

# FACE RECOGNITION UNDER LOSSY COMPRESSION

*Mustafa Ersel Kamaşak and Bülent Sankur*

Signal and Image Processing Laboratory (BUSIM)  
Department of Electrical and Electronics Engineering  
Boğaziçi University  
Bebek 80815 Istanbul, Turkey  
{kamasak, sankur}@busim.ee.boun.edu.tr

## ABSTRACT

We investigate the extent to which a large image archive can be compressed without critically detracting its recognition performance. We tested the performance of eigenface-based methods with transform, subband and vector quantization based coders. The compression-recognition trade-off is measured in terms of correct identification percentages as well as the corruption of the face space.

## 1. INTRODUCTION

Face recognition is an emerging research area. It has many applications, such as restriction of access, man-machine interface, nonintrusive identification for ATM transactions etc. It is a challenging pattern recognition problem. The main object is to develop a robust and automatic face recognition system, which will identify face images taken with varying illumination conditions, facial expressions and facial accessories. The face images must be identified from a face database consisting of a few hundred individuals to huge ones containing millions of shots.

In the literature many recognition techniques have been proposed, which are tested with different face databases. The most popular face recognition techniques are correlation methods, dimension reduction techniques with parametric discrimination like “Eigenfaces” [4] or “Fisherfaces” [6], and “Hidden Markov Model” based recognition techniques. These techniques have reached over 90% success ratio in large face databases with various preprocessing effects, involving scale, orientation and illumination normalizations of the face image.

As the face database gets larger, they have to be maintained in a compressed platform to overcome storage and bandwidth limitations. Therefore identification of faces in the compressed domain is becoming a relevant problem. The compression techniques will cause loss of information. In this paper, we try to investigate, how information loss caused by compression affects the face recognition per-

formance. We also try to determine to what extent a face database can be compressed and still be with a recognition scheme.

## 2. FACE RECOGNITION TECHNIQUES

### 2.1. Correlation

Correlation is the simplest algorithm for the face recognition task. It is based on the matching between the test image and the face images in the database. However, it has many drawbacks. It is computationally expensive, and requires the storage of all face images in their original form. Also, this method is sensitive to scale, illumination conditions, rotation, facial expressions, accessories, and the resolution of face images in the database.

Brunelli and Poggio described a correlation based method from frontal face images [5]. It is based on the matching of templates corresponding to facial features. To reduce computational complexity they first detect the location of features by using multiscale feature templates. Nevertheless, this technique is still computationally expensive.

The recognition ratio reported for the mentioned method [5] is 96%. Due to its sensitivity against scale and location of faces in the scenes, it needs robust and extensive preprocessing.

### 2.2. Eigenfaces

The “Eigenface” method [4] is based on the Karhunen-Loeve Transformation of face images. It maximizes the scatter of the linearly projected face classes in the so-called “Face Space”. Recall that such a projection is optimal for the reconstruction of any image from a lower dimensional space, but it may not be optimal for discrimination among classes.

In this method, the faces are mapped to a new feature space of dimension  $m$ , from an  $n$  dimensional space, where  $m \ll n$ , that is

$$x_k \implies y_k \quad x_k \in \mathcal{R}^n \quad \text{and} \quad y_k \in \mathcal{R}^m$$

In this expression,  $x_k$  denotes the “face image” as a vector consisting of lexicographically ordered  $n$  pixels, while  $y_k$  is the vector of the  $m$  eigencoefficients of the same image.

New feature vectors  $y_k$  can be found by linear transformation:

$$y_k = W^T x_k \quad (1)$$

where  $W \in \mathcal{R}^{n \times m}$  is a matrix with orthonormal columns.

If the scatter matrix  $S_T$  is defined as

$$S_T = \sum_{k=1}^N (x_k - \mu)(x_k - \mu)^T \quad (2)$$

where  $N$  is the number of images in the training set and  $\mu \in \mathcal{R}^n$  is the mean of the images in the training set, then one can choose  $W_{opt}$  such that

$$W_{opt} = arg \max_W |W^T S_T W| \quad (3)$$

Columns of  $W_{opt}$  correspond to the eigenvectors of the scatter matrix, which have the  $m$  largest eigenvalues.

Then all faces in the database are projected to this new space and their feature vectors are calculated. A test image is identified by using the Euclidian distance between the feature vectors of the test face and the faces in the database. The new face is identified as  $k^{th}$  person if

$$k = arg \min_k \|Y_{test} - Y_k\|$$

This maximized scatter contains not only interclass variations, but also intraclass variations, which is unwanted information for classification problems. The variations within a class result from illumination conditions, facial expressions and accessories. Because of the intraclass variations, classes are not well clustered in the new feature space.

Discarding of the most significant three principal components is suggested in [6]. These components are believed to encode the illumination variations and backgrounds of the scenes, so discarding them may improve the performance of this technique.

Eigenface technique has been tested on the relatively large FERET database [9]. The reported correct classification rates are 96% over lighting conditions, 85% over orientation variations, and 64% over scale variations.

### 2.3. Fisherfaces

Fisher’s Linear Discriminant Analysis (LDA) is a class specific method. This method chooses a transform matrix,  $W$ , that maximizes the ratio of interclass scatter matrix  $S_B$  to within class matrix  $S_W$ , where

$$S_B = \sum_{i=1}^c N_i (\mu_i - \mu)(\mu_i - \mu)^T$$

$$S_W = \sum_{i=1}^c \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

$c$ : number of classes

$\mu_i$ : mean image of class  $X_i$

$N_i$ : number of samples in class  $X_i$

$$W_{opt} = arg \min_W \frac{|W^T S_B W|}{|W^T S_W W|} \quad (4)$$

$$= [w_1 w_2 \dots w_m]$$

where  $w_i$  is a set of nonzero generalized eigenvectors of  $S_B$  and  $S_W$  corresponding to  $m$  largest eigenvalues.

$$S_B w_i = \lambda_i S_W w_i \quad i = 1, 2, \dots, m$$

There are at most  $c - 1$  nonzero eigenvalues, and so an upper bound of  $m$  is  $c - 1$ .

$W_{opt}$  cannot be found directly, since  $S_W$  is singular. The rank of  $S_W$  is less than  $N - c$ . To overcome this problem “Fisherface Method” is proposed [6]. In this method the image is reduced to  $N - c$  dimensional space using principal components, so that  $S_W$  is nonsingular. Then using the “Fisher Linear Discrimination” described above the dimension can be reduced to  $c - 1$  or lower dimensional space.

$$W_{opt} = W_{pca} W_{fld} \quad (5)$$

where

$$W_{pca} = arg \max_W |W^T S_T W|$$

$$W_{fld} = arg \max_W \frac{|W^T W_{pca}^T S_B W_{pca} W|}{|W^T W_{pca}^T S_W W_{pca} W|}$$

The correct classification performance of this technique is 99.4% under variations in lighting, facial expressions and accessories. The performance of “Eigenface Method” on the same database is only 80%.

### 2.4. Hidden Markov Model

Hidden Markov Models are used extensively for speech processing, where data is one dimensional. To overcome the high computational cost of fully connected two dimensional hidden markov model, multi-model representations are proposed and used for character recognition.

One dimensional hidden markov model for face identification is also proposed [10]. In this scheme, each face is assumed to be in an upright, frontal position. Images can be converted into one dimensional temporal sequences or one dimensional spatial sequences. Spatial sequences were preferred, where features will occur in a predictable order. This ordering suggests the use of a top-bottom model, where only transitions between adjacent states in a top to bottom manner are allowed.

In this method, faces in the training set are sampled using a sliding window, from top to down with some overlap. These samples are fed to left-to-right HMM. For each face, HMM parameters are calculated.

$$\lambda^{(i)} = (A, B, \pi) \quad 1 \leq i \leq N \quad (6)$$

where  $N$  is the number of faces in the training set,  $A$  is the transition probability matrix,  $B$  is the observation probability matrix, and  $\pi$  is the initial state distributions.

Matrix  $A$  measures the probability of going from one facial band to another. After training,  $A^k$  will contain the transition probabilities from one facial band to another across the face and the thickness of the various bands. Matrix  $B$  measures the probability of observing a feature vector, given that we are looking at a particular facial band. After training  $B^k$  will contain feature vector distributions for each face in the database.  $\pi$  will not provide any discriminating information, because all observations start from top.

To identify a test face, it is sampled using the same window used in training and converted to one dimensional observation sequence,  $O$ . It is identified as  $k^{th}$  person if

$$k = \arg \max_i P(O|\lambda^{(i)}) \quad 1 \leq i \leq N \quad (7)$$

There are three HMM parameters which affect the performance of the model; number of HMM states, the height of sampling window, and amount of overlap during sampling. Number of HMM states determines the number of features used to characterize the face. The height of the sampling window determines the size of the features that the model extracts and the overlap determines how likely feature alignment is and it is expected that a large overlap would increase the likelihood of preserving the alignment.

It has been seen that large overlap in the sampling results in better recognition performances, and as the overlap becomes noticeable the effect of window height is limited. Also best results are obtained with four or more states [10]. The performance of this technique is reported as 84% in a small face database, consisting of 24 individuals. It is also stated that ‘‘Eigenface’’ method performs 73% correct recognition on the same database [10].

## 2.5. Matching Pursuit Filters

The original matching pursuit idea of Mallat and Zhang [2] uses a greedy heuristic to iteratively construct a best-adapted decomposition of a function  $f$  on  $\mathcal{R}$ . The algorithm works by choosing at each iteration  $i$  the wavelet  $g$  in the dictionary  $\mathcal{D}$  that has maximal projections onto residue of  $f$ . The best-adapted decomposition is selected by the following greedy strategy. Let  $R^0 f = f$ ; then  $g_i$  is chosen such that

$$|\langle R^i f, g_i \rangle| = \max_{g \in \mathcal{D}} |\langle R^i f, g \rangle| \quad (8)$$

where

$$R^{i+1} f = R^i f - \langle R^i f, g_i \rangle g_i$$

for  $i \geq 1$

Each wavelet in the expansion is selected by maximizing the right hand term in (8). This equation allows for an expansion based on a single function, and minimizes the reconstruction error.

Matching pursuit filters can also be applied to pattern recognition [1]. To extend the technique from one dimensional function decomposition to two dimensional pattern recognition

- A dictionary of two dimensional wavelets are used.
- Instead of minimizing reconstruction error, a problem specific cost function,  $C_g$  is minimized in (6). Filters are designed to encode the similarities of the faces in the training set. Therefore  $C_g$  is chosen such that coefficients of the features cluster. On the other hand, filters for face recognition are designed to encode the differences among the faces in training set. The cost function for face recognition is chosen such that it selects the coefficients that discriminate the features of the individuals.

This algorithm does not optimize a global cost function for the selection of wavelet functions, but uses a greedy algorithm, which chooses the next wavelet in the expansion by minimizing a cost function. The designed filters will capture both local and global features, which contrasts with linear parametric discrimination methods, where face images are treated as vectors, and the representation is global.

Performance of the ‘‘Matching Pursuit Filters’’ using five facial features is 95.4%, where the database consists of 311 individuals [1].

## 3. COMPRESSION SCHEMES AND CODING ARTIFACTS

The following three methodologies seem to form a fairly representative set of compression techniques:

- Vector Quantization : This scheme is especially interesting because VQ can be made to incorporate certain low-level image processing tasks and the coding task can be coupled with classification performance [8].
- JPEG is included since at present it seems to be the industry standard, despite an increasing number of close competitors [8].

- The subband based schemes seem to be the closest competitor to JPEG. We have chosen the SPIHT algorithm which exploits the embedded zero three coding concept [7, 8].

All these schemes incur into certain artifacts at low bit rates, such as blocking, blurring, edge busyness or staircase effect, loss of detail, shading gradations, ringing etc. For example block-based schemes like VQ and JPEG incur into unpleasant blocking artifacts and staircasing of edges. Subband coding schemes on the other hand cause ringing artifacts and blurring.

#### 4. EXPERIMENTAL RESULTS

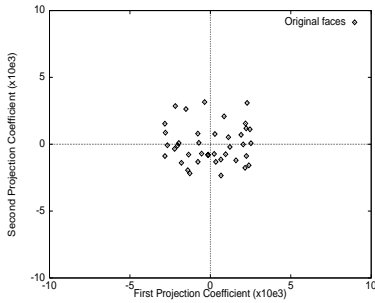


Figure 1: Scatter of face classes on the two principal components without compression

The lossy compression schemes cause the loss of discriminating information of face images. It can be seen from the comparison of Fig 1 and Fig 2, that compressed face images cause the corresponding feature space to shrink. This shrinking means that discrimination and recognition performance is degraded.

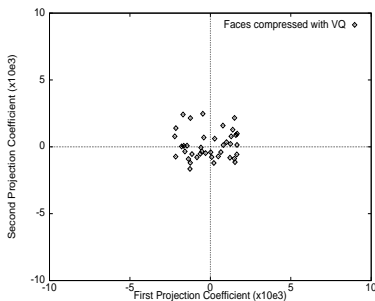


Figure 2: Scatter of face classes on the two principal components after compression with VQ

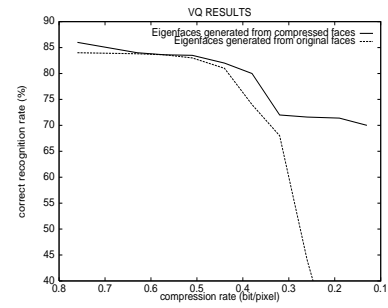


Figure 3: Recognition performance with faces compressed using VQ

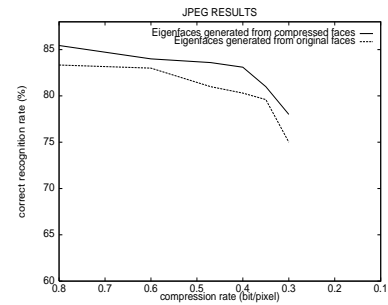


Figure 4: Recognition performance with faces compressed using JPEG

The main result from these experiments is that the face images can be compressed using any of the stated compression schemes down to 0.4 bit/pixel without a significant information loss. Fig 3 and Fig 4 show, respectively, the correct recognition percentage as a function of compression rate, using the vector quantization and JPEG algorithms. In Fig 5, the performance of the SPIHT algorithm is shown, where one can compress the images down to 0.2 bit/pixel before reaching the breakdown point. In this sense SPIHT algorithm outperforms JPEG and VQ. In these plots one can also observe that, there is a difference of few percentage points between operating with eigenfaces obtained from the original face images and with eigenfaces from their compressed versions.

Using “distance from face space” metric [4] - which is a measure of faceness of images - we explore how these compression schemes preserve the discriminating features of the compressed faces. From Fig 6, SPIHT was again the best algorithm that preserves the features of faces.

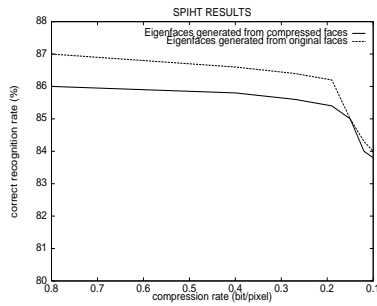


Figure 5: Recognition performance with faces compressed using SPIHT

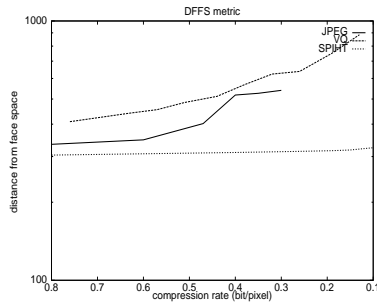


Figure 6: Faceness of the images with compression measured using dffs metric

## 5. CONCLUSION

Large face archives need to be maintained in compressed form. We have determined that face images can be compressed down to 0.2 bit/pixel (a ratio of 40:1) rates without any major deterioration in recognition performance using "Eigenface" technique. The subband coding algorithm (SPIHT) proves to be the most robust scheme among the three compression methods investigated.

In this study, we have witnessed that multiresolution type compression schemes caused the least deterioration to the recognition performance of eigenface-based methods. Therefore we conjecture that, to couple a multiresolution compression scheme with a multiresolution recognition technique will prove beneficial. The specific recognition algorithm we intend to explore is the "Matching Pursuit Filters" applied to face identification problem [1]. This is a nonlinear and nonparametric technique, which is based on adaptive wavelet expansion.

## 6. REFERENCES

- [1] Phillips J., "Matching Pursuit Filters Applied to Face Identification", IEEE Transactions on Image Processing, Vol 7, No 8, Aug 1998, pp 1150-1164.
- [2] Mallat G., Zhang Z., "Matching Pursuit Filters with Time-Frequency Dictionaries", IEEE Transactions on Image Processing, Vol 41, No 12, Dec 1993, pp 3397-3415.
- [3] Freeman W., Adelson E., "The Design and Use of Steerable Filters", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 13, No 9, Sep 1991, pp 891-906.
- [4] Turk M., Pentland A., "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, Vol 3, pp 71-86, No 1, 1991.
- [5] Brunelli R., Poggio T., "Face Recognition: Features versus Templates", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp 1042-1052, Vol 15, October 1993.
- [6] Peter N., Hespanha J., Kriegman D., "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 19, No 7, July 1997.
- [7] Said A., Pearlman W. A., "A New and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", IEEE Transactions on Circuits and Systems for Video Technology, Vol 6, pp 243-250, June 1996.
- [8] Rabbani M. J., *Digital Image Compression Techniques*, SPIE Optical Engineering Press, 1991.
- [9] Pentland A., Starner T., Eteoff N., Masoiu A., Oliyide O. and Turk M., "Experiments with Eigenfaces", Looking at People Workshop IJCAI'93, Chamberry, France, August 1993.
- [10] Samaria F., Fallside F., "Face segmentation for Identification and Feature Extraction using Hidden Markov Models", Image Processing: Theory and Applications, 1993.