Application of Biometric Algorithms to MPEG-7

D.P. IRACLEOUS A. IONAS
Computer Science Department
University of Hertfordshire, IST College
Pireos 72, Moschato 18346, Athens
GREECE
iracleous@ieee.org

Abstract: - MPEG-7 specification introduces a new way of describing audio-visual content by the creation of descriptors which provide significant information about not only the content, but also the behavior of the multimedia information. In this paper we conducted a research about the applicability of available face recognition techniques to be used in MPEG-7 specification. We evaluated the main approaches of face recognition and we mainly focus to Eigenface approach. The approach is analyzed, in order to understand the philosophy of the technique and be able to discuss if it could be combined with the MPEG-7 specification.

Key-Words: - MPEG-7 specification, Biometric algorithms, eigenvector identification algorithms.

1 Introduction

Face recognition has been evolved rapidly during the past thirty years. As defined by many researchers [1, 5] the face recognition can be stated as follows: "*Given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces*". The face recognition process can be used in many fields such as IDs, driving licenses, passports, information security, law enforcements, surveillance and many others.

In general, biometric characteristics are unique characteristics of a human that can be measured and extracted from a biometric sample for the purpose of biometric identification. The methods based on his/her biometric characteristics such as fingerprints, iris scan, voice recognition, have a great disadvantage; they all require the cooperation of the person that is about to be recognized except from the face recognition. That is why that technique received so much attention including multiple conferences and systematic empirical evaluations such as FERET and XM2VTS protocols.

An automated Face Recognition procedure includes the capture of the image of the person that we want to recognize, the normalization of that image and the identification through our database.

Face recognition algorithms are consisted of two major parts; face detection and normalization and face identification [2]. The first part can process an image and detect the faces in it, exclude the rest of the image and keep the part with the face in the desired format. The second part takes the normalized image and performs the face recognition based on the given database. The algorithms that consist of both parts are called fully automatic algorithms and the algorithms that consist of only the second part are called partially automatic algorithms.

The ISO/IEC 19794-5:2005 standard describes the attributes of the facial images which are about to be used for automated face recognition [6]. To improve the face recognition accuracy this standard specifies not only a data format, but also scene constraints, photographic properties and digital image attributes for the basic, frontal, full frontal and token frontal Face Image Types.

The FERET [7] protocol is actually an evaluation protocol for Face Recognition Techniques. The main idea is to have an evaluation method for the results of the various algorithms in order to be able to assess the performance of each face recognition algorithm and technique. This can happen through a common image database (the FERET database) which is available to the public in order to be able to test the algorithms with common data and produce the resulting success scores. From those scores the performance of each algorithm can be evaluated and compared.

The FERET [5, 7] program was initiated to create a large database of facial images which was gathered independently from various algorithm developers. In order to maintain a level of consistency throughout the database, the same physical setup was used in each photography session. The FERET database contains a set of images for 1199 individuals along with 365 duplicate sets of a total of 14,126 images. A duplicate set is a second set of images for an individual that was taken on a different day. For

ISBN: 978-1-61804-018-3

some individuals the time that elapsed between the first and the last capture exceeded the two years. This time lapse was important because the researchers had the opportunity to study, for the first time, changes in a subject's appearance that occur over a year.

2 Problem Formulation

The database that we used is provided by "AT&T Laboratories Cambridge" which is available through Cambridge University for academic. The database contains a set of 400 images taken from 40 individuals (ten for each one). The images were taken at different times during various light conditions, facial expressions and facial details. All the images have a homogenous dark background with the individuals in upright frontal position.

From a technical perspective, all the images are in PGM format with a size of 92x112 pixels, with 256 grey levels per pixel. The images are organized in 40 directories each one having ten different images corresponding to one person.

The software for which the algorithm was written is the Mathworks Matlab which saves fragments of code in .m files which can be executed from the main command line. In this particular "program" each .m file is a function and the first function that we have to call is the "load_database". The function opens the "faces" directory which contains 40 sub-directories (S1 to S40) each one having the ten images for each person. Now that we have all the data from the images stored in a matrix we run the "face_recognition" function which contains the main algorithm for the face recognition process. First we generate a random number from 1 to 400 to pick up a random image from the w matrix. We store that image to a new matrix characterized from the variable r and we store the rest of the images into a matrix v. So we have a single column matrix containing the selected image and we have a 399 columns matrix containing the rest of the 399 images.

After the separation of the selected image we calculate the mean value for the rest of the 399 images and we subtract it from the v matrix which contains them. The above calculation makes the images to distinguishing more from each other. Then we calculate the eigenvectors of the correlated matrix and we pick N samples corresponding to the largest Eigen values, where N is the number of signatures that we want to compare. In order to make the algorithm efficient we need to find a balance for the number of signatures as while this number raises, the face recognition process is more accurate but it needs more time to perform the recognition process.

Finally, we calculate the signatures for each image and we compare it with the signature of the selected image. The closest sample is probably the person from the database that we are looking for.

2.1 The Eigenface Approach

Eigenface method of face recognition is one of the most popular approaches in the specific field [13] [14]. [14] [15] This method has become one of the most well-known methods of face recognition due to it relatively simple nature, strong mathematical foundation and reasonably successful results. The method attempts to reduce a facial image, to the most variant features, such that recognition can be performed in a substantially reduced image space than that of the direct correlation method. However, rather that detecting and measuring a set of specific features on a facial image, the method maintains a holistic representation, discarding any need for feature detection other than that required for the initial image alignment. It requires the application of Principal Component Analysis (PCA) [11], also known as the Karhunen- Loeve transform.

Supposing we have a training set of images of 92 by 112 pixels. These images could be represented as two-dimensional arrays of pixel intensity data. Similarly, vectors of 10304 (92x112) dimensions could represent the images. Interpreting these vectors as describing a point within 10304 dimensional space means that every possible image occupies a single point within this image space. In theory similar images should occupy points within a fairly localized region of this image space. We wish to extract the region of image space that contains faces, reduce the dimensionality to a practical value, and maximize the spread of different faces within the image subspace.

Suppose we have a set of face images $\Gamma_1, \Gamma_2, \Gamma_3, \ldots \Gamma_M$. The average face of the set is calculated by

$$\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n. \tag{1}$$

Each face differs from the average by the vector $\Phi_i=\Gamma_i-\Psi$. An example of an average face Ψ is shown in Fig. 1.

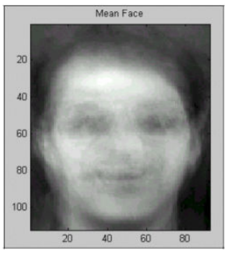

Fig. 1 – Average Face Ψ (Taken from [16])

This training set of very large vectors is then subject to Principal Component Analysis, which seeks a set of M orthonormal vector u_n which best describes the distribution of the data. The k^{th} vector, u_k is chosen such that:

$$\lambda_\kappa = \frac{1}{M}\sum_{n=1}^{M}(u_\kappa^T \Phi_n)^2 \quad (2)$$

is a maximum, subject to:

$$u_l^T u_k = \delta_{lk} = \begin{cases} 1, & if\ l = k \\ 0, & if\ l \neq k \end{cases} \quad (3)$$

The vectors u_k and scalars λ_k are the eigenvectors and eigenvalues, respectively, of the covariance matrix:

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T \ = AA^T \quad (4)$$

where the matrix $A = [\Phi_1\ \Phi_2 \ldots\ \Phi_M]$.

The matrix C, however, is N^2 by N^2, and determining the N^2 eigenvectors and eigenvalues is an intractable task for typical image sizes. We need a computationally feasible method to find these eigenvectors.

If the number of data points in the image space is less than the dimension of the space $(M<N^2)$, there will be only $M - 1$, rather than N^2, meaningful eigenvectors. The remaining eigenvectors will have associated eigenvalues of zero. Fortunately we can solve for the N^2 – *dimensional eigenvectors* in this case by first solving for the eigenvectors of an M by M matrix – in practice we solve for example a 16 × 16 matrix rather than a 16,384 × 16,384 matrix – and then taking appropriate linear combinations of the face images Φ_i. Consider the eigenvectors v_i of A^TA such that:

$$A^TAv_i = \mu_i v_i \quad (5)$$

If we multiply both sides by A, we have:

$$AA^TAv_i = \mu_i Av_i \quad (6)$$

from which we see that Av_i are the eigenvectors of $C = AA^T$.

Following the analysis, we construct the M by M matrix $L=A^TA$, where $L_{mn} = \Phi_m^T\Phi_n$, and find the M *eigenvectors,* v_i of L. These vectors determine linear combinations of the M training set face images to form the eigenfaces u_l,

$$u_l = \sum_{k=1}^{M} v_{lk}\Phi_\kappa, \qquad l = 1, \ldots, M. \quad (7)$$

With this analysis the calculations are significantly reduced, from the order of the number of pixels in the images to the order of the number of images in the training set. In practice the training set of face images will be relatively small and the calculations become quite manageable. The associated eigenvalues allow us to rank the eigenvectors according to their usefulness in characterizing the variation among the images.

2.2 MPEG-7 Technical Description

MPEG-7 Multimedia Description Schemes (DSs) are metadata structures for describing and annotating Audio-Visual (from now on AV) content. The DSs provide the ability to describe in XML the important concepts related to AV content description and content management in order to simplify searching, indexing, filtering and access. The DSs are defined using the MPEG-7 Description Definition Language (DDL), which is based on the eXtensible Markup Language, and are initialized as documents or streams. The resulting descriptions can be expressed in a textual form like human readable XML, or in a compressed binary form for storage and transmission. [8]

The main target of the MPEG-7 is to allow interoperable manipulation of AV content, by enabling interoperability among the devices that deal with AV content description. MPEG-7 describes specific features of AV content information and content management which can take either a textual XML form suitable for editing, searching and filtering or a binary form suitable for storage, transmission and storage delivery.

3 Evaluation

In this part we are going to describe the metric that we have used to measure the efficiency of the algorithm. The main idea is to count the success rate of the face recognition process and measure the efficiency of the algorithm in distorted images. First we start with non-distorted images and then we begin to distort the images gradually to discover how the algorithm behaves. Then we export the results where we keep a record of the total face recognition cycles completed along with the time that elapsed to complete the process.

To define the metric we need to create a mathematical function to recognize whether the recognition process is successful or not and then run the tests automatically. Finally, when we have a set of training images and we pick and index x_1, it is removed from the range of images so if an image has an index 256 and we pick an image with smaller

ISBN: 978-1-61804-018-3

index for recognition, then the previous image's index is reduced by one.

As the metric is defined and the appropriate functions are written we can proceed with the evaluation of the algorithm. The main idea is to apply filters on the images and check the efficiency and the behavior of the algorithm. We know that we have 400 images which represent 40 individuals. If we pick an image, the resulting correct images can be 9. We consider that a sufficient number of cycles for each level of distortion should be 1000 times.

To proceed with the distortions we will use functions of Matlab from the Image Processing Toolbox.

Value	Description
'gaussian'	Gaussian white noise with constant mean and variance.
'poisson'	Generates Poisson noise from data instead of adding artificial noise to the data.
'salt & pepper'	On and Off Pixels.
'speckle'	Multiplicative noise.
'average'	Averaging filter
'disk'	Circular averaging filter (pillbox)
'gaussian'	Gaussian lowpass filter
'laplacian'	Approximates the 2-D Laplacian operator
'log'	Laplacian of Gaussian Filter
'motion'	Approximates the linear motion of a camera
'prewitt'	Prewitt horizontal edge-emphasizing filter
'sobel'	Sobel horizontal edge-emphasizing filter
'unsharp'	Unsharp contrast enhancement filter

Table 1 – imnoise() and fspecial() noise types

All the above distortions can be applied with various parameters which are different for each type. In order to proceed we will separate those distortions in two main categories; the distortions that can be applied multiple times exponentially (noises) and the distortions that can be applied once (filters). We need to perform this action in order to be able to compare the results.

In Fig. 2 we can see the graph generated from the above results.

If we observe the Fig. 2 we can conclude that the algorithm success rate is reduced significantly until the distortion level reach 9% and then until the 50% distortion the success rate is reduced in very small segments.

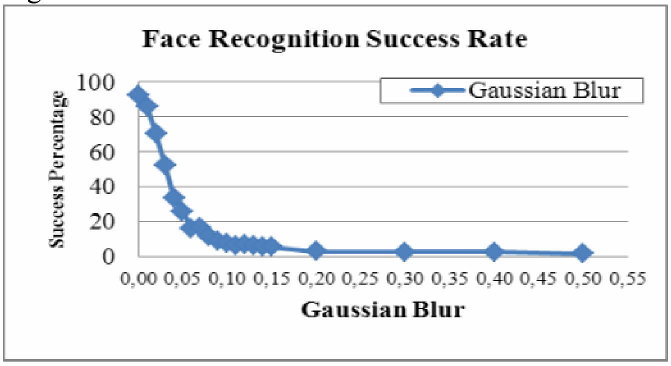

Fig. 2 – Gaussian Blur Success Rate.

If we gather the results from the three distortions, the following graph is produced (Fig. 3):

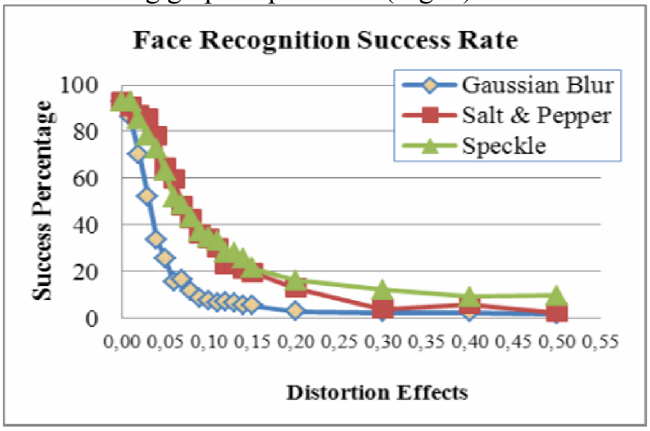

Fig. 3 – Exponential Distortions Graph.

Speaking in general terms we can say that the algorithm loses its efficiency when a small level of distortion is applied. If we compare the distortion filters we could say that the Gaussian Blur affects the face recognition success rate in lower percentages as the other two have almost identical results.

4.3.2 Filter Based Distortions

Distortions were applied once with their default levels of application. The result from the application of the distortion filters is shown in Fig. 4.

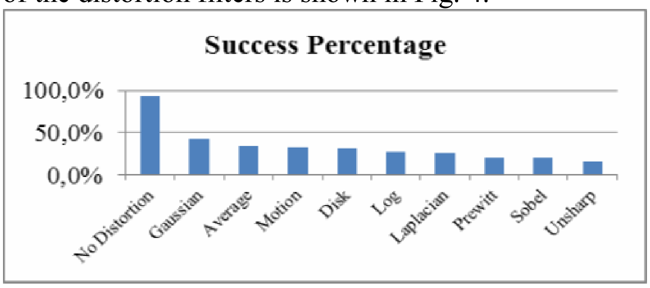

Fig. 4 – Filter Based Distortions Results

ISBN: 978-1-61804-018-3

Reviewing the results from both noise and filter distortion techniques we can observe that the algorithm is losing its efficiency in a significant level by the application of any of them. In some examples the application of some noise by 1% affects the face recognition results by 6% and usually after the distortion level of 30% -40% the algorithm actually cannot perform successful face recognition and returns the same result again and again. To prevent that in real life examples image clarifying algorithms are used, which normalize the input image to produce more reliable results.

4 Proposed MPEG-7 descriptors

We could create a Description Scheme (DSs) which would have Face Position descriptors along with Coding Schemes for the implementation of the calculation of the average image. The main idea is to have a large image containing all the images of the training set, the face position descriptor would be responsible to give the coordinates of each face and a Coding Scheme would be responsible for the calculation of the average image Ψ. In that way we could reduce the time needed for the face recognition procedure.

The above theories could be implemented in the MPEG-7 Description Definition Language (DDL) using XML schema as basis. That XML could have the appropriate nodes to define the desired descriptors. A sample of the proposed XML can be bound in Fig. 5.

```
<Mpeg7 type="complete" xmlns="http://www.mpeg7.org/2001/MPEG-7_Schema"
    xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
    xsi:schemaLocation="http://www.mpeg7.org/2001/MPEG-7_Schema">
    <ContentDescription xsi:type="ContentEntityType">
        <MultimediaContent xsi:type="ImageType">
            <Image>
                <ImageCoordinate type="face">10 35 25 62</ImageCoordinate>
            </Image>
        </MultimediaContent>
    </ContentDescription>
</Mpeg7>
```

Fig. 5 – MPEG-7 Theoretical Face Location.

5 Conclusion

In this paper we studied face recognition algorithms based in eigen-analysis and we performed detailed evaluation which we explained how they work and how MPEG-7 format can be customized to use them.

MPEG-7 actually introduces a new era of multimedia information which will enhance the audio-visual experience.

References

[1] Deluisgarcia, R., et al., Biometric identification systems. *Signal Processing*, 2003. 83(12): p. 2539-2557.

[2] A.S. Tolba, A.H.E.-B., and A.A. El-Harby, Face recognition: A literature review. *International Journal of Information and Communication Engineering*, 2006. 2(2).

[3] Riccio, D. and J. Dugelay, Geometric invariants for 2D/3D face recognition. *Pattern Recognition Letters*, 2007. 28(14): p. 1907-1914.

[4] Shih-Fu Chang, T.S., Atul Puri, Overview Of Mpeg-7 Standard. *IEEE Transactions on circuits and systems for video technology*, 2001. 11(6).

[5] W. Zhao, R.C., A. Rosenfeld, P.J. Phillips, Face Recognition: Literature Survey. *ACM Computing Surveys*, 2000. 35(4): p. 399-458.

[7] S. Rizvi, P.J.P.a.H.M., The FERET Verification Testing Protocol for Face Recognition Algorithms. *In Proc. of Inter. Conf. on Autom. Face and Gesture Recognition*, 1998: p. 48-53.

[8] Moving People Experts Group, *MPEG-7 Overview Official Report*. 2002.

[9] Jie Yang, H.Y., William kunz, An Efficient LDA Algorithm for Face Recognition.

[11] Draper, B., Recognizing faces with PCA and ICA. *Computer Vision and Image Understanding*, 2003. 91(1-2): p. 115-137.

[12] Poggio, R.B.a.T., Face Recognition: Features versus templates. *IEEE Transactions on pattern analysis and machine intelligence*, 1993. 15(10): p. 1042-1052.

[13] Pentland, M.T.a.A., Eigenfaces for Face Recognition. *Journal of Cognitive Neuroscience*, 1991. 3(1).

[14] Hyun-Chul Kim, D.K., Sung Yang Bang, Face recognition using the mixture-of-eigenfaces method. *Elsevier*, 2002: p. 1549-1558.

[15] Kim, H.-C., et al., Face recognition using the second-order mixture-of-eigenfaces method. *Pattern Recognition*, 2004. 37(2): p. 337-349.

[16] Amit Kumar, P.S., Shishir Kumar, Face Recognition using Neural Network and Eigenvalues with Distinct Block Processing. *Department of CSE, Jaypee University of Engineering & Technology*, Guma (MP) India.

[17] Peter N. Belhumeur, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on pattern analysis and machine intelligence*, 1997. 19(7): p. 711-720.