Recognition of Non-symmetric Faces Using Principal Component Analysis

N. Krishnan
Centre for Information Technology & Engineering
Manonmaniam Sundaranar University,
Tirunelveli-627012, India
Krishnan17563@yahoo.com

G. Raja Jothi
CSE Dept.
Sethu Institute of Technology
Pulloor, Kariapatti
grajajothi@yahoo.com

G. Lakhs Aaron
CSE Dept.
Thiagarajar college of
Engineering, Madurai
lakhsaaron@rediffmail.com

Abstract

All the human faces are symmetric approximately up to 90% - 95% only. No face is 100% symmetric. Based on this property that 10% difference is present in any human face is given as the input to the face recognition system. The intensity variations of the faces are equalized first. Then the left and right face difference is given as the input to the database and to the face recognition system. This will reduce the storage half the size compared to any other method and provides better recognition for the non-symmetric and non-frontal faces. It also works well for the symmetric faces. Here eigenface method is used for recognition.

Keywords: *Eigenface, symmetryzation, Difference method, face recognition, non-symmetric face.*

1. Introduction

Face recognition provides us a convenient way to identify and recognize a person in a large database. With face recognition, we can recognize a person by just taking a photo of that person[1],[3]. User no longer needs to scan his fingerprint or iris for personal identification but just need to stand in front of a camera. The system can check its database to recognize the person from his image.

Face Recognition has a fundamental importance in our social relationships, being extremely valuable to our simple and daily activities. It is very useful to identifying individual in international traffic points, in crime scenes, in access control and in many other fields. Some of the areas in which face recognition used are: Credit card, driver's license, passport & personal identification, Mug shot matching, Bank/store surveillance, Crowd surveillance, Expert identification, Electronic mug shots book, and Electronic lineup, Reconstruction of face from remains and Computerized Aging.

Lighting conditions of a face is a major problem for face recognition [2]. The same person under different lighting condition may be perceived quite differently. We almost cannot recognize two people even with our eyes. So it will dramatically decrease the accuracy of a face recognition system. In the left image of Figure1.1, the dominant light source is nearly head-on; in the right of that image, the dominant light source is from above and to the right.

In this paper the Eigen face method is used for the face recognition. It will verify whether a new face image belongs to one of the individuals whose face images were stored previously in a database in a similar way, and report if there was recognition, affirming the person's identity. When images are taken in inadequate illumination conditions, the approach identifies this situation by comparing the sum of pixel values on two sides of the face. That is performed by applying the **symmetryzation technique** and, according to the case, either the dark side of the face is reconstructed from its clear side or the average of the face with its inverse one is calculated. Thus, one achieves an equalizing of face illumination.

Figure 1.1 Example face for illumination problem

So, Eigenvectors and Eigenfaces are used to represent the faces with the acceptable storage requirements and the not well illuminated faces are preprocessed using the symmetryzation property before given to the recognition process using Eigenface method. After the application of the symmetryzation techniques, there is an improvement of 72% in the approach performance.

Suppose if the face is not symmetric or not a frontal face, for example, if one side of the face has some big scar or damage then the difference between both sides are more. In such cases one cannot use this symmetryzation process directly to correct the illumination conditions, because of the loss of originality of the face.

To overcome this, in the proposed method, the original face is divided into two symmetric parts. Then the right side image (or left side) is converted to make it as the

left image (or right image). Then the difference between the original left image and the converted image is calculated. It must be different for different persons. The presence of a scar on one side of a face could be one such example. The difference gives the variation in the symmetric value. This difference image is given to the Eigen face method for further face recognition.

We present the Eigenface method for face recognition in the next section. In section 3, the symmetryzation techniques and in section 4, its limitations are discussed. Section 5, explains the new difference method. Section 6, gives some of the results. Finally section 7, gives our concluding remarks.

2. Eigenface method

Various face recognition methods are developed such as Eigenface , Fisherface, Elastic bunch graph matching, etc., and they work up to the maximum level of 90% recognition. None of the methods provide 100% recognition due to the lighting conditions, facial expressions; pose variations and orientation problems etc.

Face recognition provides us a convenient way to identify and recognize a person in a large database. The Eigenface method produces a better recognition with minimal storage requirements. Recognition performance decreases quickly because of lighting variations, the head size, or scale, is misjudged. The head size in the input image must be close to that of the Eigenfaces for the system to work well.

In that cases where every face image is classified as known, Eigenface achieved approximately 96% correct classification averaged over lighting variation, 85% correct averaged over orientation variation, and 64% correct averaged over size variation.

2.1 Eigenfaces Initialization

1. Acquire an initial set of face images (the training set)
2. Calculate the Eigenfaces from the training set, keeping only the M images that correspond to the highest eigenvalues [4], [5], [6],[7]. These M images define the face space. As new faces are experienced, the Eigenfaces can be updated or recalculated
3. Calculate the corresponding distribution in M-dimensional weight space for each known individual, by projecting their face images onto the "face space."

2.2 Eigenfaces Recognition

1. Calculate a set of weights based on the input image and the M Eigenfaces by projecting the input image onto each of the Eigenfaces.
2. Determine if the image is a face at all by checking to see if the image is sufficiently close to "face space."
3. (Optional) Update the Eigenfaces and/or weight patterns.
4. If it is a face, classify the weight pattern as either a known person or as unknown.

2.3 Eigenfaces Problems

1. Recognition performance decreases quickly as the head size, or scale, is misjudged. The head size in the input image must be close to that of the Eigenfaces for the system to work well
2. In the case where every face image is classified as known, a sample system achieved approximately 96% correct classification averaged over lighting variation, 85% correct averaged over orientation variation, and 64% correct averaged over size variation

3. Symmetrization Process

The not well-illuminated images are preprocessed. The center light, right light and left light images, all images with illumination problems, are submitted to a symmetryzation algorithm [8]. The amount of pixels values of each face side faces is computed. If the sum of one side is greater than 2/3 of the total, then that side is reflected to get the full image using the human face symmetric property. Otherwise the average of the face is calculated with its inverse one, adding the face with its inverse one and dividing by two. In the symmetryzation process, the dark or occluded side of the face is reconstructed from the clear side, in an inverse way, taking advantage of the approximate symmetry between the two sides of the faces.

Figure 3.1 shows some images before and after the symmetryzation process application; in the first three pairs of images the simple symmetryzation was applied, and in the last pair the symmetryzation of the average of the face was applied with its inverse one. Each pair shows before and after the symmetryzation process application.

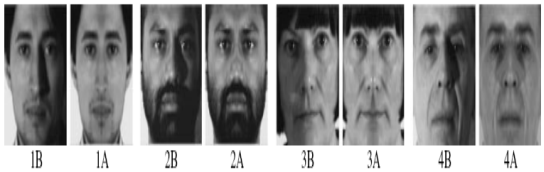

Fig. 3.1 Symmetryzation process.

4. Limitations of Symmetryzation method

The symmetryzation process works well and improve the results up to 72% as stated by the authors. But there is a big limitation as stated below.:

- Suppose if the face is not symmetric, that is for example, if the one side of the face has some big scar or damage, the difference between both sides is more.
- And even the face is symmetric but the face image is not a frontal image then symmetryzation method not works well.

Hence in such cases one cannot apply this symmetryzation process directly to correct the illumination conditions. If one applies the symmetryzation then the originality of the face will be lost. These situations are explained in the following figures.

Figure 4.1 shows non-frontal faces and faces after symmetryzation are given. The face after applying the symmetryzation gives a more different face, which is actually not a face.

Therefore, if we apply the symmetryzation process for the non-frontal faces with illumination problems then the face after symmetryzation is widely different from the original face. This face is actually given to the training database. Hence, if the training set contains the frontal faces and non-frontal faces of the person for different pose and light variations then the average face will more differ from the original face. It will reduce the recognition rate i.e., the output Euclidean distance will be more.

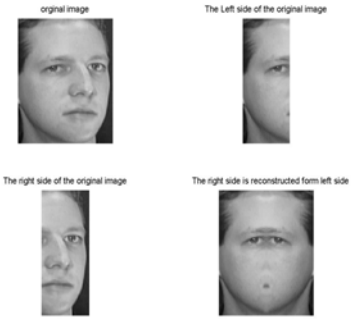

Figure 4.1 (a)

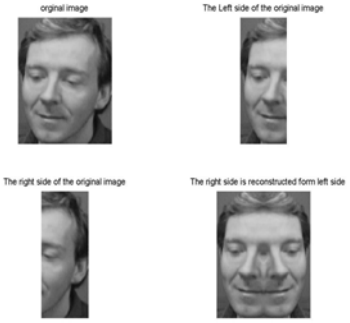

Figure 4.1 (b)

Figure 4.1 Non- frontal after symmetryzation

Figure 4.2 A non symmetric face with symmetryzation

Like the non frontal faces, the non symmetric faces also make problem in symmetryzation method as shown in the above figure 4.2. The fourth part of the image in figure 4.2 shows the face after symmetryzation which is different from the original face.

5. New difference method

The symmetryzation process works well and improves the results up to 72% as stated by the authors. But the limitations of non-frontal and non-symmetric faces reduce its efficiency.

To avoid the limitations of the symmetryzation process, in this paper a new method is proposed named difference method. By nature, all the human bodies are symmetrical about the central axis (excluding the handicapped persons). If we divide the body into 2 equivalent parts about the axis, then any one side can be get from the other side by taking the mirror image of that. This will be applicable to the human face also. But the science says, faces are not 100% symmetric. At least they are non-symmetric upto 5-10%. So all faces contain some non-symmetric parts. It may due to the hairstyle, makeup, lighting conditions, and scars or by natural (like eye difference, mouth shape etc.).

By this property of the face, the 10% non-symmetric part is extracted from the face and then it is given to the face recognition system. In the database also we store only this non-symmetric part.

Before going to extraction, the original image taken is resized to a square matrix so that it can be divided into 2 equal parts. Then the illumination conditions are checked. Using contrast stretching or the histogram equalization the image contrasts are corrected.

The above contrast-enhanced image is now divided into two parts. The left (or right) side face part is taken and the mirror image of this part was calculated by reflection that is 180° rotation from left to right (or right to left) side. This will give the non-symmetric part of the face, which is different for different persons. For example if the eye of one side is different from the other side, then the difference value gives the variation in the symmetric value.

The training set in the database also contains this difference face for different pose and lighting variations. If we give an input face then its difference- non-symmetric part is given to the face recognition system. Here we used Eigen face method for recognition.

Due to the difference image the space required to store the face is reduced to half the size that is required by the Eigen method. For example, the training set we taken for experiment contains faces for 3 different persons, 8 pose variations for each person. So it contains 24 faces and the Eigen method needs 274KB storage space. But the difference method require only 123KB storage space.

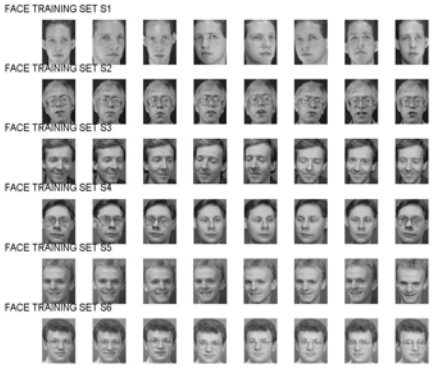

Fig 5.1 Sample faces in the database

Pseudo code for the difference method

//To Enhance the Intensity of the Input Image

```
function image=symmetricaverage(img)
K = constraststretch(img);
        row=rowsize(img)
        col=colsize(img)
        d=col/2;
if row > col
        u=imresize(img,[col,col]);
else
        u=imresize(img,[row,row]);
end
        r=imageadd(img,u);
        e=imagedivide(r,2);
        J = contraststretch(e);                (Or)
        J=histogramequalization(e);
         image=J;
```

//To Calculate Input Image

```
function image = symmetricsub(img)
        row=rowsize(img)
        col=colsize(img)
        d=col/2;
        left1=image(1:row,1:d);
        right1=image(1:row,d+1:col);
        left2=reflect(right1);
        s=imagesubtract(left1,left2);
         image=s;
```

The following figure 5.2 shows the left and right side faces and converted face of any one side so that it becomes left (or right) side face and the difference between the converted face and the opposite side face. Figure 5.3 gives the input faces used in the difference method.

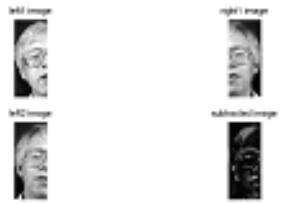

Figure 5.2 (a)

Figure 5.2 (b)

Figure 5.2 Images in difference method

6. Experimental Results

For the Testing, 3 classes for 3 persons with 8 pose variations are given to the databases. Then the 9th and 10th images belongs to each set is given to the three methods (Eigenface, Symmetric and Difference methods) separately as the input and the observations are listed in the tables 6.1, 6.2 and 6.3.

Figure 5.3 Subtracted Images used as input in the difference method

The Euclidean values of above methods are plotted as a bar graph for each method with the 6 outputs. In the x axis the input images are taken and in the y axis the Euclidean distances are taken.

Table 6.4 shows the normalized values for the 3 methods. As compared to other two methods it provide slightly lesser Euclidian value.

From the following chart in Figure 6.1, compared to the Eigenface and the Symmetric method, the new difference method gives better result by producing the minimal Euclidean distances. The threshold value for each class is different. If the Euclidean distances and the space face distances are within the acceptable threshold value of that class, then we have a better match.

Eigenface Method

Table 6.1 Results of Eigen face method

Input Face belongs to set	Nearest Class Number	Min Distance between input and faces in the database	Distance from face-space
S1(9)	1	1.9023e+006	8.5026e+009
S1(10)	1	1.7207e+006	4.7841e+009
S2(9)	2	1.0425e+006	4.0779e+009
S2(10)	2	9.0525e+005	4.6265e+009
S3(9)	3	1.5582e+006	5.8986e+009
S3(10)	3	1.8282e+006	5.4546e+009

S-indicates the set, 9 or 10 – indicates 9th or 10th face in the particular set. For example if the threshold value for class 3 is between 1.5 and 1.8 then the difference method provides more better result than the other two. It is better suitable for non-symmetric and non –frontal faces. It also works well for symmetric faces.

After Symmetrization

Table 6.2 Results of Symmetrization method

Input Face belongs to set	Nearest Class Number	Min Distance between input and faces in the database	Distance from face-space
S1(9)	1	3.1557e+006	1.8873e+010
S1(10)	1	4.6390e+006	6.9137e+009
S2(9)	2	3.0978e+006	1.1913e+010
S2(10)	2	9.0641e+005	1.2580e+010
S3(9)	3	2.0264e+006	2.1116e+010
S3(10)	3	2.5000e+006	2.0229e+010

New Difference Method with Eigenface

Table 6.3 Results of Difference method

Input Face belongs to set	Nearest Class Number	Min Distance between input and faces in the database	Distance from face-space
S1(9)	1	1.2631e+006	1.7576e+009
S1(10)	1	1.2620e+006	1.7229e+009
S2(9)	2	2.2498e+006	2.6521e+009
S2(10)	2	7.3124e+005	7.0021e+008
S3(9)	3	1.1356e+006	1.2793e+009
S3(10)	3	1.1469e+006	1.3186e+009

Table 6.4 Normalized Values

Input Image	Eigen Method	Symmetric Method	Difference Method
S1(9)	19.023	31.557	12.631
S1(10)	17.207	46.390	12.620
S2(9)	10.425	30.978	22.498
S2(10)	9.053	9.064	7.312
S3(9)	15.582	20.264	11.356
S3(10)	18.282	25.000	11.469

7. Summary and Conclusions

In this paper, the illumination problems are solved for face recognition. The approach using Eigenfaces and PCA is quite robust in the treatment of face

images with varied facial expressions and with glasses. It is also quite efficient and simple in the training and recognition stages, dispensing low level processing to verify the facial geometry or the distances between the facial organs and their dimensions. However, this approach is sensitive to images with uncontrolled illumination conditions.

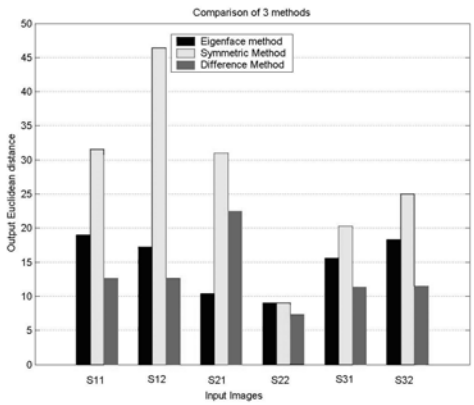

Figure 6.1 Comparison Chart

The application of the symmetryzation procedure improves significantly the Eigenfaces algorithm performance concerning images in unsuitable illumination conditions. But the symmetryzation limits the non-frontal and non-symmetric faces. The new difference method solves these problems and provides slightly better results than the above-mentioned Eigen face and symmetric methods.

Features of the difference method

- Simple to implement and use
- Use of Eigen face method reduces the storage for the image.
- It is further reduced to half the size of the storage required in the Eigen face method because of the difference image used here. So database size is less
- It also provides better improvement in face recognition by producing minimal Euclidean distances and face space distance.
- Data loss is less
- Difference method is better for non-symmetric (at the most all the faces are non-symmetric) and non-frontal faces
- It works well at illumination problems also

In the future this method will be tested with much more face sets of symmetric, non-symmetric and non-frontal faces, and along with the other face recognition methods in various facial expressions, lighting conditions and in different pose variations. That may give better results than this Eigenface method.

References

[1] William A. Barrett, "A Survey of Face Recognition Algorithms and Testing Results " , 1998 IEEE.

[2] Tang Ho Man, Sunny, "Face Recognition Algorithms Review -Term Paper December 2001".

[3] Thomas Fromherz, Peter Stucki, Martin Bichsel "A Survey of Face Recognition", MML Technical Report, No 97.01, Dept. of Computer Science, University of Zurich, Zurich, 1997.

[4] M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces", Proc. Of IEEE Conf. on Computer Vision and Pattern Recognition, pp. 586-591, June 1991.

[5] Matthew Turk and Alex Pentland,"Eigenfaces for Recognition", Vision and Modeling Group, The Media Laboratory, Massachusetts Institute of Technology, In J. of cognitive neuroscience,1991, vol.3,no.1, pages 71 to 86.

6] Belhumeur, P.N.; Hespanha, J.P.; Kriegman, D.J. "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection", Pattern Analysis and Machine Intelligence, IEEE Transactions on, Volume: 19, pp 711-720, Issue: 7 , July 1997.

[7] Jun Zhang, Yong Yan, and Martin Lades, "Face Recognition: Eigenface, Elastic Matching, and Neural Nets", Proceedings of the IEEE, VOL. 85, NO. 9, September 1997, p 1423 – 1435.

[8] P.Quintiliano, R.Guadagnin, and A.Santa-Rosa "Practical Procedures to improve face recognition based on Eigenfaces and Principal Component Analysis", Pattern recognition and Image analysis, vol.11,No.2, 2001, pp. 372 -375.