Journal of Soft Computing and Artificial Intelligence

Journal homepage: https://dergipark.org.tr/en/pub/jscai

International Open Access

Volume 03
Issue 02

December, 2022

Research Article

Can Similarity Measures Techniques be Used to Model Face Recognition?

***Enes Algül*[1]**

[1]*Bingol University, School of Engineering and Architecture, Department of Computer Science, Bingöl, Türkiye*

ARTICLE INFO

Article history:
Received December 12, 2022
Revised December 14, 2022
Accepted December 16, 2022

Keywords:
Distance Methods
Classification
Eigenface
Facial Recognition
PCA

ABSTRACT

Facial recognition is used efficiently in human-computer interactions, passports, driver's licence, border controls, video surveillance and criminal identification, and is an important biometric's security option in many device-related security requirements. In this paper, we use Eigenface recognition based on the Principal Component Analysis (PCA) to develop the project. PCA aims to reduce the size of large image matrices and is used for feature extraction. Then, we use the euclidean distance method for classification. The dataset used in this project was obtained by AT&T Laboratories at Cambridge University [1]. The training dataset contains grayscale facial images of 40 people; each person has 10 different facial images taken from different angles and emotions.

This study aims to give researchers a hunch before they start to develop image recognition using deep learning methods. It also shows that face recognition can be done without deep learning.

1. Introduction

The combination of two Greek words, Bios (life) and Metron (measures) generate the term Biometrics [2]. Physical attributes and biometric features are unique to each person, and that separates each person apart from the others. These features include fingerprint [3], face, hand, iris [4], speech, smell, and gait. Biometric identity, in other words, the biometric signature cannot be easily stolen, forgotten and difficult to be copied by someone else. Therefore, it provides higher and more reliable security in identifying the person. Some biometric features, such as the iris [5], are stable over time and fingerprints never change from childhood to the end of life. In this work, face recognition has been researched in greater depth with a broader range of techniques.

This paper aims to investigate existing biometric security techniques and principles of facial recognition. The research question is: "Can similarity measurement techniques be used to model the face authentication system?".

In this work, Face recognition was developed with Eigenface recognition based on PCA [6]. For this system, 400 images of 40 individuals have been trained. Then 40 known and 15 unknown facial images have been tested to check the accuracy. The project performance was measured by two standard rate methods: False Acceptance Rate (FAR) and False Rejection Rate (FRR).

The distance methods were researched for classification. Euclidean, Manhattan and Mahalanobis distance methods are the three methods examined in this study. The Euclidean distance method was chosen to measure the distances between facial images. Then, the threshold value was selected by multiplying the maximum value of the Euclidean Distance by 0.8. The threshold value was obtained as a result of many experiments. This multiplication process was inspired by Nasser Abouzakhar's work [7] and used in this project.

[1] Corresponding Author
e-mail: enes.algul12@gmail.com

2. Related Work

2.1. Principle Component Analysis (PCA)

PCA has been used in a variety of studies including computer science, meteorology, and neuroscience. PCA [8–10] was first used for facial recognition by Kirby and Sirovich in 1987. Later, Matthew A. Turk and Alex P. Pentland discovered a new facial recognition technique using Eigenface based on PCA in 1991 [11]. Their invention has been used for static and automatic real-time face recognition. This invention has helped to develop a face recognition system with fewer matrices multiplications. The purpose of using PCA is to reduce the size of large image sets (matrix). For example, suppose there are M grayscale images in a dataset for training, and the size of the images are $N \; x \; N$. In order to build a matrix of images, $2D$ images need to be converted to $1D$. Therefore, each image is represented by $1D$ a column vector, and the size of this column is N^2. There are M images, and if they are stored into one matrix, the size of this matrix will be $N^2 \; x \; M$. Their covariance matrix needs to be calculated to find relationships between each column vector. Then, this covariance matrix will be used to extract eigenvectors and eigenvalues.

Let's represent the matrix of the dataset with A.

Covariance Matrix $= AA' = (N^2 \; x \; M)(M \; x \; N^2)$;

The size of the covariance matrix $= (N^2 \text{ x } N^2)$. This is too large to manage. For example, if the number of images is 400 and the size of each image is $256 \; x \; 256$, then the size of the covariance matrix $=$ $(65536 \; x \; 65536)$. With the PCA technique, instead of Covariance $= AA'$, Covariance $= AA' =$ $(M \; x \; N^2)(N^2 \; x \; M)$ is calculated. Consequently, the dimension of the Covariance matrix is $=$ $(400 \; x \; 400)$ and $(M << N^2)$. This size of the matrix is easier to manage.

2.2. Eigenface Technique

The eigenface technique [12,13] is a set of eigenvectors commonly used to recognize the human face [14]. The eigenvalues and eigenvectors are derived from the covariance matrix. The eigenface [15,16] can be obtained by multiplying the eigenvectors with the normalized matrix.

Eigenvectors can only be applied to square matrices. The covariance matrix in this work is square $(400 \; x \; 400)$. The dominant eigenvectors in the covariance matrix correspond to the higher value and represent more characteristics and features. Eigenvectors were sorted by the eigenvalues. After the eigenvalues are sorted in descending order, the first eigenvectors have more characteristic features than the second one. The last eigenvector is showing the least characteristic features. Therefore, k number of eigenvectors were chosen heuristically to train the dataset. In this paper, 124 out of 400 eigenvectors were selected, and other eigenvectors were neglected. Then, the eigenfaces were calculated.

2.3. Threshold Value

Threshold [13] value is chosen heuristically. There is no specific method developed to find the threshold value. Commonly, it is chosen by multiplying 0.8 by the maximum values of minimum values of Euclidean distance.

In this article, the purpose of using threshold value is to decide whether the newly inputted image is a face image or not. If the distance between the inputted image and training images is less than the threshold value, the inputted images will have more facial features. Therefore, we can say that if it holds less than the threshold value it is identified as a face image. For this reason, choosing the threshold value affects the performance and result of the accuracy.

2.4. Distance Methods

To measure the distances between images for comparison purposes, three methods are commonly used. The equations of Euclidean, Manhattan, and Mahalanobis Distances are as follows:

Euclidean Distance:

$$d(X,Y) = \sqrt{\sum_{i=1}^{M}(X_i - Y_i)^2} = ||X - Y|| \quad (1)$$

Manhattan Distance:

$$d(X,Y) = \sum_{i=1}^{M} |X_i - Y_i| \quad (2)$$

Mahalanobis Distance:

$$d(X,Y) = \sqrt{(X_i - Y_i)^T C^{-1} (X_i - Y_i)} \quad (3)$$

where C refers to the covariance matrix.

3. Dataset

The dataset contains 40 individual persons' images, and each person has 10 images, each one taken from different angles and emotions. The dataset was prepared by Cambridge University at AT&T laboratories. Each image has two dimensions and is a grayscale image as seen in Figure 1. All images are in the same sizes. Each image has a width of 112 pixels and a height of 92 pixels

Figure 1 The first image of each person in the dataset.

4. Algorithm

Part 1:

Step 1: Prepare a data set for training (*40* persons' *10* individual images, a total of *400* images).

Step 2: Resize whole images at the same size and convert them to grayscale images ($2D$).

Step 3: Convert each image in the dataset from $2D$ matrix ($112 \, x \, 92$) to $1D$ column vector ($10304 \, x1$). Then build a new matrix ($10304 \, x \, 400$) where each column represents an images of the dataset. Here, I refers to $2D$ graysacle original images, Γ denotes $1D$ images.

$$\Gamma = (\Gamma_1, \Gamma_2, \Gamma_3, \ldots, \Gamma_M)$$

where $M = 400.$

Step 4: Compute the average column Ψ.

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \quad (4)$$

Step 5: Normalise the data set. Subtract the average column vector from each column of the dataset's matrix.

$$\Phi = \Gamma_i - \Psi \quad (5)$$

$$A = (\Phi_1, \Phi_2, \Phi_3, \ldots, \Phi_M) \quad (6)$$

Here, Φ denotes adjusted columns, and $A(10403 \, x \, 400)$ denotes the normalised matrix.

Step 6: Calculate the covariance (C) matrix. Use $C = A^t A$ instead of $C = AA^t$ to reduce the dimensional of the matrix (use PCA).

Step 7: Calculate eigenvalues (V) and eigenvectors (D).

$$[V \, D] = eig(C) \quad (7)$$

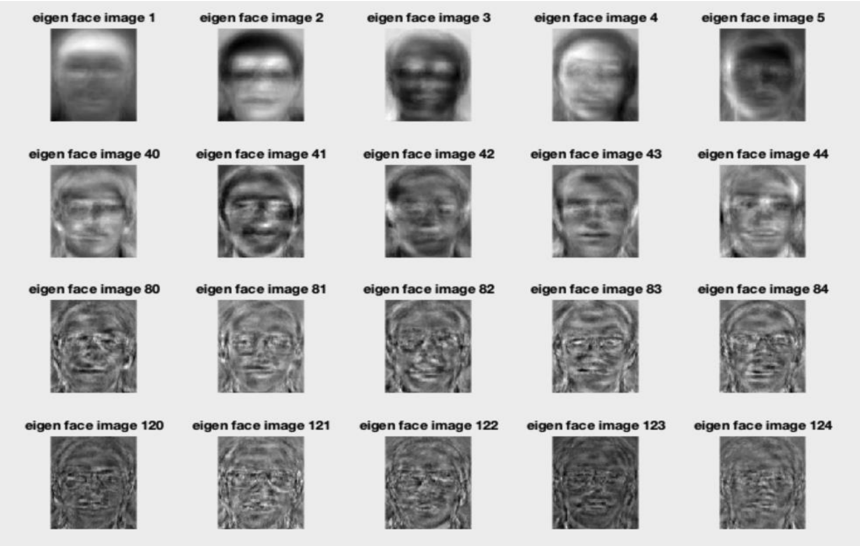

Figure 2 The Eigenfaces look like ghost image

Step 8: Sort the eigenvalues (D) in descending order, then find the corresponding eigenvectors (V). Calculate the eigenfaces (U) by multiplying the normalised images (A) and eigenvectors (V). Heuristically choose the most (more relevant) k (124) eigenfaces (U).

$$U = AV \tag{8}$$

Step 9: Multiply transpose of the $k(124)$ Eigenfaces and normalised images (A) to compute weights W

$$W = U^t A \tag{9}$$

Part 2:

Step 1: Input a new image.

Step 2: Convert inputted image to a grayscale image.

Step 3: Resize the inputted image.

Step 4: Convert $2D$ image to $1D$ column vector.

Step 5: Subtract the average column from the inputted column vector.

Step 6: Calculate the weights.

Step 7: Classify the images using Euclidean distance methods.

Step 8: Decide the threshold value by multiplying the max distance by 0.8.

Step 9: Compute the distance between inputted image and trained images for classification.

Step 9: Multiply transpose of the $k(124)$ Eigenfaces and normalised images (A) to compute weights W

$$W = U^t A \tag{9}$$

5. Experiments and Evaluations

The recognition performance was analyzed using two standard rate methods. These methods are False Acceptance Rate (FAR) and False Rejection Rate (FRR).

The distance between inputted image and the trained image in the dataset is calculated by the Euclidean Distance method. Then the calculated distances were compared by the Threshold value. If the distance is greater than the threshold, it is rejected, otherwise accepted. Then it matches with the image where the distance between them is minimum.

The distances between false images and trained images were calculated. The percentage of distances lesser than the threshold value represents the FAR. On the other hand, the distances of known images and trained images were calculated. The percentage of distances greater than the threshold value represents the FRR. 15 unknown faces were tested. 6 out of them did not match with any face image in the dataset. 9 images have been accepted even if they should not have had. So, the percentage of FAR is

%60. On the other hand, *40* known images have been tested. *1* out of them rejected though it should have been. That is, the percentage of FRR is *%2.5*. The threshold value is the same. This accuracy is not sufficient to use in real-time systems. Because even if the threshold value was very high the accuracy of unknown images tested was too low.

The FRR and FAR percentages are inversely proportional. The threshold value is affect their percentage. If the value of the threshold increased, the FAR is decreased while FRR increased as shown in Table *1*.

The experimental results on the Cambridge University, AT&T dataset using the eigenface-based on PCA for facial recognition represent that the classification accuracy on known images is *%97.5*. The dataset is small compared to other datasets used in deep learning applications. Despite this, very high result has been obtained.

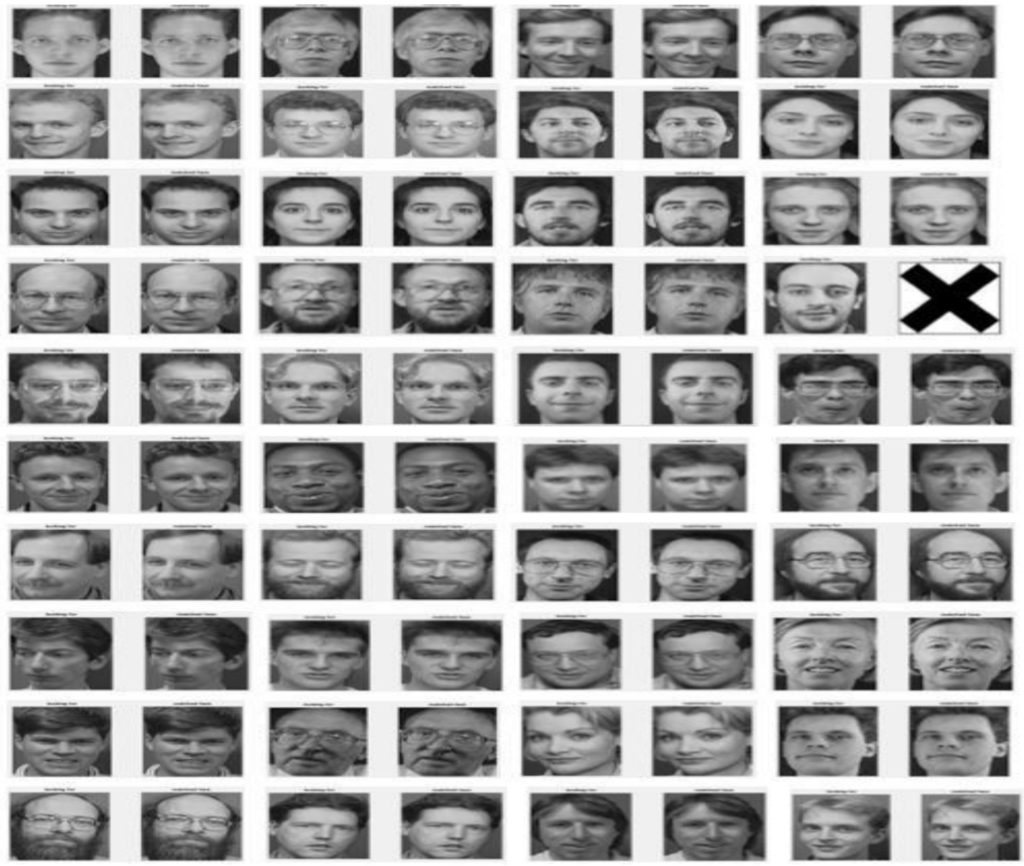

Figure 3: The table of FAR, FRR and threshold value (the numbers in the brackets represent the number of tested images)

Table 1. The table of FAR, FRR and threshold value (the numbers in the brackets represent the number of tested image)

Threshold	%FAR	%FRR	FAR(15)	FRR (40)
0.2	0	77.5	0	31
0.4	20	27.5	3	11
0.6	40	15	6	6
0.8	60	2.5	9	1

In the use of deep learning methods, we can use mini-batches to train large datasets. Otherwise, it requires high memory, powerful CPU and GPU to train large image datasets. In the use of the Eigenface method, all data is combined into a single matrix. This method will produce an extremely large matrix for large datasets. Working on a large matrix is very risky because it requires a lot of memory to train. The Eigenface method is computationally very expensive.

6. Conclusion

In this work, we have developed a project to classify a facial image dataset. We have used the eigenface method based on Principal Component Analysis (PCA) to recognise facial images. The dataset was obtained from AT &T laboratories at the University of Cambridge. PCA was used to reduce the dimension of the covariance matrix from the (10304 x 10304) to the (400 x 400). Then, 400 eigenvectors were derived from the covariance matrix used to find eigenfaces. The largest 124 out of 400 eigenfaces were used to recognise faces. The Euclidean Distance method was used to classify and find distances between tested images and trained images. In this work, 40 known and 15 unknown images were tested to obtain results.

References

[1]. "The database of faces, [accessed 7 dec. 2022]." [Online]. Available: https://cam.orl.co.uk/facedatabase.html

[2]. C. E. Bugge, J. Burkhardt, K. S. Dugstad, T. B. Enger, M. Kasprzycka, A. Kleinauskas, M. Myhre, K. Scheffler, S. Ström, and S. Vetlesen, "Biometric methods of animal identification," Course notes, Laboratory Animal Science at the Norwegian School of Veterinary Science, pp. 1–6, 2011.

[3]. S. Prabhakar, A. K. Jain, and S. Pankanti, "Learning fingerprint minutiae location and type," Pattern Recognition, vol. 36, no. 8, pp. 1847–1857, 2003.

[4]. H. Mehrotra, M. Vatsa, R. Singh, and B. M, "Does iris change over time?" PloS one, vol. 8, no. 11, p. e78333, 2013.

[5]. H. Mehrotra, M. Vatsa, R. Singh, and B. Majhi, "Does iris change over time?" PloS one, vol. 8, no. 11, p. e78333, 2013.

[6]. K. S. P. Reddy and D. K. N. Raju, "Design and implementation of an algorithm for face recognition by using principal component analysis (pca) in matlab," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 6, no. 10, pp. 115–119, 2016.

[7]. N. Abouzakhar and P. Enjamuri, "An enhanced eigenfaces-based biometric forensic model," in Procs 4th International Conference on Cybercrime Forensics Education & Training, 2010.

[8]. S. Satonkar Suhas, B. Kurhe Ajay, and B. Prakash Khanale, "Face recognition using principal component analysis and linear discriminant analysis on holistic approach in facial images database," Int Organ Sci Res, vol. 2, no. 12, pp. 15–23, 2012.

[9]. M. Karnan, "Face recognition using multiple eigenface subspaces," Journal of Engineering and Technology Research, vol. 2, no. 8, pp. 139–143, 2010.

[10]. A. Kaur and T. Sarabjit Singh, "Face recognition using pca (principal component analysis) and lda (linear discriminant analysis) techniques," International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 3, pp. 308–310, 2015.

[11]. M. Turk and A. Pentland, "Eigenfaces for recognition," Journal of cognitive neuroscience, vol. 3, no. 1, pp. 71–86, 1991.

[12]. M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and K. D. Gupta, "Evaluating the performance of eigenface, fisherface, and local binary pattern histogram-based facial recognition methods under various weather conditions," Technologies, vol. 9, no. 2, p. 31, 2021.

[13]. S. Gupta, O. Sahoo, A. Goel, and R. Gupta, "A new optimized approach to face recognition using eigenfaces," Global Journal of Computer Science and Technology, 2010.

[14]. A. Ghosh, P. K. Kundu, and G. Sarkar, "Similarity detection of illuminance images using eigenface method," Journal of The Institution of Engineers (India): Series B, pp. 1–5, 2022.

[15]. M. Slavković and D. Jevtić, "Face recognition using eigenface approach," Serbian Journal of electrical engineering, vol. 9, no. 1, pp. 121–130, 2012.

[16]. M. Imran, M. Miah, H. Rahman, A. Bhowmik, and D. Karmaker, "Face recognition using eigenfaces," International Journal of Computer Applications, vol. 118, no. 5, 2015.