

# Pseudo 2D Hidden Markov Model and Neural Network Coefficients in Face Recognition

Domenico Daleno<sup>1,2</sup>, Lucia Cariello<sup>1,2</sup>,

Marco Giannini<sup>1,2</sup> and Giuseppe Mastronardi<sup>1,2</sup>

*<sup>1</sup>Department of Electrotechnical and Electronics, Polytechnic of Bari,  
Via Orabona, 4 – 70125 Bari – Italy*

*<sup>2</sup>e.B.I.S. s.r.l. (electronic Business In Security), Spin-Off of Polytechnic of Bari,  
Via Pavoncelli, 139 – 70125 Bari – Italy*

## 1. Personal Identification

For thousands of years, humans have instinctively used some physical characteristics (such as face, voice, posture, etc.) to recognize each other. About half the 800, A. Bertillon, chief of criminal identification section of the Paris police, plans to use some measures of the human body (height, length of arms, feet, fingers, etc.) to identify those responsible crimes. Towards the end of the nineteenth century, this original idea was further developed through the discovery (due studies F. Galton and E. Henry) the distinctiveness of fingerprints: they uniquely identify a person. Today, in full digital era, huge numbers of people use individual recognition techniques based on the identification of human characteristics, not only in justice but in civil and military applications. In fact, the only way to conclusively identify an individual is to recognize the personal characteristics. These are defined biometric features and, the technology behind this identification is called Biometric. The term Biometric, from the greek bios (life) and meters (measure), in computer sense, means the automatic identification or verification of the identity of a person based on physical characteristics and/or behavioral (CNIPA, 2004).

Biometric feature is described as a physiological or behavioral characteristic that can be measured and subsequently identified to confirm the identity of a person. We can then divide the biometrics in:

- physical biometric: it is that based on data derived from measurements made on a person's physical characteristics such as iris, fingerprint, facial features, hand or other;
- behavioral biometric: it is that based on aspects linked to behavioral characteristics such as, for example, the issue of voice, dynamic signing, or the type of gait.

As each biometric process starts with a preliminary phase called "enrollment" in which, generally, the person must provide the biometric system, through a sensor, its characteristic physical and behavioral, which is then converted into a mathematical model (template), two operating modes of biometrics are:

- 1:1 (one-to-one) for which the generated data by the biometric sensor are compared with a single template, creating, so, a verification process (Verification);
- 1: N (one-to-many) for which data are compared with a set of templates contained in a file, realizing, so, a process of identification (identification).

It is essential to emphasize that in biometric field two are terms usually used:

- Physical access: procedure for establishing ownership of the person entering a room, building area or area;
- Logical access: procedure for establishing ownership of the subject to make use of a computer resource.

For example, an employee of a firm could enter the office (physical access) via a biometric check between his physical characteristic (such as a fingerprint) and that deposited on a smart-card (process 1:1). To gain access to his computer (logical access) the same employee's fingerprint could be compared with that of authorized users, stored in archive (1: N) (Bossi, 2002; Maio, 2004).

### 1.1 Biometric Process

Biometric systems are characterized by a process of using that, in principle, it can be traced to the comparison operation of a physical characteristic or behavioral acquired by a person, with one or more of the same samples previously recorded. Both the recording that the comparison is made according to the following sequence of steps (CNIPA, 2004):

- ✓ *Stage of Registration (Enrollment)*: in the process of enrollment, the user provides the biometric system a physical or behavioral feature by a capture device (such as a fingerprint scanner or a video camera). The sample is processed to extract the distinctive informations, which form the so-called template that can be defined as a mathematical representation of biometric data. The template consists essentially of a sequence of numbers from which it is generally impractical his reconstruction and it is, theoretically, comparable to a user's "physical password".

At the end of the enrollment process, the template is registered. The registration is the most difficult step because of the importance of the choices to be made. First is necessary to identify as to save the template: because of the sensitivity of data and the possible impact on privacy, the information should be encrypted. Second is indispensable determined where to store and where to save the model, for example on a chip card in a database, a local workstation or directly on the capture device.

The different possibilities lead to restrictions: if a system that must handle a large number of users is used, the latter two types are not applicable to matters concerning the physical size and required computing power. By using a database, is important to consider that the data could be stolen and used in a manner not acceptable. Saving in a chip can be a good solution; however, is necessary to sign digitally the saved template and to apply security techniques which take into account the fault-based attacks (Bossi, 2002);

- ✓ *Verification step*: During the verification process, the acquisition of the sample and extraction of the template are made as before. The latter is compared with that already acquired to obtain both an authentication and recognition.
- ✓ *Authentication phase*: if the objective is the subject's authentication , the biometric system attempts to provide an answer to the question "The person is who he claimed to be?",

making a comparison 1 to 1 between the template of the subject and the reference template stored in the archive (or on a smart card). Authentication requires that the identity is provided, for example, typing a username or a pin and the output of the comparison algorithm is a score, which is positive if it occurs above a certain threshold, and negative if below this threshold. The threshold for comparison is an adjustable parameter of the system (CNIPA, 2004).

- ✓ *Recognition/identification phase*: in this case, the system determines the user's identity, or attempts to provide an answer to the question "Who is the user?", making a lot of confrontations with the biometric data models registered in its archives. When the search algorithm produces as output a score higher than the so-called "threshold", is reported a match (called "matching" or "hit") (CNIPA, 2004). Authentication is generally a cooperative process (ouvert), while identification may also be a poster or even hidden from users (covert). While in the cooperative process the subject voluntarily manifest his own identity, usually to go to a place (physical access) or use a service (logical access); in the case of hidden biometrics, the physical and/or behavioral characteristics are matched, without the person knows, with those stored in an archive.
- ✓ *Performance Mesurement*: in this performance of a biometric system are evaluated according to three parameters: size, speed and accuracy (Bossi, 2002). The size of the model have relevance to extract device storage used, consider, for example to smart-card having a memory limited. The speed with which gives a positive or negative response is discriminating about the possible use in identification rather than verification. Accuracy is a rather critical parameter to determine because of the approach probabilistic biometric systems adopted in the choice. The types of errors that can make a biometric system are essentially two: False acceptances, an unauthorized user is authenticated by the system because its footprint is quite similar to a model previously filed; False discards, an authorized user is rejected by the system because its footprint is not sufficiently similar to the model with which it was compared.

## 1.2 Biometric Techniques

Currently the efforts of the scientific community and industrial research are oriented to the study of those variables that permit reliable identification of individuals. Biometric identification techniques are indeed aimed at identifying a person based on its unique physiological or behavioral characteristics, difficult to alter or simulate. The most common evaluate the follow features:

- Fingerprints;
- Iris
- Retina vasculature
- Dynamics of attaching the signature
- Face
- Hand geometry
- Vocal timbre
- Multiple biometrics

### 1.3 Face Recognition

The recognition of facial features is perhaps one of biometric technologies more fascinating and that users consider psychologically less repulsive. System for face recognition is based on the physical characteristics of the face and is then the closest, in theory, to the human concept of "personal recognition". The enrollment usually takes a few seconds that are required to frame more static images of the face. Some systems can classify the user from multiple angles obtaining a three dimensional model of the face. This last, according to the different acquisition modes, varying in size from 100 to 3,500 byte (CNIPA, 2004). The user's acceptance of the feature based biometric recognition is generally high, since the natural and not invasive nature of the acquisition method. The sensor's prices may also be in the range of the hundreds euro's for logical access and personal computer systems, but they can remarkably increase for more sophisticated systems. Moreover the face recognition biometric technique has the advantage to be low invasiveness (no physical contact) and to provide the possibility of acquiring a distance a subject to recognize. Usually the first step of any fully automatic system that analyzes the information contained in faces, e.g. identity verification, is the Face Detection. Face detection is concerned with finding whether or not there are any faces in a given image (usually in gray scale) and, if present, return the image location and content of each face.

#### 1.3.1 Face Recognition Phases

In general, facial recognition can be decomposed into four phases (Medugno et al., 2007):

- *Pre-processing*: This means ensuring that the image which is applied to the recognition process meets certain required standards: for such that the face is located in the center of the image and provided part of the same; that the background satisfies certain constraints, and so on. Usually this phase is done by sampling equipment designed to image through mechanisms that tend to prevent the user from providing distorted images: an example may be the sensors necessary to capture the image when the subject is an acceptable distance.
- *Phase segmentation or localization*: is the exact location of the face or certain parts of it. This phase arises from the need to characterize, through some characteristic features, the face of a subject.
- *Feature Extraction Phase*: maybe it is the core of the whole face recognition process. A feature it's a characteristic useful for distinguish a face from another. It can be extracted from the image through different kind of processes. Usually, higher is amount of extracted features, the higher is the capacity of discrimination between similar faces. Some interesting features are, for example, the eyes or hairs color, the nose or the mouth shape. Those features are usually referred as locals because they refer to a particular and restricted area of the image.
- *Recognition Phase*: once the image is associated with an array of values, the recognition problem reduces itself to a widely studied problem in the past literature: the main part of those is then mainly related to the features extraction. The recognition problem can be divided into three phases: deciding over which features the recognition will be done; automatic extracting the chosen parameters from the face digitalized image; classifying the faces over the acquired parameters base.

## 2 State of art of Two - Dimensional Face Recognition

The efforts of researchers over the past 30 years have resulted in many sophisticated and mature 2D face recognition algorithms. In this section it is represented a brief description recurring methods existing in literature for face recognition.

The Principal Component Analysis (PCA) is one of the most successful techniques that have been used in image recognition and compression. PCA is a statistical method under the broad title of *factor analysis*. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. The PCA techniques consist of: eigenfaces in which the face images are projected onto a features space that best encodes the variation among known faces images, that is use a nearest neighbor classifier (Turk & Pentland, 1991), (Craw & Cameron, 1996); feature-line based methods, which replace the point-to-point distance with the distance between a point and the feature line linking two stored sample points (Li & Lu, 1999); Fisherfaces (Swets & Weng, 1996; Belhumeur et al., 1997; Zhao et al., 1998), which use Linear/Fisher Discriminant Analysis (FLD/LDA) (Liu & Wechsler, 2000); Bayesian methods, which use a probabilistic, distance metric (Moghaddam & Pentland, 1997); and SVM methods, which use a support vector machine as the classifier (Phillips, 1998). Utilizing higher-order statistics, Independent Component Analysis (ICA) is argued to have more representative power than PCA, and hence may provide better recognition performance than PCA (Bartlett et al., 1998). Being able to offer potentially greater generalization through learning, neural networks/learning methods have also been applied to face recognition. One example is the Probabilistic Decision-Based Neural Network (PDBNN) method (Lin et al., 1997) and the other is the evolution pursuit (EP) method (Etemad & Chellappa, 1997).

The category of feature based (structural) matching methods, using the width of the head, the distances between the eyes and from the eyes to the mouth, etc. (Kelly, 1970), or the distances and angles between eye corners, mouth extrema, nostrils, and chin top (Kanade, 1973). More recently, a mixture-distance based approach using manually extracted distances was reported (Manjunath et al., 1992; Cox et al., 1996). Without finding the exact locations of facial features, Hidden Markov Model (HMM) based methods use strips of pixels that cover the forehead, eye, nose, mouth, and chin (Samaria & Young, 1994), (Samaria, 1994; Nefian & Hayes III, 1998). (Nefian & Hayes III, 1998) reported better performance than (Samaria, 1994) by using the KL projection coefficients instead of the strips of raw pixels. One of the most successful systems in this category is the graph matching system (Wiskott et al., 1997), (Okada et al., 1998) which is based on the Dynamic Link Architecture (DLA). Using an unsupervised learning method based on a Self Organizing Map (SOM), a system based on a convolutional neural network (CNN) has been developed (Lawrence et al., 1997).

Moreover, in the hybrid method category, we will briefly review the modular eigenface method (Pentland et al., 1994), an hybrid representation based on PCA and Local Feature Analysis (LFA) (Penev & Atick, 1996), a flexible appearance model based method (Lanitis et al., 1995), and a recent development (Huang et al., 2003) along this direction. In (Samaria, 1994), the use of hybrid features by combining eigenfaces and other eigenmodules is explored: eigeneyes, eigenmouth, and eigen-nose. Though experiments show slight improvements over holistic eigenfaces or eigenmodules based on structural matching, we

believe that these types of methods are important and deserve further investigation. Perhaps many relevant problems need to be solved before fruitful results can be expected, e.g., how to optimally arbitrate the use of holistic and local features.

Many types of systems have been successfully applied to the task of face recognition, but they all have some advantages and disadvantages. Appropriate schemes should be chosen starting from the specific requirements of a given task. Most of the systems reviewed here focus on the subtask of recognition, but others also include automatic face detection and feature extraction, making them fully automatic systems (Moghaddam & Pentland, 1997; Wiskott et al., 1997; Lin et al., 1997).

### 3. Artificial Neural Network

An artificial neural network is a system that tries to reproduce the operation of biological neural networks or, in other words, is an emulation of the biological neural system. This approach give the chance of performing tasks that a linear program is not able to do exploiting its capability of learning with no needs of writing new code lines. These advantages have a cost. They need a good training to operate correctly and their computing time can be high for large Neural Networks. According to what has been said, an Artificial Neural Network is an adaptive nonlinear system that learns to perform a function from data. Adaptive means that the system parameters are changed during operation, normally called the training phase. After the training phase the Artificial Neural Network parameters are fixed and the system is deployed to solve the problem at hand (the testing phase). The Artificial Neural Network is built with a systematic step-by-step procedure to optimize a performance criterion or to follow some implicit internal constraint, which is commonly referred to as the learning rule. The input/output training data are fundamental in neural network technology, because they convey the necessary information to "discover" the optimal operating point. The nonlinear nature of the neural network processing elements (PEs) provides the system with lots of flexibility to achieve practically any desired input/output maps. In the case of supervised Neural Networks, in order to train an ANN, an input is presented to the neural network and a corresponding result is set at the output. The error is the difference between the desired response and the actual system output. The error information is fed back to the system so that it can adjust its parameters in a systematic way, following the adopted learning rule.

The process is repeated until the performance is acceptable. It comes clear that the performances of the trained Neural Network would be heavily influenced by the dataset that was used for the training phase. If it does not cover a significant portion of the operating conditions or if they are ambiguous, then neural network technology is probably not the right solution. On the other hand, if there is plenty of data and the problem is poorly understood to derive an approximate model, then neural network technology is a good choice. This operating procedure should be contrasted with the traditional engineering design, made of exhaustive subsystem specifications and intercommunication protocols. In artificial neural networks, the designer chooses the network topology, the performance function, the learning rule, and the criterion to stop the training phase, while the system automatically adjusts the parameters.

Thus it is difficult to bring a priori information into the design and, when the system does not work properly, it is also hard to refine the solution in a following step. At the same time,

ANN-based solutions are extremely efficient in terms of development time and resources, and in many difficult problems artificial neural networks provide performance that is difficult to match with other technologies. At present, artificial neural networks are emerging as the technology of choice for many applications, such as pattern recognition, prediction, system identification, and control.

When creating a functional model of the biological neuron, there are three basic components of importance. First, the synapses of the neuron are modelled as weights. Operating in this way, the strength of the connection between an input and a neuron is noted by the value of the weight. Inhibitory connection will have negative weight values, while positive values designate excitatory connections. The next two components model the actual activity within the neuron cell. An adder sums up all the inputs modified by their respective weights. This activity is referred to as linear combination. Finally, an activation function controls the amplitude of the output of the neuron. An acceptable range of output is usually between 0 and 1, or -1 and 1.

Mathematically, this process is described in the Fig. 1:

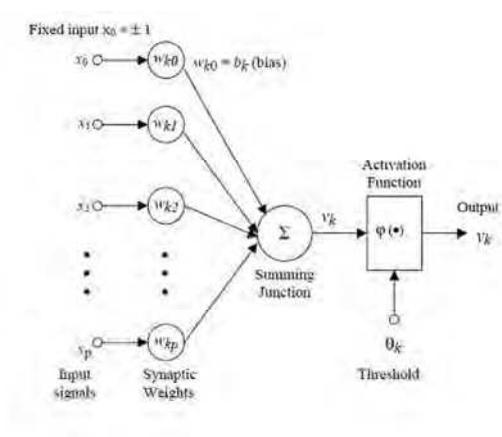


Fig. 1. Artificial Neural Network Process

From this model the interval activity of the neuron can be represented as:

$$v_k = \sum_{j=1}^p w_{kj} x_j \tag{1}$$

The output of the neuron,  $y_k$ , would therefore be the outcome of some activation function on the value of  $v_k$ .

The activation function is functions that compel the output of a neuron in a neural network inside certain values (usually 0 and 1, or -1 and 1). In general, there are three types of activation functions, denoted by  $\Phi()$ . First, there is the **Threshold Function** which takes on a value of 0 if the summed input is lower than a certain threshold value ( $v$ ), and the value 1 if the summed input is greater than or equal to the threshold value.

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \quad (2)$$

Secondly, there is the **Piecewise-Linear function**. This function too admits values of 0 or 1 as input, but can also take on values belonging to that interval, depending on the amplification factor in a certain region of linear operation.

$$\varphi(v) = \begin{cases} 1 & v \geq \frac{1}{2} \\ v & -\frac{1}{2} > v > \frac{1}{2} \\ 0 & v \leq -\frac{1}{2} \end{cases} \quad (3)$$

Thirdly, there is the **sigmoid function**. This function can range between 0 and 1, but it is also sometimes useful to use the -1 to 1 range. An example of the sigmoid function is the hyperbolic tangent function.

$$\varphi(v) = \tanh\left(\frac{v}{2}\right) = \frac{1 - \exp(-v)}{1 + \exp(-v)} \quad (4)$$

The pattern of connections between the units and the propagation of data are clustered into two main class:

- **Feed-forward neural networks**, where the data flow from input to output units is strictly feedforward. The data processing can extend over multiple layers of units, but no feedback connections are present, that is, connections extending from outputs of units to inputs of units in the same layer or previous layers.
- **Recurrent neural networks** that do contain feedback connections. Contrary to feed-forward networks, the dynamical properties of the network are important. In some cases, the activation values of the units undergo a relaxation process such that the neural network will evolve to a stable state in which these activations do not change anymore. In other applications, the change of the activation values of the output neurons are significant, such that the dynamical behaviour constitutes the output of the neural network.

#### 4. Hidden Markov Models

The Hidden Markov Models are stochastic models which provide a high level of flexibility for modelling the structure of an observation sequence. They allow for recovering the (hidden) structure of a sequence of observations by pairing each observation with a (hidden) state. Hidden Markov Models (HMMs) represent a most famous statistical pattern recognition technique and can be considered as the state-of-the-art in speech recognition. This is due to their excellent time warping capabilities, their effective self organizing learning capabilities and their ability to perform recognition and segmentation in one single step. They are used not only for speech and handwriting recognition but they are involved in modelling and processing images too. This is the case of their use in the face recognition field.

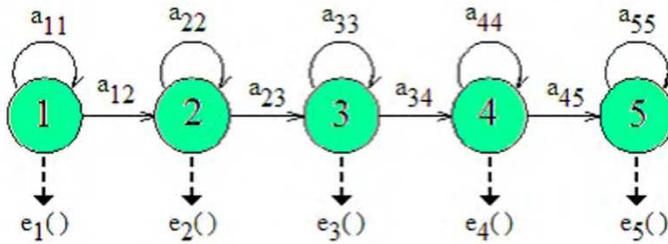


Fig. 2. "Left to Right" Hidden Markov Model - 5 state

#### 4.1 One - Dimensional Hidden Markov Models

The HMM are characterised by two interrelated processes (Samaria & Young, 1994):

1. An unobservable Markov chain with a finite number of states, a state transition probability matrix and an initial state probability distribution.
2. A set of probability density functions for each state.

The elements that characterised a HMMs are:

- $N = |S|$  which represent the number of states of the model. Where  $S$  is the set of the states and can be shown as  $S = \{s_1, s_2, \dots, s_n\}$ , where  $s_i$  is one of the states that can be employed by the model. To describe the system,  $T$  observation sequences are used, where  $T$  is the number of observations. The state of the model at time  $t$  is given by  $q_t$  in  $S$ ,  $1 < t < T$ ;
- $M = |V|$  is the number of different observation symbols. If  $V$  is the set of all possible observation symbols (also called the codebook of the model), then  $V = \{v_1, v_2, \dots, v_M\}$ ;
- $A = \{a_{ij}\}$  is the state transition probability matrix, where  $a_{ij}$  is the probability that the state  $i$  became the state  $j$ :

$$a_{ij} = p(q_t = s_j \mid q_{t-1} = s_i) \quad (5)$$

where  $1 \leq i, j \leq N$ , with constraint  $0 \leq a_{ij} \leq 1$ , and  $\sum_{j=1}^N a_{ij} = 1$ ,  $1 \leq i \leq N$

- $B = \{b_j(k)\}$  the observation symbol probability matrix,  $b_j(k)$  is the probability to have the observation  $k$  when the state is  $j$ :

$$b_j(k) = p(O_t = v_k \mid q_t = S_j) \quad (6)$$

where  $1 \leq j \leq N$ ;  $1 \leq k \leq M$ ; and  $O_t$  is the observation symbol at time  $t$ .

- $\Pi = \{\pi_1, \pi_2, \dots, \pi_N\}$  is the initial state distribution:

$$\pi_i = p(q_j = S_i) \quad (7)$$

where  $1 \leq j \leq N$ .

Using a shorthand notation, a HMM is defined by the following expression:

$$\lambda = (A, B, \pi). \quad (8)$$

The training of the model, given a set of sequences  $\{O_i\}$ , is usually performed by means of the standard Baum-Welch re-estimation, which determines the parameters  $(A, B, \pi)$  that maximize the probability  $P(\{O_i\} | \lambda)$ .

#### 4.2 Pseudo Two-Dimensional Hidden Markov Models

Pseudo Two-Dimensional Hidden Markov Models (P2D-HMMs) are an extension of the one-dimensional HMMs, applied to two-dimensional data. Fig. 3 shows a general schema of P2D-HMMs, which are also known as planar HMMs, that are stochastic automata with a two-dimensional arrangement of the states. A planar HMM can be seen as a vertical One-dimensional HMM that links together an indefinite number of super-states.

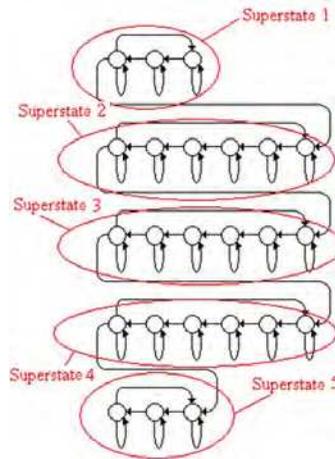


Fig. 3. General schema of a Pseudo Two-Dimensional Hidden Markov Model

Considering that a facial image can be subdivided into stripes, thus allowing the implementation of P2D-HMMs for modelling this kind of elaboration. Each stripe is aligned to one of the super-states of the P2D-HMMs, resulting in a horizontal warping of the pattern. Furthermore, the stripes can be vertically disposed, within the super-state, in a manner that the pattern related to the stripe result to be aligned to the vertical HMM states. In a similar way, it is possible to model any kind of data that can be considered represented by means horizontal stripes. The recognition process achieved by means of P2D-HMMs is pretty similar to the recognition process made with one-dimensional HMM as it was showed by Samaria (Samaria, 1994). The P2D-HMMs can be trained using the standard Baum-Welch algorithm and the recognition step can be carried out with the standard Viterbi algorithm.

The super-states is the model of the sequence of rows in the image and the linear 1D-HMMs, which are inside the super-states, are used to model each row (Nefian, 1998). The states sequence in each rows is independent from the states sequences of neighbouring rows.

Figure 3 shows the particular structure of the P2D-HMM that we use: the schema is 3-6-6-6-3, where the 1<sup>st</sup> and the 5<sup>th</sup> super-states are constituted by a left to right 1D-HMM with 3 states, while the 2<sup>nd</sup>, the 3<sup>rd</sup> and the 4<sup>th</sup> super-states are constituted by a left to right 1D-HMM with 6 states.

The formal representation of a Pseudo Two-Dimensional Hidden Markov Models can be given by the expression  $\Lambda = \{\lambda, A, B, \Pi\}$  where,

- $\lambda = \{\lambda^{(1)}, \lambda^{(2)}, \dots, \lambda^{(N)}\}$  is the set of  $N$  possible super-states in the model.
- $\lambda^i$  is a 1DHMM super-state, whose parameters are  $\lambda^i = \{s^i, V, A^i, \pi^i\}$

In different words,

- $s = \{s_1^i, s_2^i, \dots, s_N^i\}$  is the set of  $N^i$  possible states of super-state  $\lambda^i$ .
- $V = \{V_1, V_2, \dots, V_L\}$  is the output alphabet (common to all super-states). In other words, for any  $t$ , there exist  $l$  such that  $o_t = v_l$ .
- $A^i = \{a_{kl}^i\}_{k=1 \dots N^i}$  is the set of transition probabilities within super-state  $\lambda^i$ .
- $B^i = \{b_k^i(l)\}_{k=1 \dots N^i, l=1 \dots L}$  is the set of output probabilities of super-state  $\lambda^i$ .
- $\pi = \{\pi_1^i, \pi_2^i, \dots, \pi_N^i\}$  is the set of initial state probabilities of super-state  $\lambda^i$ .
- $A = \{a_{ij}\}_{ij=1 \dots N}$  is the set of transition probabilities through the states of the P2DHMM.
- $\Pi = \{\pi_1, \pi_2, \dots, \pi_N\}$  is the set of initial super-state probabilities of the P2DHMM.

Similarly to the one-dimensional model, the Pseudo two-dimensional Hidden Markov Models will associate a state sequence  $Q$  to an observation sequence  $O = \{o_{xy}\}_{x=1 \dots X, y=1 \dots Y}$ . The state sequence  $Q$  will consist of two levels.  $Q$  is primarily a super-state sequence  $Q = \{Q_1, Q_2, \dots, Q_N\}$  indicating the super-state corresponding to the sequence of lines of observation  $O = \{O_1, O_2, \dots, O_N\}$ . Each state line  $Q_y$  is composed itself of states  $q_{xy} (Q_y = \{q_{1y}, q_{2y}, \dots, q_{xy}\})$ , each of them indicating the state of the corresponding 1DHMM at a position  $(x; y)$ .

A formal expression of the parameters of a P2DHMM can be given as follows:

- Super-state transition probability:  $a_{ij} = P[q_y = \lambda^j | q_{y-1} = \lambda^i]$ .
- Initial super-state probability:  $\pi_i = P[Q_1 = \lambda^i | \Lambda]$ .
- State transition probability of super-state  $\lambda^i$ :  $a_{kl}^i = P[q_{xy} = s_l^i | q_{x-1y} = s_k^i]$ .
- State output probability of super-state  $\lambda^i$ :  $b_{ij}^i(l) = P[o_{xy} = v_l | q_{xy} = s_j^i]$ .
- Initial state probability:  $\pi_j^i = P[q_{1y} = s_j^i | \lambda^i]$ .

### 4.3 Hidden Markov Models applied to Face Recognition

The HMM can be applied to image processing. In consideration of the fact that the image can be seen as a two dimension matrix of data, according to Samaria, space sequences must be considered (Samaria, 1992). The idea is again to exploit the vertical sequential structure of a human face. A sequence of overlapping horizontal stripes are built on the image and the sequence of these stripes is labeled by means of a 1DHMM. Considering frontal face images, the facial region can be considered as the sum of 5 regions: forehead, eyes, nose, mouth and chin (Nefian & Monson, 1998).

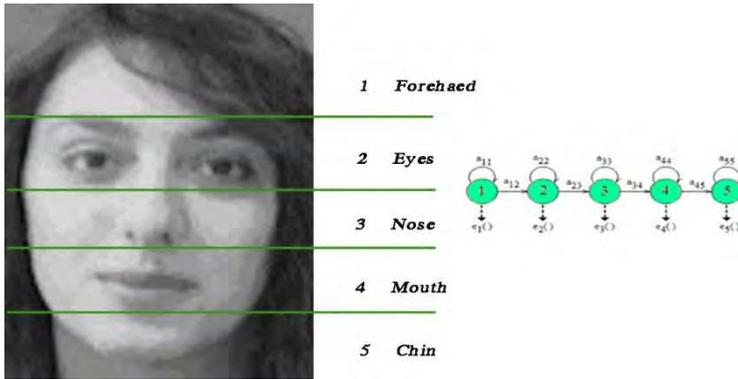


Fig. 4. The significant facial regions

Each of these facial regions (facial band) will correspond to a state in a left to right 1D continuous HMM. The Left-to-right HMM used for face recognition is shown in the previous figure. To recognize the face  $k$  the following HMM has been trained:

$$\lambda(k) = (A(k), B(k), p(k)) \tag{9}$$

The HMM should be trained for each person that we want to recognize subsequently. The HMM training, that equals to an enrolment operation for every subject of the database, requires a grey scale image of the face of each person. Each image of width  $X$  and height  $Y$  is divided into overlapping blocks of height  $L$  and width  $W$ . The amount of overlap between bounding blocks is  $M$ .

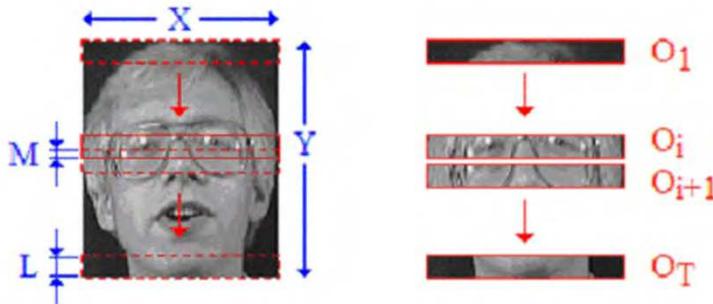


Fig. 5. Extraction of overlapping blocks from the face

The number of blocks extracted from each face image and the number of observation vectors  $T$  are the same and are given by:

$$T = \frac{(Y - L)}{(L - M)} + 1 \tag{9}$$

The system recognition rate is significantly affected by the parameters  $M$  and  $L$  that, for this reason, should be chosen accurately. Increasing the overlap area  $M$  can significantly increase the recognition rate because it allows the features to be captured in a manner that is independent of their position along the vertical axis. The choice of parameter  $L$  is more delicate. An insufficient amount of information about the observation vector could arise from a small value of the parameter  $L$  while, on the contrary, large values of  $L$  are dangerous as the probability of cutting across the features increase. However, as the system recognition rate is more sensitive to the variations in  $M$  than in  $L$ ,  $M \leq (L - 1)$  is used.

## 5. The System Proposed

The system for face recognition proposed, showed in the figure below, is an hybrid system as showed built as a cascade connection of two different systems: an Artificial Neural Network, existing in literature (Bevilacqua et al., 2006), and different representation of P2D-HMMs.



Fig. 6. The proposed hybrid system.

The system's input is an image of a person that must be recognised and the output is its identification with the corresponding rate of recognition. The experiments will be performed on a database obtained by the combination of the Olivetti Research Laboratory database (Samaria & Harter, 1994), and other profiles photos of persons disguised with dark glasses or bandage. These images are ".bmp" files in grey scales of 92x112 pixels.

The hybrid schema was built executing the following steps:

1. Training and saving of Artificial Neural Network;
2. Transformation of photos in HTK format;
3. Training of different P2D-HMM structures, and identification of the Validation Set subjects, for control a proper training of system;

### 5.1 Training and Saving of Artificial Neural Network

The considered faces are sequenced in observation windows, according to the Samaria model already described in the previous section, where the number of blocks extracted from each face image equals the number of observation vectors  $T$ , and is obtained from Eq. 9.

Table 1 collects the values of the parameters for the observation windows after the manipulation operated by this system.

<b>X = width photo = 92 pixels</b>	<b>T = number of blocks for photos = 103</b>
<b>Y = height photo = 112 pixels</b>	<b>XxY = photo dimension = 10304 pixels</b>
<b>L = height block = 10 pixels</b>	<b>XxL = block dimension = 920 pixels</b>
<b>M = blocks overlapping = 9 pixels</b>	<b>XxM = overlapping dimension = 828 pixels</b>

Table 1. Characteristic parameters of photos and blocks.

The Artificial Neural Network utilized in this system uses the EBP (Error Back Propagation) algorithm and its task is to extract the main features from the image and then store them in a sequence of 50 bits, reducing the complexity of the problem and compressing the bitmap images in order to represent them with a number of coefficients smaller than pixels. The image is a facial feature of a face image; from this area we consider 103 segments of 920 pixels that represent the observable states of the model (Bevilacqua et al, 2006).

Now all of these sections are divided into features of 230 pixels, that are the input of the Artificial Neural Network. The ANN is composed of three layers where the first layer is formed by 230 neurons, one neuron per each pixel, the hidden layer is composed by 50 units and the last layer by 230 neurons. After the training, the ANN is able to work as a pure linear function, the input of the first layer must be the same of the output of the last layer. The "compressed image" is described by 50 bits that are the outputs of an hidden layer consisting of an Heaviside function processing elements. For any window of 230 pixels we have an array of 50 elements, this means that a section of 920 pixels is compressed in a 4 sub-windows of 50 binary values array each. The matrix weights, referred to the connections between the inputs and the hidden layer, codifies the image bitmap, while the matrix weights associated to the connections between the hidden layer and the outputs, decodes the sequence of bits. Each of the 103 blocks of 920 pixels (4x230) gives 103 observation vectors with 200 coefficients (4x50) and the compression rate equals to

$$\frac{(103 \times 920)}{(103 \times 200)} = 4,6 \tag{10}$$

By observing the schema of Fig. 7 it is possible to note that the "Training Set" used for ANN is composed by 300 photos: 10 face images for each of the first 30 samples of the database. The training function is iterated 200 times for each photo and, at the end of the training phase, the neuron weights are saved in a ".bin" file. Finally the ANN is tested with other images, of the same size of the training images, representing the same subject used for the training, but, of course, different from those belonging to the training set features.

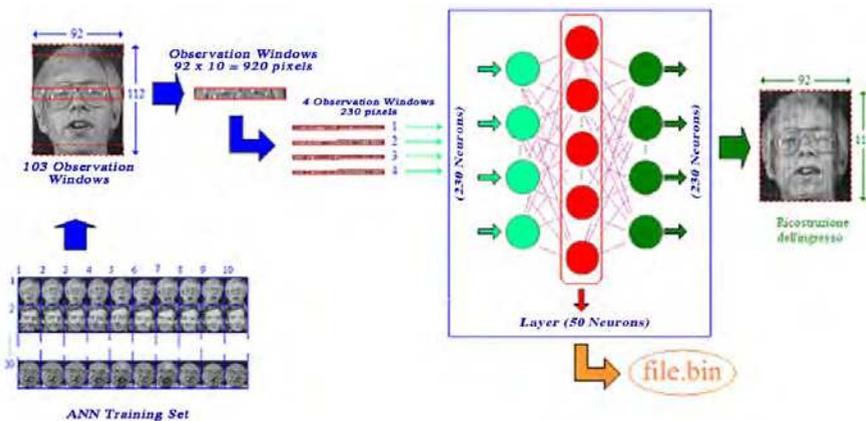


Fig. 7. Schema of the ANN training phase

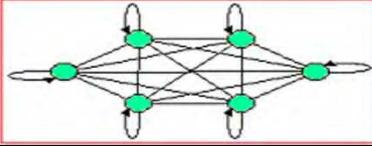
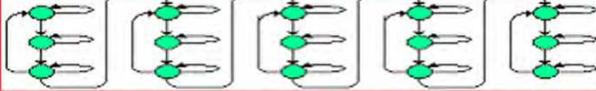
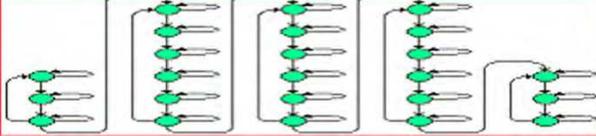
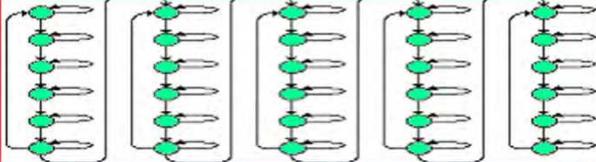
### 5.2 Transformation of Photos in HTK Format

After compressing the image containing the face into an observation vector of 103 elements of 200 binary (1/0) values, it will be computed by the Pseudo 2D Hidden Markov Models. The operations of building and manipulating the Hidden Markov models has been computed by the Hidden Markov Model ToolKit (HTK) (Young and Young, 1994). The HTK supports HMMs using both continuous density mixture Gaussians and discrete distributions and can be used to build complex HMM systems.

Finally, is necessary to transform the ANN output “.bin” file into another binary file in HTK format. The HTK like binary file has got a header, that should accomplish the HTK syntax, and 20600 coefficients (103x200), according the “Little Endian” data storage, which is commonly used by Motorola processors, IBM and Sun. Little Endian format provides the least significant byte is stored in the first memory location while the most significant byte is the last memory location.

### 5.3 Training of Different P2D-HMM Structures and Identification of the Validation Set subjects

Every subject populating the database was used to train the Pseudo 2D Hidden Markov Model and a Markov Model was associated to each of them. The different Hidden Markov Model structures were then trained. The table below reports the training results of one Ergodic HMM with 5 state and four Pseudo 2D Hidden Markov Model structures, that differs one by the others for the number of states in a super-state. The Table helps the comparison between the different performance and the choice of the structure that gives the best recognition rate. In table 2 are represented different Hidden Markov Model structures.

<p><i>HMM</i> <b>5Ergodic</b></p>					
<p>Pseudo 2D HMM <b>3-3-3-3-3</b></p>					
<p>Pseudo 2D HMM <b>3-6-6-6-3</b></p>					
<p><i>Pseudo</i> 2D HMM <b>6-6-6-6-6</b></p>					

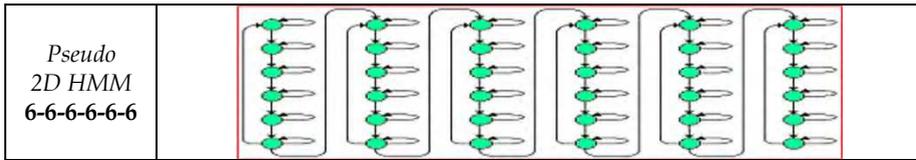


Table 2. Different Hidden Markov Models structures.

After the P2D-HMM training process was completed, it was possible to proceed with the recognition phase, according to the schema shown in Fig. 8.

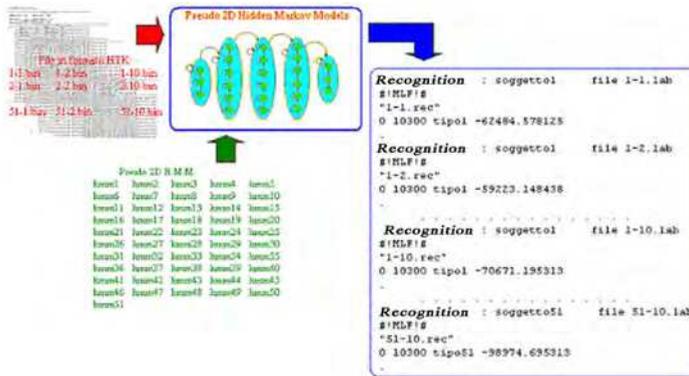


Fig. 8. Schema of recognition phase.

The Viterbi algorithm is applied to each of the P2D-HMMs, built after the training phase, using the same HTK file (26-5.bmp in Fig.9). Each of the P2D-HMMs returns a logarithmic probability value. The highest probability value identifies the P2D-HMM and so the corresponding recognised sample as showed in Figure 9.

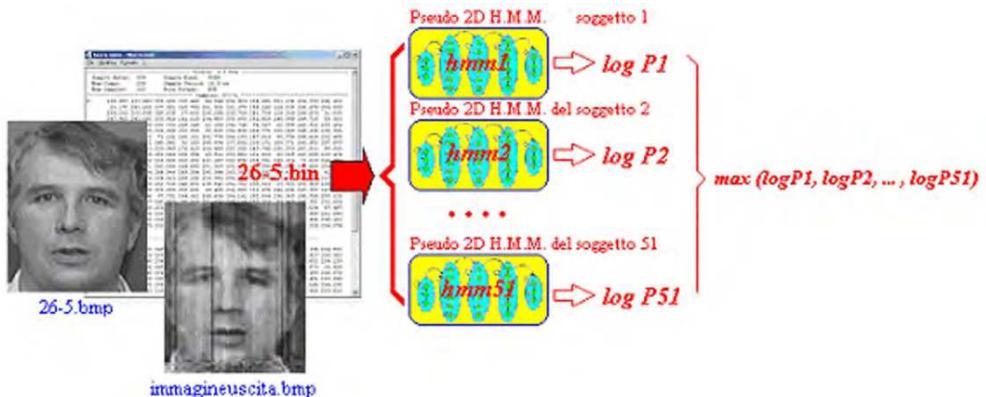


Fig. 9. Example of identification by logarithmic probability.

At the end of the process the final outcome of the identification is the recognised person and the logarithmic probability value of his similarity to the template.

## 6. Experimental Result

As said in the preceding paragraphs, different Hidden Markov Model structure was tested on a database obtained as a combination of the Olivetti Research Laboratory database together with other photos of persons camouflaged wearing dark glasses, scarf or bandage, in order to check system reliability. The results are shown in the Table 3, here below.

Hidden Markov Models	The exact identification $100 - \frac{n^{\circ}errors}{5,1}$
Pseudo 2D 3-3-3-3-3	99.80 % (1 error on 510 photo)
Pseudo 2D 3-6-6-6-3	100 %
Pseudo 2D 6-6-6-6-6	99.80 % (1 error on 510 photo)
Pseudo 2D 6-6-6-6-6-6	99.80 % (1 error on 510 photo)
5-Ergodic	98.82 % (6 error on 510 photo)

Table 3. Rates of recognition obtained from the different implemented P2D-HMMs

The recognition rate was satisfying for all the HMM tested structures, but the system using the HMM structure 3-6-6-6-3 gave a percentage of identification of 100%, that is to say that any of the 510 photo tested were properly recognized.

Subsequently was made an experimental comparison of the results obtained with the hybrid system ANN-P2DHMM (using an HMM with structure 3-6-6-6-3) with the most important face recognition algorithms proposed in the literature when applied to the ORL images.

Methods	Recognition Rate	Reference
Eigenface	90.5%	Samaria, 1994
Pseudo 2D HMM feature: gray values	94.5%	Samaria, 1994
Convolutional Neural Network	96.2%	Lawrence et al., 1997
Pseudo 2D HMM feature: DCT Coefficients	99.5%	Eickeler, 1998
Ergodic HMM + DCT	99.5%	Kohir & Desai, 1998
<b>Pseudo 2D HMM + Neural Network Coefficients</b>	<b>100%</b>	<b>This work.</b>

Table 4. Comparative results on ORL database.

Table 4 resumes the results obtained and highlights that the hybrid system that combines Artificial Neural Networks and Pseudo 2D Hidden Markov Model produced the best Recognition Rate.

This result encourages the prosecution of the research to obtain a fundamental surplus to enhance the P2D-HMMs potentiality, allowing an efficient and sure personal identification process.

## 7. Reference

- Bartlett, M.S.; Lades, H.M. & Sejnowski, T. (1998). Independent Component Representation for Face Recognition, *Proceedings of SPIE Symposium on Electronic Imaging: Science and Technology*, pp. 528-539, 1998
- Belhumeur, P.N.; Hespanha, J.P. & Kriegman, D.J. (1997). Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 19, pp. 711-720, 1997
- Bevilacqua, V., Mastronardi, G., Pedone, G., Romanazzi, G. & Daleno, D. (2006). Hidden Markov Model for Recognition using Artificial Neural Networks. Springer-Verlag, Heidelberg, New York, 2006.
- Bossi, F. (2002). Biometrica per la sicurezza. <http://microservo.altervista.org/Collaborazioni/biometrica%20per%20la%20sicurezza.pdf>, March 2002
- CNIPA. (2004). "Brevi note sulle tecnologiche biometriche". <http://www.privacy.it/cnipabiometria.html>, January 2004
- Cox, I.J.; Ghosn, J. & Yianilos, P.N. (1996). Feature-based Face Recognition Using Mixture-Distance, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 209-216, 1996
- Craw, I. & Cameron, P. (1996). Face Recognition by Computer, *Proceedings of British Machine Vision Conference*, pp. 489-507, 1996
- Etemad, K. & Chellappa, R. (1997). Discriminant Analysis for Recognition of Human Face Images. *Journal of the Optical Society of America A*, Vol. 14 (1997), pp. 1724-1733
- Kanade, T. (1973). *Computer Recognition of Human Faces*, Basel and Stuttgart: Birkhauser
- Kelly, M.D. (1970). *Visual Identification of People by Computer*, Technical Report AI-130, Stanford AI Project, Stanford, CA
- Lanitis, A.; Taylor, C.J. & Cootes, T.F. (1995). Automatic Face Identification System Using Flexible Appearance Models. *Image and Vision Computing*, Vol. 13 (1995), pp. 393-401.
- Lawrence, S.; Giles, C.L.; Tsoi, A.C. & Back, A.D. (1997). Face Recognition: A Convolutional Neural-Network Approach. *IEEE Transaction on Neural Networks*, Vol. 8 (1997), pp. 98-113
- Li, S.Z. & Lu, J. (1999). Face Recognition Using the Nearest Feature Line Method. *IEEE Transaction on Neural Networks*, Vol. 10 (1999), pp. 439-443
- Lin, S.H.; Kung, S.Y. & Lin, L.J. (1997). Face Recognition/Detection by Probabilistic Decision-Based Neural Network. *IEEE Transaction on Neural Networks*, Vol. 8 (1997), pp. 114-132
- Liu, C. & Wechsler, H. (2000). Evolutionary Pursuit and its Application to Face Recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 22 (2000), pp. 570-582

- Maio, D. (2004). Introduzione ai sistemi biometrici, <http://www.cnipa.gov.it/site/files/introduzione%20ai%20sistemi%20biometrici%20prof%20Maio.pdf>, 2004
- Manjunath, B.S.; Chellappa, R. & Malsburg, C.v.d. (1992). A Feature Based Approach to Face Recognition, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 373-378, 1992
- Medugno, V., Valentino, S. & Acampora, G. (2007). Sistemi biometrici <http://www.dia.unisa.it/professori/ads/corsosecurity/www/CORSO9900/biometria/index.htm>
- Moghaddam, B. & Pentland, A. (1997). Probabilistic Visual Learning for Object Representation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 19 (1997), pp. 696-710
- Nefian, A. V. & Monson, H. (1998). Hidden Markov Models for Face Recognition, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP98)*, 2721-2724, 1998.
- Nefian, A. V. (1998). A Hidden Markov Model-based approach for face detection and recognition. *Proposal for a Doctoral Dissertation*. 1998.
- Okada, K.; Steffans, J.; Maurer, T.; Hong, H.; Elagin, E.; Neven, H. & Malsburg, C.v.d. (1998). The Bochum/USC Face Recognition System and how it Fared in the FERET Phase III Test, In: *Face Recognition: From Theory to Applications*, Wechsler, H.; Phillips, P.J.; Bruce, V.; Soulie, F.F. & Huang, T.S., (Eds.), pp. 186-205, Springer-Verlag, Berlin
- Pentland, A.; Moghaddam, B. & Starner, T. (1994). View-based and Modular Eigenspaces for Face Recognition, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994
- Penev, P. & Atick, J. (1996). Local Feature Analysis: A General Statistical Theory for Object Representation. *Network: Computation in Neural Systems*, Vol. 7 (1996), pp. 477-500
- Phillips, P.J. (1998). Support Vector Machines Applied to Face Recognition. *Advances in Neural Information Processing Systems*, Vol. 11 (1998), pp. 803-809
- Samaria, F. (1992). Face identification using Hidden Markov Model. In *1st Year Report Cambridge University Engineering Department*, London, UK, 1992.
- Samaria, F. (1994). Face Recognition Using Hidden Markov Models. *PhD Thesis, University of Cambridge*, 1994.
- Samaria, F. & Young, S. (1994). Face HMM based architecture for face identification. In *Image and Computer Vision*, pages 537-583, 1994.
- Young, S.J. & Young, S.J. (1994). The HTK Hidden Markov Model Toolkit: Design and Philosophy. *Entropic Cambridge Research Laboratory, Ltd*, Vol. 2, p.p. 2-44
- Samaria, F. & Harter, A. (1994). Parameterisation of a Stochastic Model for Human Face Identification. *Proceedings of 2nd IEEE Workshop on Applications of Computer Vision*, Sarasota FL, December, 1994.
- Samaria, F. & Young, S. (1994). HMM Based Architecture for Face Identification. *Image and Vision Computing*, Vol. 12 (1994), pp. 537-583
- Samaria, F. (1994). *Face Recognition Using Hidden Markov Models*, PhD Thesis, University of Cambridge, UK
- Swets, D.L. & Wang, J. (1996). Using Discriminant Eigenfeatures for Image Retrieval. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 18 (1996), pp. 831-836
- Turk, M. & Pentland, A. (1991). Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, Vol. 3 (1991), pp. 72-86

- Wiskott, L.; Fellous, J.M. & Malsburg, C.v.d. (1997). Face Recognition by Elastic Bunch Graph Matching. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 19 (1997), pp. 775-779
- Zhao, W.; Chellappa, R. & Krishnaswamy, A. (1998). Discriminant Analysis of Principal Components for Face Recognition, *Proceedings of International Conference on Automatic Face*



## Face Recognition

Edited by Milos Oravec

ISBN 978-953-307-060-5

Hard cover, 404 pages

**Publisher** InTech

**Published online** 01, April, 2010

**Published in print edition** April, 2010

This book aims to bring together selected recent advances, applications and original results in the area of biometric face recognition. They can be useful for researchers, engineers, graduate and postgraduate students, experts in this area and hopefully also for people interested generally in computer science, security, machine learning and artificial intelligence. Various methods, approaches and algorithms for recognition of human faces are used by authors of the chapters of this book, e.g. PCA, LDA, artificial neural networks, wavelets, curvelets, kernel methods, Gabor filters, active appearance models, 2D and 3D representations, optical correlation, hidden Markov models and others. Also a broad range of problems is covered: feature extraction and dimensionality reduction (chapters 1-4), 2D face recognition from the point of view of full system proposal (chapters 5-10), illumination and pose problems (chapters 11-13), eye movement (chapter 14), 3D face recognition (chapters 15-19) and hardware issues (chapters 19-20).

### How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Domenico Daleno, Lucia Cariello, Marco Giannini and Giuseppe Mastronardi (2010). Pseudo 2D Hidden Markov Model and Neural Network Coefficients in Face Recognition, Face Recognition, Milos Oravec (Ed.), ISBN: 978-953-307-060-5, InTech, Available from: <http://www.intechopen.com/books/face-recognition/pseudo-2d-hidden-markov-model-and-neural-network-coefficients-in-face-recognition>

# INTECH

open science | open minds

### InTech Europe

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.