obtain faces to decide how similar they are, firstly face detection on a given image is performed. The system contains two parts, face detection and evaluation of similarity. In the face detection part of the work, a multilayer neural network (MNN) is implemented [1, 2, 3, 4]. The network is trained to distinguish faces from non-faces. In the training phase, the samples in the training set (280 face, 320 non-face pre-processed images) were presented into the network, the desired output values for the two group were set as (1,0) respectively and the phase was completed when the total error had a smaller value than the threshold value of 0.001. After training the network, to detect all faces image containing faces is scanned while it is scaled down to the size of the samples in the training set, 25x25, step by step preserving the height/width ratio of the original image. In each step, a window with the size of 25x25 is travelled on the image and its content is presented into the network. If the computed value of the output of the network is greater than a specified threshold, the window whose content is fed into the network is covering a face in its current position and its corner coordinates are recorded after being scaled up to the original size. Accordingly, very close windows covering the same face might appear in the result. In such a case, since the windows are close enough to be included in the same group, they are reduced to one window which is the mean of the others.

In face detection, since possible faces on input image are included on the skin part, skin parts were focused to decrease the image part to be searched for faces. With this aim, to detect skin parts input image is converted into YCbCr color space. Each pixel is examined to decide whether or not it is a skin pixel according to its color components and a binary image is generated and possible noise values in the skin regions found are eliminated through one erosion operation and finally, contour of the skin region is found. After detecting the contours on the binary image, separate groups are covered by rectangles and the ones whose size are too small which can not contain a face are eliminated. The result obtained in the skin detection is processed in examination based on the neural network in the face detection part. These steps are illustrated in the Figure 2.

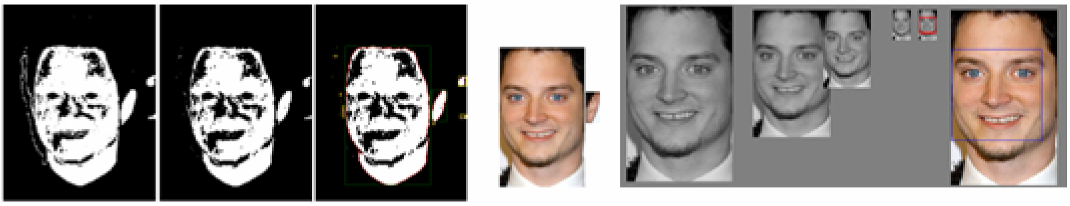

Figure 2. Face detection steps

Principal Component Analysis [5, 6, 7, 8] is the method preferred in similarity evaluation part of the work. Principal Component Analysis provides an efficient way of comparing faces according to the similarities and differences between them representing faces on a lower dimensional space. Four distance measures, voting based on them were utilized in evaluation of the results obtained in Principal Component Analysis.

Listing similar faces in images included in a database designed to be used in similarity evaluation was performed successfully and most similar faces to a given face which was not included in the database could also be listed.

3. Principal Component Analysis (PCA)

PCA provides an orthogonal projection basis leading to dimensionality reduction and feature selection. PCA involves the calculation of the eigen-value decomposition of a data covariance matrix, usually after mean centring the data for each attribute. The data to which PCA is to be applied contains the images. The data matrix is constructed by placing image vectors into the columns. Image vectors have the row pixels in images in order. Namely, if there are M images of size N*N, then the matrix X has the size of N^2*M, $X = [\, x_1\ x_2\ \dots\ x_M \,]$, $x_i = [\, d_1\ d_2\ \dots\ d_{N^2} \,]^T$, $i=1,2,\dots, M$. The data, Q, is prepared for the covariance matrix calculation by subtracting the mean, $avr = \frac{1}{M} * \sum_{i=1}^{M} x_i$, $q_i = x_i - avr$. The data, Q, is to be transformed by using the eigenvectors of the covariance matrix of the data. Since the size of the covariance matrix, $Cov(Q) = QQ^T$, is N^2*N^2 and the number of eigen vectors of this matrix is N^2, generating these eigen vectors, (u_i), has a high computational burden ($Cov(Q) = \frac{1}{M} * \sum_{i=1}^{M} q_i q_i^T = QQ^T$ (N^2*N^2)). Therefore, the eigenvectors are derived from the eigenvectors of the matrix Q^TQ, $Q^TQ\, v_i = \mu_i v_i$, where v_i is the eigen vectors and μ_i is the eigen-values of Q^TQ. The size of Q^TQ is M*M and it has M eigen vectors with M components. When both sides of the equation are multiplied by Q, $Q\,Q^TQ\, v_i = \mu_i\, Qv_i$ is obtained where u_i =

Qv_i is the eigen vector of QQ^T (Qv_i is normalized in order to have $\| Qv_i \| = 1$) and μ_i is the eigen value. The data is to be expressed in terms of these eigen vectors. Eigen vectors are ordered by eigen value from highest to lowest to get the components in order of significance. $x<<M<<N^2$ eigen vectors are chosen and the data is transformed by using them, $y_i = u_i^T q_j$, $i=1, \ldots, x$; $j=1,\ldots, M$, $Y_j=[y_1\ y_2 \ldots y_x]^T$,$Y = [Y_1, Y_2, \ldots Y_M]$. y_i is the i th component of the transformed image Y_j . Small eigen-values and their associated eigenvectors are ignored using a threshold value, $(\sum_{i=1}^{\chi} \lambda_i / \sum_{i=1}^{M} \lambda_i) > 0.9$. The lower dimensional vector u captures the most expressive features of the original data. In this way, the data is compressed by reducing the number of dimensions without much loss of information. The comparison after transformation can be performed by using a distance measure like normalized Euclidian.

$$\sum_{i=1}^{K} \frac{1}{\lambda_i} (\chi_i - \chi_i^*)^2 \tag{1}$$

Preprocessing is applied to images before PCA. Scaling them to the size of 100*100, converting to gray level, equalization of the average pixel value and applying a mask are the preprocessing steps. An example for these steps is given below (Figure 3a). In this way a database with 31 samples where there is at least 2, at most 5 images are available for each person is constructed (Figure 3b). 15 eigen vectors are chosen to transform the data, since their eigen values were big enough (Figure 3c). In Figure 3d, eigen vectors can be seen as images after they are scaled into the range of [0,255].

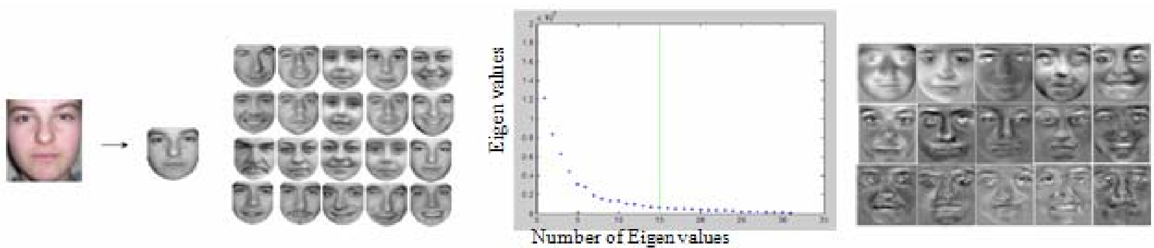

Figure 3. (a) Preprocessing example, (b) Some samples from database, (c) Eigen values, (d) Eigen vectors

4.2. Results

When PCA is applied to the images in the database and the relationships among transformed images are compared by using normalized Euclidian distance, the success rate is 80% (Figure 4a). In the calculation of success rate, the positions of the images in the result list are used to find the contribution values for each person then the average of them gives the final rate. If all the images belonging to the person in the input image appear in the first places in the result list consecutively, that sample contributes to the success rate with the value of 1. For instance, if there are 3 images in the database belonging to the person in the input image and 2 of them appear in the first 3 places in the result list, the value to be added to the resulting rate becomes 2/3. In Figure 4, for each sample in the database evaluation results are given according to the calculations explained. In order to have a different interpretation, a voting strategy (Borda count, [9]) based on different similarity measures which are normalized Euclidian (1), Euclidian (2), City-Block (3) and Chess-board (4) distance is also used. In this way, the success rate is 91%, (Figure 4c).

$$(\sum_{i=1}^{K} (\chi_i - \chi_i^*)^2)^{\frac{1}{2}} \tag{2}$$

$$\sum_{i=1}^{K} |\chi_i - \chi_i^*| \tag{3}$$

$$max(|\chi_i - \chi_i *|), i = 1,...,K \quad (4)$$

In order to cover the all the images which do not appear in the expected places in the result, a slightly different way of calculation can be followed. For instance, if for the input image, there are 3 images belonging to the same person and in the result list the images appear in the 1st, 2nd and 4th places, the value (31+30+28)/(31+30+29) = 0.98 is added to the success rate calculation for this sample. For normalized Euclidian distance, the success rate in this way becomes 95%, (Figure 4b). Similarly, according to voting, for the same example the value is calculated as (6+5+3)/(6+5+4) based on 6 votes and the final rate is %98 (Figure 4d). When the database is expanded to have 65 samples, the success rate based on voting is 83%, (Figure 4e).

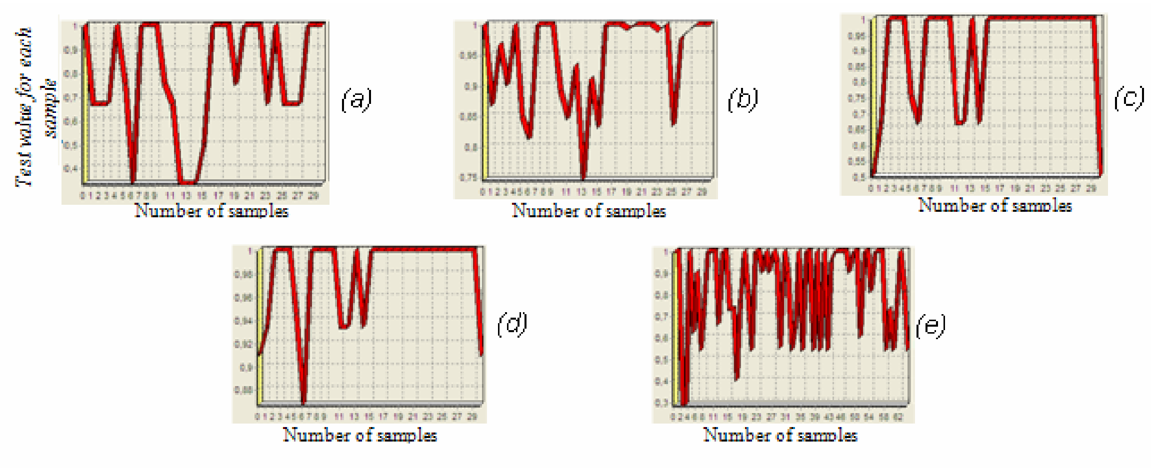

Figure 5. Experimental results

5. Conclusion

A system based on PCA and MNN is presented for evaluation of similarity among human faces in images. Experimental results on an example database demonstrate the validity of the system. In the future, more work should be done to apply the method to more complex images.

Literature

1. H.A. Rowley, S. Baluja, and T. Kanade, *Neural Network-Based Face Detection*, In Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition, CVPR, (1996), 203-207.
2. H.A. Rowley, S. Baluja, and T. Kanade, *Rotation Invariant Neural Network-Based Face Detection*, In Proceedings of IEEE Conference on Computer Vision and PR, (1998), 38-44.
3. M. Negnevitsky, *Artificial Intelligence A Guide to Intelligent Systems*, 1. Ed., Addison-Wesley, England, (2002).
4. E. Öztemel, *Yapay Sinir Ağları*, 1. Basım, Papatya Yayıncılık, İstanbul, (2003).
5. www.snl.salk.edu/~shlens/pub/notes/pca.pdf *A tutorial on PCA,* 24 April (2007).
6. M. Turk, A. Pentland, *Eigenfaces for Recognition*, Journal of Cognitive Neurosci., 3, 1 (1991) 71-86.
7. M. Turk, *A Random Walk through Eigenspace*, IEICE Trans. Inf. & Syst., E84-D, 12, (2001) 1586-1595.
8. M. Turk, A. Pentland, *Face Recognition Using Eigenfaces*, Proc. of the IEEE Conf. On Computer Vision and Pattern Recognition, (1991), 586-591.
9. V.V. Nabiyev, *Algorithms: From Theory to Applications*, 1. Ed., Seçkin Pub, Ankara, in Turkish, (2007), 799 p.