



## A Review of Face Recognition Techniques for In-Camera Applications

Title	A Review of Face Recognition Techniques for In-Camera Applications
Author(s)	Iancu, Claudia; Costache, Gabriel; Corcoran, Peter M.
Publication Date	2007-08-20
Publisher	IEEE

# A Review of Face Recognition Techniques for In-Camera Applications

Claudia Iancu, Peter Corcoran and Gabriel Costache  
Dept. of Electronic Engineering, NUI Galway, Ireland.

**Abstract**— This paper presents an overview of a number of classical face recognition techniques, including an assessment of their suitability for applications in real-time embedded systems such as digital cameras, and a generalized discussion of the training requirements for different face recognition techniques. Also, an introduction to face recognition approaches based on hidden Markov Models (HMM) is made, and their advantages and benefits are shown, particularly their ability to provide acceptable levels of accurate recognition for unseen collections of facial images. Finally, a detailed description/implementation of a working HMM test bed is done, together with a presentation of some preliminary results.

## I. INTRODUCTION

Face recognition remains one of the most important areas in computer vision with a successful history, and still captures the attention of many researchers from both academic and industrial environments. One of the main reasons for this interest is the large area of commercial applications where face recognition systems could be used starting with small, basic login applications and finishing with high security access control, secure biometric based transactions or security surveillance which requires a very high accuracy.

Digital cameras continue to gain territory in the fight with film photo cameras and are one of the CE industry's recent success stories. However as users switch from conventional to digital photography they find themselves with rapidly growing collections of digital images. Few consumers have the time and personal discipline to manually catalog and organize these growing personal image collections.

We showed in [1] how we can help users to automatic sort their digital collection of images using the people in the images as patterns. For this we have to detect and classify the individuals, which involve algorithms for face detection and recognition.

For PC applications there are many alternatives for face detection and recognition, but for embedded applications these problems became more challenging due to the lack of resources. We also have to consider the fact that we are working with consumer images with large variations that affect the accuracy of classical face recognition algorithms.

One conclusion from our previous work is that by detecting and analyzing the images as they are captured in the digital camera, the user will be provided with preliminary results directly from the camera, plus the time needed to train big collection of images on the PC will decrease.

Taking advantage of recent advances in embedded face detection [2] we propose to analyze the use of embedded face recognition algorithms in camera applications.

Special issues have to be addressed for the algorithms in order to make possible their implementation in hardware. Most of these issues regard the complexity of the algorithms, their speed and their special requirements regarding needed memory.

## II. FACE REGION PREPROCESSING ALGORITHMS

### A. Face Detection

The problem of detecting faces in image is well known [15, 16] and will not be discussed in detail in this paper. We have evaluated a number of automatic face detection techniques and the best success rates, in our experience, are obtained from a variant of the algorithm described in [16] but with significant customizations to utilize the hardware image processing capabilities of a digital camera.

It is worthwhile remarking that 90%+ of the time initially required by our algorithm was spent in the face detection phase. Fortunately the recent availability of highly optimized in-camera face detection systems has solved this bottleneck. Thus practical in-camera face recognition systems are likely to become a reality in the next 18-24 months.

### B. Face Region Normalization

After a face region is detected it is generally necessary to try and align and/or resize the region so that it can be subsequently analyzed by standard face recognition tools. The system described in this paper has been restricted to 2-D normalization techniques. This is mainly to reduce the computational requirements of the normalization algorithms with a view to an eventual in-camera implementation. Note that we have achieved useful preliminary results on 3D normalization using active appearance model (AAM) techniques [17] but the effectiveness and accuracy of these is still somewhat dependent on the prototype models employed.

## III. FACE RECOGNITION ALGORITHMS

The goal of our system is the detection and recognition of faces in consumer images. This is a non-trivial extension of the basic face detection/recognition problem which has been well researched over the past 25 years or so. In particular, consumer images have a wider variation of pose, illumination and orientation than is typical in the data sets employed by most researchers [1]. Much of our approach has been focused on finding methods of combining the outputs of several face

recognition techniques to generate a more accurate recognition score than is possible by using a single recognition technique [17]. Thus in this section we review some of the known techniques for face recognition, discussing the relative advantages and disadvantages of each technique.

There are many algorithms for face recognition reported in the literature. Initial approaches [9] were based on a simple comparison of distances between different features of the face (e.g. eyes, nose, mouth etc). This face template category of algorithms has the disadvantage that their accuracy is directly influenced by the accuracy of the feature detection algorithm.

#### A. DCT Based Face Recognition

The next generation of algorithms were based on the overall appearance of the face. They are usually based on projecting the face image into a face space and using classification algorithms in this face space for computing similarities between faces. One of these techniques is based on computing the DCT spectrum of the face image and then using the entire spectrum of just a part of it for classification. The classification algorithms can vary from simple distances between DCT coefficients vectors or more complex algorithms employing neural network techniques [7], or support vector machines [5,8]. We remark that as most digital cameras feature hardware to generate a local DCT transform on image data that such techniques can be particularly advantageous with respect to computational efficiency for in-camera implementation.

A series of preliminary tests of DCT-based face recognition algorithms using Euclidean and L1 distance metrics were performed to determine similarity measures for detected face regions. It was noted that the tested algorithm is robust to small variations in rotation and translation of the face but is strongly influenced by variations in illumination. This influence persists even when histogram equalization is applied to the image prior to computing the DCT spectrum. The advantages of the tested algorithm are its simplicity and the straightforward implementation in hardware (taking advantage of the DCT hardware present in practically all digital cameras to support JPEG image compression). The main disadvantage is the low accuracy when working with consumer images with a high degree of face variation.

#### B. Component Analysis (Statistical) Techniques

Another important group of algorithms are called sub-space methods. Examples include principal component analysis (PCA) or eigenface techniques [3], linear discriminate analysis (LDA) or fisherface techniques [4] and independent component analysis (ICA) [6]. The main disadvantage of these methods is that these components are data dependent so every time the collection of faces changes the basis vectors for projection should be recomputed which leads to significant workflow issues for in-camera applications. The face projections can be classified using simple distances or more complex algorithms.

We tested the PCA approach with Euclidean distance on several collections of images. The results were good for standard face recognition databases with limited variations but for consumer images the accuracy was significantly lower. Where the basis vectors are computed offline and they are only

stored on the camera the projection of new faces can be done very easily using only vector multiplications. Unfortunately the key difficulty is how to dynamically incorporate new data and refine the basis vectors without retraining the entire data set after each new image acquisition [17, 18].

An important result demonstrated during this research is that we can use an incremental PCA algorithm for updating the PCA data without completely retraining the collection of images. This algorithm could make possible the use of PCA algorithm for in-camera face recognition and other applications with limited resources. Further details are given in [17] and a journal paper has been submitted on this important topic.

#### C. Hidden Markov Model Techniques

One possible alternative to embedded face recognition that we tested is a 2D variant of Hidden Markov Models (HMMs) called embedded HMMs (EHMM). The main advantage of the HMMs is that the models for each person are build independently. So every time we want to add a new person to the collection we just have to add a new model without modifying the other models.

The observations needed to train the models are the DCT coefficients which are fast to compute, and the wavelet coefficients, which are more robust to small variations. By proper training these models can cope with face variations which is very important when working with consumer images.

### IV. EMBEDDED HMM FACE MODELS

The theory of 1-D HMMs was developed in the 1960s by Baum et al.[10]–[13] and has earned its popularity mainly due to the successful application in speech recognition. HMMs are a set of statistical models used to characterize the statistical properties of a signal. HMMs consist of two interrelated processes: an underlying, unobservable Markov chain with a finite number of states, a state transition probability matrix and an initial state probability distribution, and a set of observations, defined by the observation density functions associated with each state [10].

The HMM classification is based on modeling the pattern to be recognized as a process represented by a succession of states. The states represent parts of the pattern and by analyzing them we can obtain sets of observations which are used to train models and further can be used for classification

This paper uses a variant of 2D HMMs called embedded HMMs. The embedded HMM was first introduced for character recognition by Kuo and Agazzi in [11] and has a large applicability in pattern recognition involving two dimensional data. The embedded HMM is a generalization of the classic HMM, where each state in the one dimensional HMM is itself an HMM. Thus, an embedded HMM consists of a set of super states along with a set of embedded states. The super states model the two dimensional data along one direction, while the embedded HMMs model the data along the other direction. The elements of an embedded HMM are:

- A set of  $N_0$  super states,  $S_0 = \{S_{0,i}\}, 1 \leq i \leq N_0$

- The initial probabilities of the super states  $\Pi_0 = \{\pi_{0,i}\}$  where  $\pi_{0,i}$  is the probability of being in super state  $i$  at time zero
- The transition probability matrix,  $A_0 = \{a_{0,ij}\}$  where  $a_{0,ij}$  is the probability of transitioning from super state  $i$  to super state  $j$
- The parameters of the embedded HMM for the super state  $k$ ,  $1 \leq k \leq N_0$ ,  $A^k = (\Pi_1^k, A_1^k, B^k)$ .

Using a shorthand notation, an embedded HMM is defined as the triplet  $\lambda = (\Pi_0, A_0, A)$ , where  $A = \{A^1, A^2, \dots, A^{N_0}\}$ .

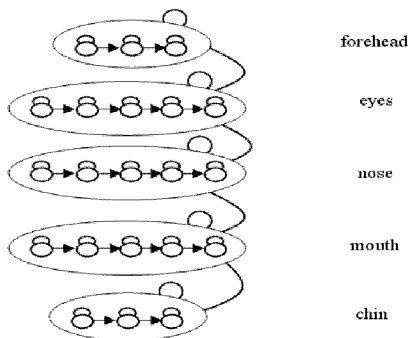


Figure 1. Embedded HMM for face recognition

This model is appropriate for face images since it exploits an important facial characteristic: frontal faces preserve the same structure of "super states" from top to bottom, and also the same left-to-right structure of "states" inside each of these "super states" [12, 13]. The state structure of the face model and the non-zero transition probabilities of the embedded HMM are shown in Fig. 1 (the configuration presented is 5 super states, each with 3, 6, 6, 6, 3 states respectively). Each state in the overall top-to-bottom HMM is assigned to a left-to-right HMM.

#### A. The observation vectors

The observation sequence for a face image is formed from image blocks that are extracted by scanning the image from left-to-right and top-to-bottom. Adjacent image blocks overlap in the vertical direction, and in the horizontal direction.

Previous experiments performed for different features, like pixels values, PCA and DCT features, showed that the best results were obtained with DCT. In this set of experiments, we compared the results for two types of observation vectors: frequencies in the 2D-DCT domain, and Daubechies wavelets, for different numbers of features used for training.

#### B. Training the face models

Each individual in the database is represented by an embedded HMM face model. A set of images representing different instances of the same face is used in the training phase. The training stages employed are:

- First, the data is uniformly segmented: the observation of

the overall top-to-bottom HMM are segmented in  $N_0$  vertical super states, then the data corresponding to each of the super states is segmented from left to right into  $N_1^k$  states. After segmentation, the initial estimates of the model parameters are obtained.

- At the next iteration, a doubly embedded Viterbi algorithm replaces the uniform segmentation. Viterbi algorithm evaluates the likelihood of the best match between the given image observations and the given HMM,  $P(O, Q | \lambda)$ , and performs segmentation of image observations by HMM states. The segmentation is done on the basis of the match found.
- The model parameters are estimated using a Euclidean distance to group vectors around the existing mixtures centers.
- The iteration stop and the parameters of the embedded HMM are estimated, when the Viterbi segmentation likelihood at consecutive iterations is smaller than a threshold.

#### C. Face recognition System

In the recognition phase, the remaining set of images not used in the training process is considered to determine the recognition performances of the system. After extracting the observation vectors as in the training phase, the probability of the observation sequence given an embedded HMM face model is computed via a doubly embedded Viterbi recognizer. The model with the highest likelihood is selected and this model reveals the identity of the unknown face.

### V. EXPERIMENTAL RESULTS

The tests were performed on a database consisting of 560 pictures, corresponding to 56 people's faces, with different head orientations, illumination and facial expressions, built by combining different collections.

Prior to analysis, a face detector[2] was applied on the entire image to extract only the face region and no other method was used to align the faces.

The illumination variation in the database was normalized using the Contrast-Limited Adaptive Histogram Equalization (CLAHE)[14] normalization technique.

CLAHE is based on examining for each pixel the histogram of an image region centered in it and assigning a new intensity for the pixel based on rank of its intensity and its histogram. The histogram is a modified form of the ordinary histogram in which the contrast enhancement induced by the method at each intensity value is limited to a selectable maximum.

Prior to recognition all faces were resized to standard 128x128 pixels and a fixed ellipse mask was applied to the face region. The EHMM architecture used for testing is presented in Fig. 1 having 5 super states with 3-6-6-6-3 states respectively.

Two tests were performed, using as observation vectors the 2D DCT coefficients extracted from each block and the Daubechies wavelets also extracted on each block. The DCT coefficients have the advantage of being very fast to compute

and the wavelet coefficients give a better description (spatial and frequency) of the block.

The recognition rates obtained for different number of faces used to build the person models are given in the next tables:

Table 1. Recognition Rates for DCT observations

Train vs Test	1vs5	2vs5	3vs5	4vs5	5vs5
Recog. Rates	66.79%	77.50%	83.57%	86.07%	88.21%

Table 2. Recognition Rates for wavelet observations

Train vs Test	1vs5	2vs5	3vs5	4vs5	5vs5
Recog. Rates	68.57%	77.50%	82.14%	86.43%	88.21%

It can be noted that both types of coefficients have similar recognition rates. For an embedded implementation, especially for in-camera applications, the DCT coefficients are preferred because they can take advantage of already existing hardware modules for their computation.

## VI. PRACTICAL IN-CAMERA IMPLEMENTATION

The principle architecture and computational elements of a practical in-camera person recognition system are illustrated in Fig 2 below. For development purposes these system modules are developed on desktop PCs and system testing and integration is performed over a local network of desktop clients, using a Linux server to implement the back-end functionality. Specific client-side modules of the system can then be readily ported and tested on various commercially available digital camera platform.

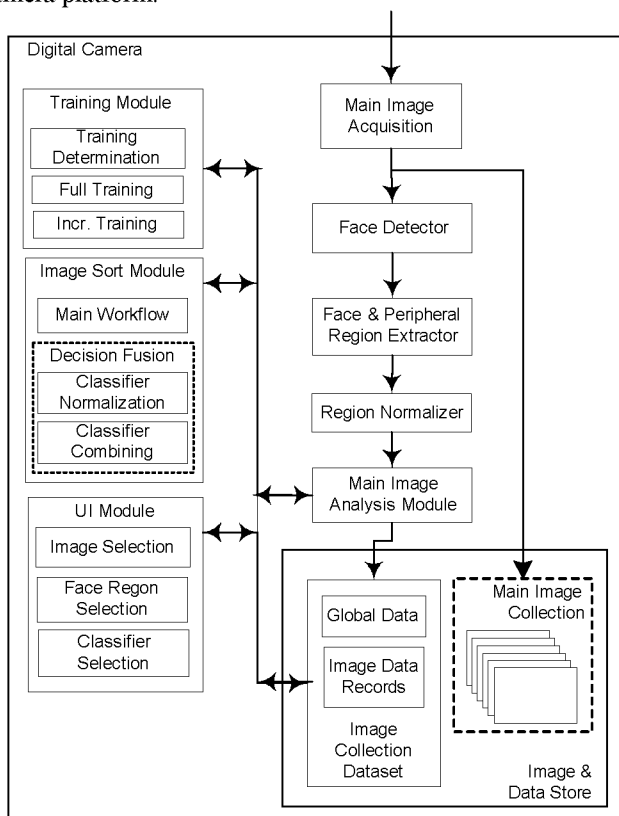


Figure 2. In-camera implementation

Following the main image acquisition process a copy of the acquired image is saved to the main image collection which will typically be stored on a removable compact-flash or multimedia data card. The acquired image is also passed to a face detector module followed by a face region extraction module and a region normalization module. The extracted, normalized regions are next passed to the main image analysis module which generates an image data record for the current image. The main image analysis module may also be called from the training module and the image sorting/retrieval module.

A UI module facilitates the browsing & selection of images, the selection of one or more face regions to use in the sorting/retrieval process. In addition classifiers may be selected and combined from the UI module.

Although currently we have only tested a limited set of face recognition classification modules it should be noted that all the system modules were designed for possible in-camera implementation either in firmware or potentially as dedicated hardware modules.

## REFERENCES

- [1] P. Corcoran, G. Costache "Automated sorting of consumer image collections using face and peripheral region image classifiers" Consumer Electronics, IEEE Transactions on Volume 51, Issue 3, Aug. 2005 Page(s):747 – 754.
- [2] <http://fotonation.com/index.php?module=product&item=23>
- [3] M. Turk and A. Pentland. Eigenfaces for recognition. J. of Cognitive Neuroscience, 3(1) pp: 71-86, 1991.
- [4] N. Belhumeur, J. Hespanha and D. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, Proc. ECCV, pp: 45-58, 1996
- [5] B. Heisele, P. Ho, T. Poggio, Face Recognition with Support Vector Machines: Global versus Component-based Approach, Proc. of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, Vol. 2, 09-12 July 2001, Vancouver, Canada, pp: 688-694
- [6] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face Recognition by Independent Component Analysis, IEEE Trans. on Neural Networks, Vol. 13, No. 6, November 2002, pp: 1450-1464
- [7] S.H. Lin, S.Y. Kung and L.J. Lin, Face recognition/detection by probabilistic decisionbased neural network. IEEE Trans. Neural Netw. 8, pp: 114-132, 1997
- [8] P.J. Phillips, Support vector machines applied to face recognition. Adv. Neural Inform. Process. Syst. 11, pp: 803-809, 1998
- [9] M. D. Kelly, Visual identification of people by computer. Tech. rep. AI-130, Stanford AI Project, Stanford, CA 1970
- [10] L.R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, Volume 77, Issue 2, Feb. 1989 pp. 257 – 286.
- [11] S. Kuo and O. Agazzi, Keyword spotting in poorly printed documents using pseudo 2-D Hidden Markov Models, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, pp. 842-848, August 1994.
- [12] A.V. Nefian, M.H. Hayes III, An embedded HMM-based approach for face detection and recognition, IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP Proceedings., 1999, pp. 3553-3556 vol.6 .
- [13] A.V. Nefian, M.H. Hayes III, Maximum likelihood training of the embedded HMM for face detection and recognition, International Conference on Image Processing, 2000. Proceedings. 2000, 10-13 Sept. 2000 pp. 33 - 36 vol.1.
- [14] SM Pizer, EP Amburn, JD Austin, et al. Adaptive histogram equalizations and its variations, Comput Vision Graph Image Process 1987; 39:355-368
- [15] H. Rowley, S. Baluja, and T. Kanade. "Neural network-based face detection" IEEE Patt. Anal. Mach. Intell., Vol. 20, pp.22-38, 1998
- [16] P. Viola, M. J. Jones "Rapid Object Detection using a Boosted Cascade of Simple Features". IEEE CVPR, pp. 511-518, 2001.
- [17] Costache, G. *PhD Thesis*, submitted to NUI, Galway, Ireland.