IDIAP RESEARCH REPORT

A Novel Statistical Generative Model Dedicated To Face Recognition

Guillaume Heusch [a,b]   Sébastien Marcel [a]

IDIAP–RR 07-39

September 2007

[a] IDIAP Research Institute
[b] Ecole Polytechnique Fédérale de Lausanne (EPFL)

IDIAP Research Institute www.idiap.ch
Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 07-39

A Novel Statistical Generative Model Dedicated To Face Recognition

Guillaume Heusch Sébastien Marcel

September 2007

Abstract. In this paper, a novel statistical generative model to describe a face is presented, and is applied on the face authentication task. Classical generative models used so far in face recognition, such as Gaussian Mixture Models (GMM) and Hidden Markov Models (HMM) for instance, are making strong assumptions on the observations derived from a face image. Indeed, such models usually assume that local observations are independent, which is obviously not the case in a face. The presented model hence proposes to encode relationships between salient facial features by using a static Bayesian Network. Since robustness against imprecisely located faces is of great concern in a real-world scenario, authentication results are presented using automatically localised faces. Experiments conducted on the XM2VTS and the BANCA databases showed that the proposed approach is suitable for this task, since it reaches state-of-the-art results. We compare our model to baseline appearance-based systems (Eigenfaces and Fisherfaces) but also to classical generative models, namely GMM, HMM and pseudo-2DHMM.

1 Introduction

Face recognition has been and is still an active research area, probably because of its wide-range of applications, including video-surveillance, user authentication and human-computer interaction to name a few. Hence, many different algorithms have been proposed to solve this task over the last thirty years. Nowadays, various systems are able to properly recognise people based on their face image. However, such results are often attained only if a sufficient amount of training data covering a reasonable range of variations (such as pose or illumination conditions for instance) is available to train the recognition system, and provided that the face is perfectly located in the image.

A face recognition system can be used in two modes: authentication (or verification) and identification. An authentication system involves confirming or denying the identity claimed by an individual. On the other hand, an identification system attempts to establish the identity of a given person out of a pool of different people. Identification generally operates on a closed-set scenario (the individual to identify is present in the database), while authentication operates on an open-set scenario, where people not present in the database could try to fool the system. Although these tasks are slightly different, both modes usually share the same classification algorithms. In this work, the focus is made on the face authentication task.

Existing face recognition algorithms are often divided into two categories: appearance-based (also referred to as *holistic*) and feature-based, depending on the way the face image is processed. In appearance-based method, the whole face image is represented as a high-dimensional vector. Due to the curse of dimensionality, such vectors cannot be compared directly. Hence, holistic methods use dimensionality reduction techniques to resolve this problem and thus derive lower-dimensional vectors for subsequent classification. The most popular examples among such approaches are based on Principal Component Analysis (PCA) and on Linear Discriminant Analysis (LDA). In PCA-based systems, also known as Eigenfaces [1], high-dimensional vectors are projected onto the subspace defined by the leading eigenvectors of the data covariance matrix. LDA-based face recognition, also referred to as Fisherfaces [2], is a supervised method: the linear projection is based on Fisher's linear discriminant formula to find a subspace where vectors of the same class are close to each other, and at the same time far from the ones belonging to other classes. The PCA or LDA subspace representation is then used for classification using a simple metric, or more sophisticated machine learning techniques, such as Support Vector Machines for instance [3]. Other dimensionality reduction techniques were applied to the face recognition problem, including Independent Component Analysis (ICA) [4], as well as non-linear methods such as Locality Preserving Projections (also known as Laplacianfaces) [5], Kernel PCA [6] [7] and Generalised Discriminant Analysis (GDA), which is actually a kernelized version of LDA [8] [9]. Amongst all these systems, empirical evaluation showed that Kernel methods, and Kernel Fisherfaces in particular, are the best for the face recognition task [9]. However, all these subspace-based approaches usually require a large amount of training data, but also a proper alignment or warping of the faces to be classified. Indeed, experiments conducted using holistic methods with automatically localised faces (i.e. when face localisation is error-prone) showed a significant drop in performance [10] [11].

Feature-based approaches are typically using a set of local observations obtained from the face image to derive a model of an individual, which is subsequently used for recognition. One of the most representative systems in this family is probably the Elastic Bunch Graph Matching (EBGM) [12]. In this case, a face image is represented by a set of wavelets coefficients arranged in a graph, whose nodes corresponds to fiducial points (eyes, tip of the nose, corner of the mouth, etc.). During the recognition process, the lattice is allowed to be deformable so as to maximise the correlation between corresponding wavelet coefficients of the gallery and of the probe image. Others recent approaches are based on Local Binary Patterns (LBP) [13] [14], where the face is represented by a set of concatenated LBP histograms, each one being computed in a different block of pixels along the image. Recognition is then performed by measuring the similarity between histograms. Other successful feature-based approaches are based on statistical generative models, such as Gaussian Mixture Models (GMM) [15], Hidden Markov Models (HMM) [16] [17] [18], or its variant [10] [19] [20]. Such systems usually decompose the face image into blocks and then learn the distribution of the blocks using one of the previously mentioned models. As compared to holistic approaches, feature-based systems have several advantages: they are more robust to little variations in pose, illumination, occlusion, expression and localisation errors [10] [21] [22]. Moreover, and in contrast to appearance-based systems, feature-based approaches are able to incorporate more a priori knowledge on the object to recognise, by selecting which features to use and how to relate them to each other.

In this paper, we propose a new statistical generative model based on *static* Bayesian Networks and especially tailored to deal with the object to be considered, that is the human face. Actually, classical generative models and GMM in particular, make strong independence assumptions on the way that face image data are generated. Indeed, in the GMM framework as applied in [15] for instance, overlapping blocks are considered to be independent, which is obviously not the case in a face image. Consider the two eyes for instance: the block containing the right eye is certainly related to the block containing the left one. HMM-based approaches, as well as models based on *dynamic* Bayesian Networks [23] are able to introduce some kind of structure into the observations, and usually performs better than GMM. In these cases however, the structure only constrain the *ordering* of the observations (i.e. the nose has to be above the mouth for instance) but do not add *relationships* between observations themselves, since Hidden Markov modelling considers that observations are independent, provided that the state is known. Hence, the main assumption that drove us towards the proposed model is that salient facial features are related to each others, and hence should not be treated as if they were independent. Actually, this paradigm along with the use of Bayesian Networks has already been successfully applied in two face processing task: face detection [24] and facial expression recognition [25]. For the task of face authentication, preliminary experiments using the proposed approach and yielding encouraging results were presented in [26]. In this contribution, we present experiments on the XM2VTS [27] and BANCA [28] databases using *automatically* located faces. Indeed, since face localisation is the necessary first step to any other face analysis task, we believe that robustness to imperfectly located faces is worth investigating. A comparison of the proposed approach to other face authentication systems is made using exactly the same settings. Namely, our system is compared to two popular appearance-based method, Eigenfaces and Fisherfaces, and also to classical generative models such as GMM, HMM and pseudo-2DHMM as applied in [10].

The remaining of this paper is organised as follows. Section 2 briefly introduce the Bayesian Networks framework while Section 3 presents the proposed model, as well as the inference and the learning algorithms in more details. In Section 4, an overview of the face and the facial features localisation systems are outlined. The experimental framework and the databases are described in Section 5 whereas results are presented in Section 6. Finally, Section 7 conclude the paper and propose possible future research directions.

2 Bayesian Networks

In this section, we will briefly describe the framework used to build the statistical generative model to represent a face. Bayesian networks (also known as belief networks or probabilistic expert systems) provide an intuitive way to represent the joint probability distribution over a set of variables: random variables are represented as nodes in a directed acyclic graph, and links express *causality relationships* between these variables. More precisely, relationships between nodes are specified through local conditional probabilities. Note that the lack of arcs between two nodes then encode a conditional independence of the associated variables.

More generally, let us define $Pa(X_i)$ as the set of parents of the variable X_i in the directed acyclic graph, the joint probability encoded by a Bayesian Network over the set of variables $\mathbf{X} = (X_1, ..., X_n)$ is given by the following chain rule:

$$P(\mathbf{X}) = \prod_{i=1}^{n} P(X_i | Pa(X_i)) \tag{1}$$

Hence, a Bayesian Network is fully defined by the *structure* of the graph and by its *parameters*, which consists in the conditional probability distributions of each variable given its parents. Note however that a variable X_i may have no parents, in which case its probability distribution is simply given by $P(X_i)$.

An important task in Bayesian Networks is inference. It consists in computing probabilities of interest, once evidence has been entered into the network (i.e. when one or more variables has been observed). In other words, entering evidence consists in either fixing the state of a discrete variable to one of its possible value or to assign a value in the case of a continuous variable. We are then interested in finding the effect this evidence has on the distribution of the others unobserved (or hidden) variables.

There are many different algorithm allowing to perform inference, the most simple and intuitive one is certainly the so-called bucket elimination [29]. The idea is to sum over the values of the irrelevant variables (i.e. the one we are not interested in) by taking the network topology into account. This algorithm has the advantage to work with any networks, including multiply-connected graphs. However, it is inefficient to handle multiple queries, since it has to be run for every variable of interest. The most renowned algorithm to perform inference in singly-connected graphs is certainly belief propagation [30]. Here, messages are passed between all the nodes until convergence and thus multiple queries are answered in a more efficient way. Another more generic method to perform exact inference, and which is both able to deal with multiple queries and multiply-connected networks is the Junction Tree algorithm [31]. In our model, this latter algorithm is used and is hence explained in more details in the next section.

3 Proposed Model

The proposed model relies on two main assumptions. First, we believe that salient facial features such as the eyebrows, the eyes, the nose and the mouth provide enough informations to discriminate two individuals. Second, it is assumed that facial features are somehow correlated and thus should not be considered independently. The proposed model trying to capture relationships between facial features is depicted in Figure 1. Shaded nodes are representing visible observations derived from the face image, whereas white nodes are representing the hidden *causes* that generated these observations. The model can be explained as follow: the nodes on the top represent unknown relationships between eyebrows and eyes (node BE), eyes and nose (node EN) and nose and mouth (node NM). Hence, these variables are used to model the relationship between the different face parts. These combinations then generate a certain type of facial features (such as a small nose, or broad lips for instance), represented by the nodes at the second level. And finally, these types of facial features generate the corresponding observations. Note that our model does introduce relationships between observations: if the node O_{le} is observed, information about the node O_{re} can be inferred through the node E for instance.

In this network, hidden nodes are discrete-valued and observed nodes are multivariate gaussians. Hence, the probability distributions of the nodes on the first and second level are given by (conditional) probability tables, whereas the distributions of the nodes corresponding to observations are given by

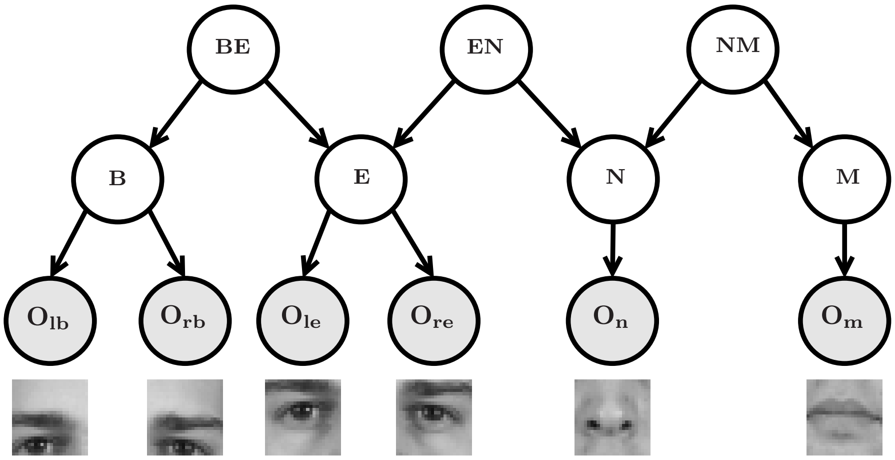


Figure 1: The proposed model: observed salient facial features are generated by a tree-structured Bayesian Network. Shaded nodes represent visible observations whereas white nodes denote hidden causes.

conditional gaussians, defined as:

$$P(O = \mathbf{o}|Pa(\mathbf{o}) = i) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma_i|^{\frac{1}{2}}} \exp\Big(- \frac{1}{2}(\mathbf{o} - \mu_i)^T \Sigma_i^{-1}(\mathbf{o} - \mu_i)\Big) \tag{2}$$

where $O = \mathbf{o}$ stands for a realisation of one of the observations and $Pa(\mathbf{o}) = i$ for a possible configuration of its parent. n is the dimension of the feature vector representing a particular observation. The mean μ_i and the covariance matrix Σ_i are the parameters of the conditional gaussian distribution and depend on the value of the parent node. Note also that in our model, diagonal covariances matrices are used. The parameters of the Bayesian Network to be learned are denoted by θ and consists in the entries of the (conditional) probability tables as well as the means and the covariance matrices of the conditional gaussians.

Ultimately, we are interested in finding how well a model fit an observed face representation. This is achieved by computing the probability of the observations given the model, usually referred to as the likelihood. Defining the set of visible observations $\mathbf{v} = (O_{lb}, O_{rb}, O_{le}, O_{re}, O_n, O_m)$, the log-likelihood $\mathcal{L}(\theta, \mathbf{v})$ of a face representation is computed by first inferring the distribution of the hidden variables using the Junction Tree algorithm, and then by summing out over the states of the hidden variables.

3.1 Inference: The Junction Tree Algorithm

The Junction Tree Algorithm [31] [32] is an exact inference algorithm which basically consists in two steps. First the directed acyclic graph is transformed into a tree-structured secondary structure and becomes an undirected graphical model. Second, messages are exchanged between nodes in this undirected representation. The Junction Tree, depicted in Figure 2, is obtained thanks to three operations on the original graph: moralization, triangulation and finally junction tree construction. Nodes of the Junction Tree are cluster of variables and are usually called *cliques* (represented as ellipses), each link is labelled with a *separator* containing the variables present in the two linked cliques (represented as squares). Each clique (respectively separator) has an associated potential, which is a real-valued function on the configurations of the set of variables in the clique (resp. separator).

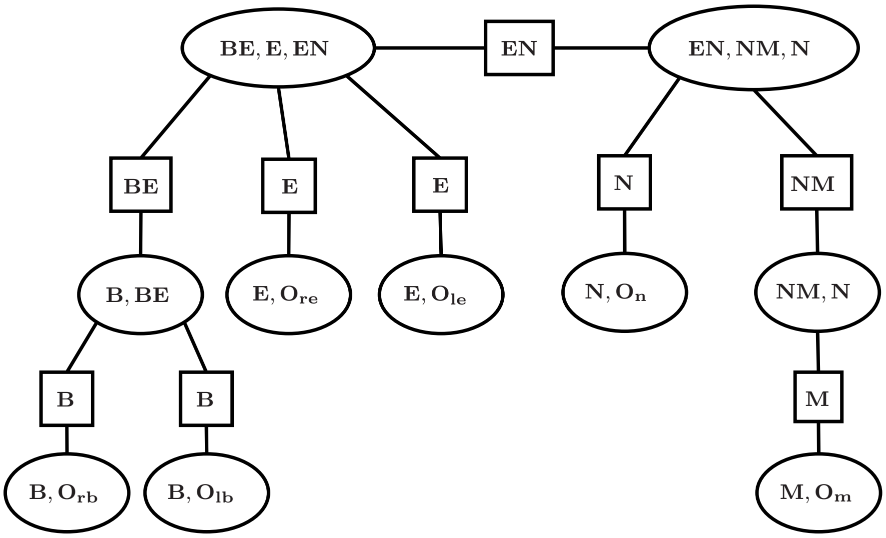


Figure 2: Junction Tree corresponding to the network in Figure 1.

Once the Junction Tree has been built and when observations are entered, clique and separator potentials are initialised such that the distribution defined by the Junction Tree matches the original distribution encoded by the Bayesian Network. Note that in our case, gaussian observations are always observed and hence all potentials are defined as tables. Then, messages between cliques are exchanged through separators in the form of potentials operations.

In order to derive a consistent message-passing algorithm along the tree, an arbitrary node is defined as the root. Messages are then forwarded from the leaves to the root. Once the root has collected all incoming messages, it sends messages back to neighbouring nodes, and so on until the leaves are reached. Hence, there are two messages that are passed along every link. At the end of the message-passing algorithm, the different clique (respectively separator) potentials contains the joint probability of the variables present in the clique (resp. separator).

3.2 Learning

Learning in Bayesian Networks refers either to structure learning, parameters learning or both [33]. In our case, we are considering networks of fixed structure, and hence are interested in learning parameters from data by maximising the likelihood. Since hidden variables are present in the proposed model, the log-likelihood of the data cannot be decomposed according to the network topology (it would have been the case if every variable would have been observed). In our case, the log-likelihood is given by:

$$\mathcal{L}(\theta, \mathbf{v}) = \log \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}|\theta) \tag{3}$$

where θ denotes the parameters of the model, $\mathbf{v}$ represents the set of variables corresponding to visible observations and $\mathbf{h}$ is the set of hidden variables. Since maximising directly Equation (3) may be difficult, we simplify the problem using the variational approximation to the Expectation-

Maximisation (EM) algorithm [34]:

$$
\begin{aligned}
\mathcal{L}(\theta, \mathbf{v}) &= \log \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}|\theta) \\
&= \log \sum_{\mathbf{h}} q(\mathbf{h}) \frac{p(\mathbf{v}, \mathbf{h}|\theta)}{q(\mathbf{h})} \\
&\geq \sum_{\mathbf{h}} q(\mathbf{h}) \log \frac{p(\mathbf{v}, \mathbf{h}|\theta)}{q(\mathbf{h})} \\
&= \sum_{\mathbf{h}} q(\mathbf{h}) \log p(\mathbf{v}, \mathbf{h}|\theta) - \sum_{\mathbf{h}} q(\mathbf{h}) \log q(\mathbf{h}) \qquad (4)
\end{aligned}
$$

where $q(\mathbf{h})$ is the variational parameter and is a distribution over the hidden variables. Furthermore, it can be shown [34] that the optimal setting (i.e. when the bound corresponds to equality) for the variational distribution $q(\mathbf{h})$ is nothing else but $p(\mathbf{h}|\mathbf{v}, \theta)$. Moreover, and since the second term in Equation (4) can be neglected (since it does not depend on θ), this formulation is then equivalent to the classical EM algorithm [35]. Note that now, the first term in Equation (4) can be decomposed according to the network topology. The maximisation can thus be done independently for each local conditional distribution.

Hence, starting with initial parameters θ_0, the iteration t of the EM algorithm is performed by first inferring the distribution of the hidden variables given the data and the current settings of the parameter θ_t using the Junction Tree algorithm (E-step), and then by finding the parameters that maximise the log-likelihood as defined by the first term in Equation (4) (M-step).

$$
\begin{aligned}
&\textbf{E-step:} && \text{compute} && p(\mathbf{h}|\mathbf{v}, \theta_t) && (5) \\
&\textbf{M-step:} && \theta_{(t+1)} \leftarrow && \operatorname*{argmax}_{\theta} \sum_{\mathbf{h}} p(\mathbf{h}|\mathbf{v}, \theta_t) \log p(\mathbf{v}, \mathbf{h}|\theta) && (6)
\end{aligned}
$$

3.3 Model Adaptation

In the context of face recognition, it is often the case that few training examples per class are available and hence the Maximum Likelihood (ML) estimates of the parameters may be inaccurate [36]. One way to circumvent the lack of client-specific training data is to estimate the ML parameters of a nearby distribution using a larger amount of training data coming from different identities and then to *adapt* this distribution using training data of each individual. This idea was first used in speaker verification [36] [37] and was also successfully applied in face authentication [10]. Although this technique is often referred to as Maximum A Posteriori (MAP) learning, one should be aware that, in this context, it is not MAP learning in the strict Bayesian sense, since no priors $p(\theta)$ are explicitly set on the parameters. Rather, the nearby distribution, referred to as the so-called *world model*, is learned using the EM algorithm with the ML criterion as formulated above. Then, the parameters of each client model are adapted from the parameters of the world model using client-specific training data in the following way:

$$\theta_{client} = \alpha \cdot \theta_{ML} + (1 - \alpha) \cdot \theta_{world} \qquad (7)$$

where θ_{ML} denotes the client parameters obtained from a Maximum Likelihood estimation using client-specific data. The adaptation parameter $\alpha \in [0; 1]$ is used to weight the relative importance of the obtained ML statistics with respect to the prior knowledge we have on the distribution, represented by the parameters of the world model.

4 Facial Features Localisation

Face recognition results in the literature are usually presented assuming *perfect* localisation of the faces, often relying on manually annotated eyes position for instance. However, in a real-world scenario, faces must be automatically detected to be further processed. Furthermore, it has been shown that performances of most of existing algorithms decreases when errors are introduced in the localisation process [10] [11]. For these reasons, we believe that the behaviour of the proposed system is worth investigating using *automatically* detected faces. Hence, we briefly present the face detection algorithm used to locate the face in the input image. We also outline the Active Shape Model [38], as this algorithm was employed to localise the salient facial features used as observations in the proposed model (see Figure 1).

4.1 Face Detection

In order to detect the face in the input image, a variant of the face detector proposed by Fröba and Ernst [39] is used. The detector employs local features based on the Modified Census Transform (MCT), which represent each location of the image by a binary pattern computed from a 3x3 pixel neighbourhood. Each input image is scanned and all possible windows in a given scale range are analysed. Each window is then classified as containing a face or not. The classification is carried out using a cascade classifier in a similar way than in [40]. Overlapping windows labelled as faces are then merged together so as to provide a unique bounding box containing the detected face. Eyes position are then inferred from the position and the scale of the bounding box. Note that if a face is missed by the detector, eyes position are estimated from other images of the same individual within the same recording session, but where the face was effectively detected.

4.2 Active Shape Model

Active Shape Models (ASM) were first introduced by Cootes et al. in [38] and consists in fitting the shape of an object (in our case, a face), using a previously learned global shape model, usually represented as a set of landmark points (see Figure 3). In order to find the shape of the object in the input image, an iterative search is applied, starting from a rough approximation of the localisation of the object (i.e. eyes location inferred from face detection). During the matching process, each point of the shape moves in the image plane to achieve the best match between the image and the model of local observations trained with the global shape model. In our work, Local Binary Patterns (LBP) are used to model the local observations, as described in [41]. Note also that, as in the original ASM, constraints are added to the displacement of each point, such that the shape of the object does not diverge.

5 Experiments

In this section, we first describe the general framework to perform face authentication using statistical generative models. Then, we present measures used to assess the performance of the systems, as well as the databases and their respective experimental protocols. Finally, the feature extraction scheme for the proposed model is described.

5.1 General Framework

In the framework of face authentication, a client claims its identity and supports the claim by providing an image of its face to the system. There are then two different possibilities: either the client is

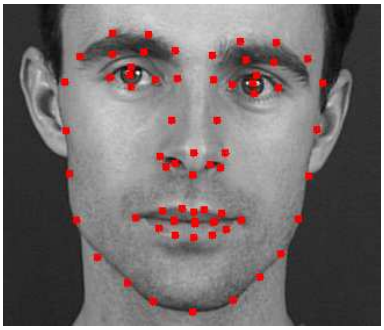


Figure 3: Landmark points of the Active Shape Model.

claiming its real identity, in which case it is referred to as a *true client*, either the client is trying to fool the system, and is referred to as an *impostor*. In this open-set scenario, subjects to be authenticated may or may not be present in the database. Therefore, the authentication system is required to give an opinion on whether the claimant is the true client or an impostor. Since modelling all possible impostors is obviously not feasible, the world-model (see Section 3.2) is used to simulate impostors, since it is trained using data coming from different identities and thus represents the model for an "average", or general individual [37].

More formally, let us denote θ_{world} as the parameter set defining the world-model whereas θ_{client} represents the client-specific parameters. Given a client claim and its face representation $\mathbf{v}$, an opinion on the claim is given by the following log-likelihood ratio:

$$\Lambda(\mathbf{v}) = \log p(\mathbf{v}|\theta_{client}) - \log p(\mathbf{v}|\theta_{world}) \tag{8}$$

where $p(\mathbf{v}|\theta_{client})$ is the likelihood of the claim coming from the true client and $p(\mathbf{v}|\theta_{world})$ is an approximation of the likelihood of the claim coming from an impostor. Based on a threshold τ, the claim is accepted if $\Lambda(\mathbf{v}) \geq \tau$ and rejected otherwise.

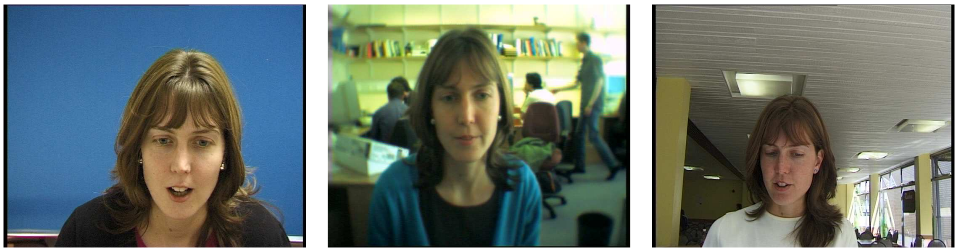
5.2 Performance Measures

Face authentication is thus subject to two type of errors, either the true client is rejected (false rejection) or an impostor is accepted (false acceptance). In order to measure the performance of authentication systems, we use the Half Total Error Rate (HTER), which combines the False Rejection Rate (FRR) and the False Acceptance Rate (FAR) and is defined as:

$$HTER(\tau, \mathcal{D}) = \frac{FAR(\tau, \mathcal{D}) + FRR(\tau, \mathcal{D})}{2} \quad [\%] \tag{9}$$

where $\mathcal{D}$ denotes the used dataset. Since both the FAR and the FRR depends on the threshold τ, they are strongly related to each other: increasing the FAR will reduce the FRR and vice-versa. For this reason, authentication results are often presented using either Receiver Operating Characteristic (ROC) or Detection-Error Tradeoff (DET) curves, which basically plots the FAR versus the FRR for different values of the threshold. Another widely used measure to summarise the performance of a system is the Equal Error Rate (EER), defined as the point along the ROC or DET curve where the FAR equals the FRR.

However, it was noted in [42] that ROC and DET curves may be misleading when comparing models. Hence, the so-called Expected Performance Curve (EPC) was proposed, and consists in an unbiased estimate of the reachable performance of a model at various operating points [42]. Indeed, in real-world scenario, the threshold τ has to be set a priori: this is typically done using a validation



(a) controlled (b) degraded (c) adverse

Figure 4: Example of the different scenarios in the BANCA database

(or development) set. Nevertheless, the optimal threshold can be different depending on the relative importance given to the FAR and the FRR. Hence, in the EPC framework, $\beta \in [0; 1]$ is defined as the tradeoff between FAR and FRR. The optimal threshold τ^* is then computed using different values of β, corresponding to different operating points:

$$\tau^* = \underset{\tau}{\operatorname{argmin}} \quad \beta \cdot \mathrm{FAR}(\tau, \mathcal{D}_v) + (1-\beta) \cdot \mathrm{FRR}(\tau, \mathcal{D}_v) \tag{10}$$

where $\mathcal{D}_v$ denotes the validation set. Performance for different values of β is then computed on the test set $\mathcal{D}_t$ using the previously found threshold. Note that setting β to 0.5 yields the Half Total Error Rate (HTER) as defined in Equation (9). Moreover, a modified version of the standard proportion test, as described in [43] is used in order to compute 95% confidence intervals around Expected Performance Curves (Figure 8).

5.3 Databases

The XM2VTS database [27] is a multi-modal database containing 295 identities, among which 200 are used as true clients (the remainder are considered as impostors). Recordings were acquired during four sessions over a period of five months under controlled conditions (blue background, uniform illumination). Each session contains two pictures of each individual. Along with the database, two experimental protocols, stating which images are used for training, validation and testing have been defined. Experiments presented in this paper uses the version 1 of the Lausanne Protocol (denoted as LP1).

The BANCA database [28] was especially meant for multi-modal biometric authentication and contains 52 clients (English corpus), equally divided into two groups $g1$ and $g2$ used for validation and testing respectively. The corpus is extended with an additional set of 30 other subjects used to train the world model. Image acquisition was performed with two different cameras: a cheap analogue webcam, and a high-quality digital camera, under several realistic scenarios: controlled (high-quality camera, uniform background, controlled lighting), degraded (webcam, non-uniform background) and adverse (high-quality camera, arbitrary conditions). Figure 4 shows examples of the different acquisition scenarios.

In the BANCA protocol, seven distinct configurations for the training and testing policy have been defined. In our experiments, the configurations referred to as Match Controlled (Mc), Unmatched Adverse (Ua), Unmatched Degraded (Ud) and Pooled Test (P) are used. All of the listed configurations use the same training conditions: each client is trained using images from the first recording session, which corresponds to the controlled scenario. Testing is then performed on images taken from the controlled scenario (Mc), adverse scenario (Ua), degraded scenario (Ud), while (P) does the test for each of the previously described configurations.

Figure 5: Illustration of cropped faces using manually located eyes (first row) and automatically located eyes (second row). Note the variations in scale between column 2 and 4 for instance.

5.4 Feature Extraction

First, faces are automatically located using the face detector described in section 4. The face detector has been trained using face images coming from the following databases: CMU, BioId, AR and Purdue. Hence, no prior knowledge on the face images used in the authentication experiments were introduced in the detection process. However, the ASM was trained on the training set of the XM2VTS database (protocol LP1), since in this case, the 68 annotations representing the groundtruth for the landmarks were available.

Feature extraction for the proposed model is performed by running the ASM on the input image, using the automatically detected eyes location as the starting point. Based on the resulting facial features locations, blocks of pixels are extracted around selected salient features (see Figure 1). In order to account for imprecisely located features, and also to increase the amount of training data, shifted blocks of a variable amount of pixels in each direction are also extracted. Note that a similar approach was already used in [22]. In order to mitigate the influence induced by variations in illumination conditions, each block is pre-processed by the LBP-based pre-processing proposed in [44]. Finally, each block is decomposed in terms of 2D Discrete Cosine Transform (2D-DCT) in order to build the final observation vectors.

Hyperparameters for the proposed model, such as the size of extracted blocks, the number of pixels for the shifted blocks, the dimension of the DCT feature vectors, the cardinality of the hidden nodes, as well as the adaptation parameter α were selected in order to minimise the Equal Error Rate (EER) on the validation set $\mathcal{D}_v$.

Regarding the other approaches, faces were first cropped from the original images, resized to 64x80 pixels, converted to grayscale and pre-processed with the same technique used for the blocks [44]. Note that the cropping step is performed using automatically detected eyes location (resulting from face detection), and hence may result in different scales and translations of face images, as illustrated on Figure 5. Features for the classical generative models, namely GMM, HMM and pseudo-2DHMM were extracted using the feature extraction scheme described in [10].

6 Results and Discussion

In this section, face authentication results using automatically located faces are presented. Hereafter, the proposed model is referred to as BNFace, and for comparison purpose, we also report experimental

Table 1: HTER Performance on the test set of XM2VTS LP1 with automatic registration.

System	LP1 [%]
Eigenfaces	27.29
Fisherfaces	28.19
GMM	12.61
HMM	13.64
P2D-HMM	**2.56**
BNFace	5.53

Table 2: HTER Performance on the test set (g2) of BANCA with automatic registration.

System	Mc [%]	Ua [%]	Ud [%]	P [%]
Eigenfaces	18.85	32.18	30.03	26.49
Fisherfaces	21.38	31.67	32.08	29.27
GMM	7.33	34.76	33.95	28.83
HMM	8.01	21.67	21.54	16.84
P2D-HMM	**2.40**	**13.49**	15.29	**12.61**
BNFace	3.85	19.94	**13.56**	12.70

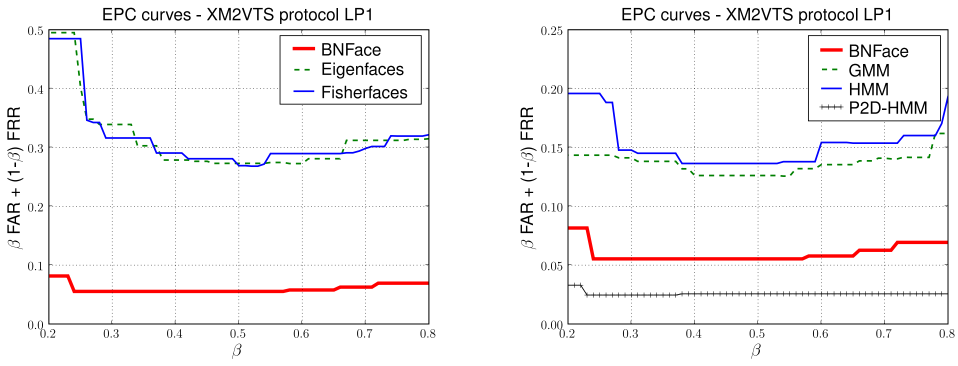
results obtained with classical generative models: GMM, HMM and pseudo-2DHMM as applied in [10], as well as two popular baseline appearance-based systems, Eigenfaces and Fisherfaces. Note that the same experimental settings (i.e. automatically detected faces) were used for each system.

6.1 Experimental Setup and Results

Presented results for the proposed model were obtained using extracted blocks of 24x24 pixels. So, for each facial feature, blocks centered on the corresponding landmark point given by the ASM are extracted. Besides, for each block, additional blocks with shifts of 2, 4 and 6 pixels in each direction are also extracted. Hence, for a single observation, we obtained 25 blocks. The first 64 coefficients were retained from the 2D-DCT on the blocks, thus resulting in final feature vectors of dimension $n = 64$. The cardinality of the hidden nodes were set to 3 at the top level, and to 8 at the second level. Finally, the adaptation parameter α was set to 0.4.

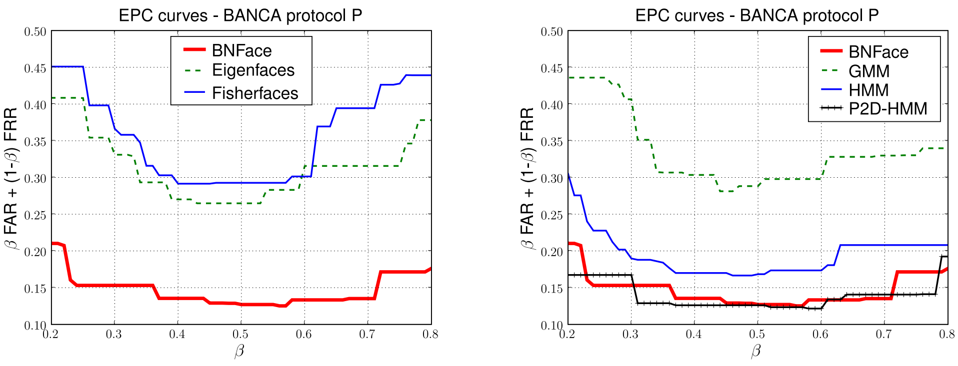
Note also that presented results were obtained following the *strict* usage of the protocols defined with each database. Hence, for the XM2VTS database, we use 600 images corresponding to all client training data, in order to train the world models, but also to build PCA and LDA matrices. For the BANCA database, the additional set containing 10 images of 30 individuals were used for the same purposes. In particular, we do not use any other corpus or database, nor mirroring the available images to build either world models or subspace representations, as it was sometimes done in other studies [10] [45]. Doing this way enables a fair comparison among the different systems, since exactly the same data and protocols were used for each tested system. Table 1 reports HTER performance obtained on the XM2VTS database for protocol LP1 and Table 2 reports HTER performance on the BANCA database for protocols Mc, Ua, Ud and P.

We also present EPCs of the different systems (Figure 6 and 7). For the sake of clarity, curves comparing the proposed system against holistic approaches are plotted on the left-hand side of the



(a) BNFace and Holistic Systems (b) BNFace and Generative Models

Figure 6: EPC curves for the test set of the XM2VTS database.



(a) BNFace and Holistic Systems (b) BNFace and Generative Models

Figure 7: EPC curves for the test set of the BANCA database, protocol P.

figures and curves comparing the proposed system to other generative models are plotted on the right-hand side. Note that only the protocol P was used for the curves on the BANCA database, since it can be viewed as a summary of the different investigated protocols (Mc, Ua, Ud).

6.2 Discussion

Compared to the popular holistic systems (Eigenfaces and Fisherfaces), the proposed system performs consistently better on both databases. Moreover, Figures 6(a) and 7(a) representing EPC curves for BNFace and both holistic systems show that the authentication error is drastically reduced at all operating points when the proposed system is used. These results are not surprising, since it has been previously shown that the performance of appearance-based systems is severely affected when faces are not perfectly aligned. Hence, conducted experiments confirms that local feature-based systems are more robust to imperfectly located faces.

Since classical generative models also uses local features to perform classification, such systems are usually less affected by the face localisation step. Indeed, they perform generally better than the holistic ones (Tables 1 and 2). Hence, a comparison of the proposed models with GMM, HMM and pseudo-2DHMM may reveal the advantages and the drawbacks at the models level, when applied to face authentication.

It must be noted that BNFace performs way better than the simple GMM-based system on both databases. This result is particularly interesting since it tends to support two stated hypothesis. First, only a subset of the face image, corresponding to salient facial features, is sufficient to perform authentication. Indeed, in the GMM framework, blocks of pixels are extracted from the whole face image. Second, it also suggests that blocks extracted from the face image are correlated and hence should not be treated as if they were independent. However, results obtained with GMM are surprising since they are much worse than previously published results on the same databases, also using automatic registration [15] [10]. Nevertheless, this fact can be explained thanks to three observations. First, the preprocessing step used in our work is different. Second, previous work used mirroring and additional data to train the models and third, we did not fine-tune the various hyperparameters, rather, we used the ones reported in [15] and [10].

In our experiments, the HMM-based system outperforms the GMM-based system on the BANCA database (note that this is the converse on the XM2VTS database). This result is in contrast to the one obtained in [10], where GMM were shown to perform better than HMM in the case of automatic face localisation. Hence, it is difficult to say whether the model itself is not appropriate to model the face or if its performance is affected by localisation errors. Nevertheless and according to [10], HMM seems to better model the face image, since it performs better than GMM when the face is manually located (similar results were also observed when reproducing this experiment). On one hand, this suggest that introducing structure to the observations, in the form of vertical spatial relationships may be useful. But on the other hand, HMM also add rigid horizontal constraints, and this may explain the contradictory results obtained with this approach. However, note that the proposed model still outperforms the HMM-based system on both databases. Hence, it suggests once again that relationships between facial features themselves, and not only on their ordering, is useful to describe a face.

The pseudo-2DHMM is the only system performing better than the proposed system. It can be mainly explained thanks to two observations: first, rigid constraints are less important than in the HMM for instance, hence pseudo-2DHMM is less affected by automatic face detection. Second, the model is able to add two-dimensional spatial constraints to the observations. Results obtained with this approach hence suggest that two-dimensional spatial relationships along the *entire* face image are important. Note that results on the Unmatched degraded (Ud) protocol of the BANCA database, the proposed model performs better than P2D-HMM. This suggest that using only a subset of the face image less affects the authentication system in the case of a strong mismatch between training and testing acquisition conditions. However, results obtained with BNFace and with P2D-HMM are close to each other, especially on the protocol P of the BANCA database. Hence, in order to better compare these two classifiers, we present the Expected Performance Curves of the two systems together with the 95% confidence interval (Figure 8). One can see that in some parts, an overlap is occurring, hence showing that the two classifier are not statistically different. The bottom part of the Figure depicts the statistical difference between the two classifiers. If the curve is above 0.95, this means that the classifiers are different with 95% confidence. As can be seen on Figure 8, the two classifiers are only statistically different with high confidence in a small range of operating points.

Although the pseudo-2DHMM approach obtained the best results, it is also the most complex. Indeed, it uses much more client-specific parameters to describe a face and is also more computationally demanding than the proposed system. In Table 3, we report the computational time to perform the three tasks involved in face authentication: world-model training, client-model adaptation (computed on average over the client) for one individual, and authentication time for one individual (also computed on average). We also report the number of client-specific parameters for the proposed system and P2D-HMM. These quantities were obtained using the BANCA database on a computer with an AMD Athlon 2,6 GHz.

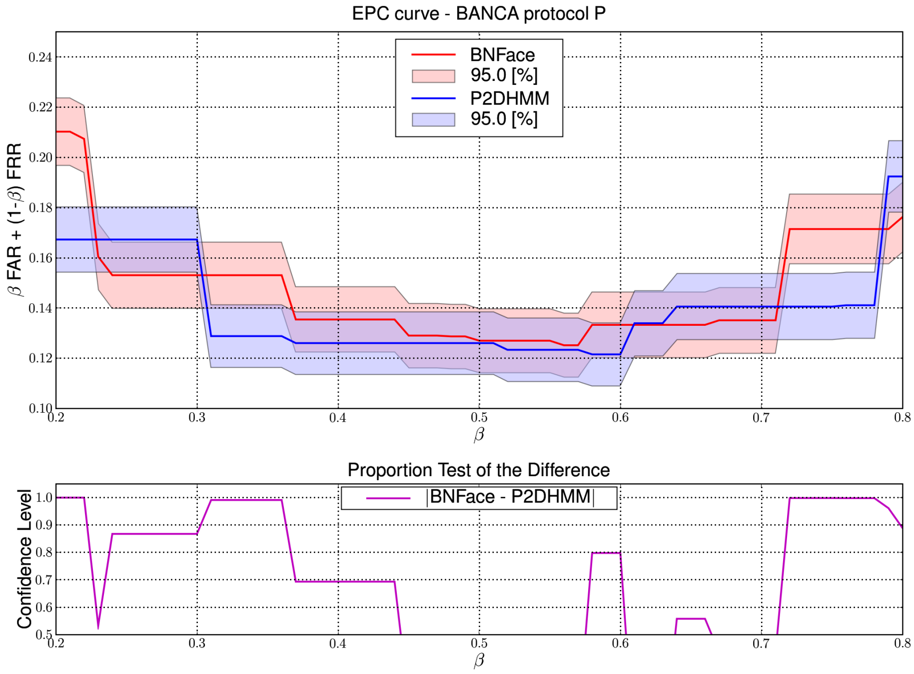


Figure 8: EPC with confidence intervals and statistical difference for BNFace and P2D-HMM on the protocol P of the BANCA database.

Table 3: Computational time on BANCA

System	world model training time	client model adaptation time	individual authentication time	number of parameters
P2D-HMM	3520 sec	$\sim$ 220.2 sec	$\sim$ 9.8 sec	73'726 [10]
BNFace	1499 sec	$\sim$ 50.2 sec	$\sim$ 2 sec	6345

Overall, obtained results suggest that the proposed model based on Bayesian Networks is suitable for the task of face authentication using automatically localised faces. Indeed, we conducted comparative experiments and the proposed model yields better performance than 4 out of 5 baseline systems. Moreover, obtained results are competitive with state-of-the-art reported in the literature on the same databases and with automatic registration [45] [46]. Note however that the proposed system strongly relies on the Active Shape Model to locate the salient facial features. Indeed, upon visual inspection of the landmarks, we remarked that, in some cases, facial features are not accurately located. Hence, our model may also suffer from such imprecision.

7 Conclusion and Future Directions

In this paper, we introduced a novel statistical generative model based on Bayesian Networks and especially tailored to deal with the object to be processed, that is two-dimensional face images. The proposed model relies on two main assumptions: first, salient facial features such as eyebrows, eyes, nose and mouth contains sufficient information to discriminate two individuals. Second, such local observations should not be treated independently. Rather, it was assumed that salient facial features are related to each others. The proposed approach was applied to the task of face authentication using automatically detected faces. Hence, the whole authentication process is made automatic, which is a desirable behaviour in a real-world scenario. Two benchmark databases were used to assess the performance of the system and show convincing results. Indeed, the proposed model outperforms classical appearance-based methods, but also classical generative models, where independence is assumed between local observations. Besides, presented results are competitive with state-of-the-art reported in the literature on the same databases. However, more complex models such as pseudo-2DHMM still performs better, although demanding more computational resources.

This work is, to the best of our knowledge, the first attempt to use *static* Bayesian Networks to tackle the face authentication problem and future research directions are manifold. Actually, we do not know which kind of information is useful to uniquely describe a face. In this work, we chose to use salient facial features as a set of observations, but other clues such as texture, colour or even shape certainly carry discriminative information. Another open issue is how to relate local observations to each others. Indeed, the structure of the network was designed according to our prior knowledge on how facial features may be related. However, we still do not know if there are actually causal relationships between features, and how these can be expressed. Nevertheless, we think that using static Bayesian Networks provide an elegant framework to describe faces, and is worth investigating.

8 Acknowledgements

This work was supported by the GMFace project of the Swiss National Science Foundation (SNSF) and by the Swiss National Centre of Competence in Research (NCCR) on "Interactive Multimodal Information Management (IM2)[1]. Softwares were implemented using the TorchVision library[2] and experiments were carried out using the PyVerif biometric verification toolkit[3].

References

[1] M. Turk and A. Pentland. Face Recognition Using Eigenfaces. In *IEEE Intl Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 586–591, 1991.

[1] `http://www.im2.ch`
[2] `http://torch3vision.idiap.ch`
[3] `http://pyverif.idiap.ch`

[2] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[3] P. J. Phillips. Support Vector Machines Applied to Face Recognition. In *Neural Information Processing Systems (NIPS)*, pages 803–809, 1999.

[4] M. Bartlett, J. Movellan, and T. Sejnowski. Face Recognition by Independent Component Analysis. *IEEE Trans. on Neural Networks*, 13(6):1450–1464, 2002.

[5] X. He, S. Yan, Y. Hu, N. Partha, and H-J. Zhang. Face Recognition using Laplacianfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(3):328–340, 2005.

[6] M-H. Yang, N. Ahuja, and D. Kriegman. Face Recognition Using Kernel Eigenfaces. In *IEEE Intl Conf. on Image Processing (ICIP)*, volume 1, pages 37–40, 2000.

[7] K-I. Kim, K. Jung, and H-J. Kim. Face Recognition Using Kernel Principal Component Analysis. *IEEE Signal Processing Letters*, 9(2):40–42, 2002.

[8] L. Shen, L. Bai, and M. Fairhust. Gabor Wavelets and General Discriminant Analysis for Face Identification and Verification. *Image and Vision Computing*, 25(5):553–563, 2007.

[9] M-H. Yang. Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition using Kernel Methods. In *IEEE Intl Conf. on Automatic Face and Gesture Recognition (AFGR)*, pages 205–211, 2002.

[10] F. Cardinaux, C. Sanderson, and S. Bengio. User Authentication via Adapted Statistical Models of Face Images. *IEEE Trans. on Signal Processing*, 54(1):361–373, 2005.

[11] Y. Rodriguez, F. Cardinaux, S. Bengio, and J. Mariéthoz. Measuring The Performance of Face Localization Systems. *Image and Vision Computing*, 24(8):882–893, 2006.

[12] L. Wiskott, J-M. Fellous, N. Krüger, and C. Von Der Malsburg. Face Recognition By Elastic Bunch Graph Matching. In *Intl Conf. on Computer Analysis of Images and Patterns (CAIP)*, pages 456–463, 1997.

[13] T. Ahonen, A. Hadid, and M. Pietikäinen. Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, December 2006.

[14] Y. Rodriguez and S. Marcel. Face Authentication Using Adapted Local Binary Pattern Histograms. In *European Conference on Computer Vision (ECCV)*, pages 321–332, 2006.

[15] F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of MLP and GMM classifiers for face verification on XM2VTS. In *Intl Conf. on Audio- and Video-based Biometric Person Authentication (AVBPA)*. Springer, 2003.

[16] F. Samaria and S. Young. HMM-based Architecture for Face Identification. *Image and Vision Computing*, 12(8):537–543, October 1994.

[17] A. Nefian and M. Hayes. Hidden Markov Models for Face Recognition. In *IEEE Intl Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 2721–2724, 1998.

[18] A. Martinez. Face Image Retrieval Using HMMs. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 35–39, 1999.

[19] S. Eickeler, S. Müller, and G. Rigoll. Recognition of JPEG Compressed Face Images Based on Statistical Methods. *Image and Vision Computing*, 18(4):279–287, 2000.

[20] A. Nefian and M. Hayes. Maximum Likelihood Training of the Embedded HMM for Face Detection and Recognition. In *IEEE Intl Conf. on Image Processing (ICIP)*, volume 1, pages 33–36, 2000.

[21] B. Heisele, P. Ho, J. Wu, and T. Poggio. Face Recognition: Component-based Versus Global Approaches. *Computer Vision and Image Understanding*, 91(1):6–21, 2003.

[22] A. Martinez. Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(6):748–763, 2002.

[23] A. Nefian. Embedded Bayesian Networks for Face Recognition. In *IEEE Intl Conf. on Multimedia and Expo (ICME)*, volume 2, pages 133–136, 2002.

[24] K. Yow and R. Cipolla. Feature-based Human Face Detection. *Image and Vision Computing*, 15(9):713–735, 1997.

[25] I. Cohen, N. Sebe, A. Garg, M.S. Lew, and T.S. Huang. Facial Expression Recognition From Video Sequences. In *IEEE Intl Conf. on Multimedia and Expo (ICME)*, volume 2, pages 121 – 124, 2002.

[26] G. Heusch and S. Marcel. Face Authentication with Salient Facial Features and Static Bayesian Network. In *Intl Conf. on Biometrics (ICB)*, volume 4642 of *Lecture Notes in Computer Science*, pages 878–887. Springer, 2007.

[27] K. Messer, J. Matas, J. Kittler, J. Lüttin, and G. Maitre. XM2VTSDB: The Extended M2VTS Database. In *Intl Conf. Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 72–77, 1999.

[28] E. Bailly-Baillière et al. The Banca Database and Evaluation Protocol. In *Intl Conf. on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 625–638, 2003.

[29] R. Dechter. Bucket Elimination: a Unifying Framework for Probabilistic Inference. In *Uncertainty in Artificial Intelligence (UAI)*, pages 211–219, 1996.

[30] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[31] G. Cowell, P. Dawid, L. Lauritzen, and J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, 1999.

[32] C. Huang and A. Darwiche. Inference in Belief Networks: A Procedural Guide. *International Journal of Approximate Reasoning*, 15(3):225–263, 1996.

[33] D. Heckerman. *Learning in Graphical Models*, chapter A Tutorial on Learning With Bayesian Networks, pages 301–354. MIT Press, 1999.

[34] M.I. Jordan, Z. Ghahramani, T.S. Jaakkolla, and L.K. Saul. An Introduction to Variational Methods for Graphical Models. *Machine Learning*, 37:183–233, 1999.

[35] A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood From Incomplete Data via the EM Algorithm. *The Journal of Royal Statistical Society*, 39:1–37, 1977.

[36] J-L. Gauvain and C-H. Lee. Maximum A Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *IEEE Trans. on Speech and Audio Processing*, 2(2):291–298, 1994.

[37] D.A. Reynolds, T.F. Quateri, and R.B. Dunn. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10:19–41, 2000.

[38] T.F. Cootes, C.J. Taylor, D. Cooper, and J. Graham. Active Shape Models: Their Training and Applications. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[39] B. Fröba and A. Ernst. Face Detection With The Modified Census Transform. In *IEEE Intl Conf. on Automatic Face and Gesture Recognition (AFGR)*, pages 91–96, 2004.

[40] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In *IEEE Intl Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 511, 2001.

[41] J. Keomany and S. Marcel. Active Shape Models Using Local Binary Patterns. RR 06-07, IDIAP Research Institute, 2006.

[42] S. Bengio, J. Mariéthoz, and M. Keller. The Expected Performance Curve. In *Intl Conf. On Machine Learning (ICML)*, 2005.

[43] S. Bengio and J. Mariéthoz. A Statistical Significance Test for Person Authentication. In *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2004.

[44] G. Heusch, Y. Rodriguez, and S. Marcel. Local Binary Patterns as an Image Preprocessing for Face Authentication. In *IEEE Intl Conf. on Automatic Face and Gesture Recognition (AFGR)*, pages 9–14, 2006.

[45] K. Messer et al. Face Authentication Test on the BANCA Database. In *Intl Conf. on Pattern Recognition (ICPR)*, volume 4, pages 523–532, 2004.

[46] K. Messer et al. Face Authentication Competition on the BANCA Database. In *Intl Conf. on Biometric Authentication (ICBA)*, volume 3072 of *Lecture Notes in Computer Science*, pages 8–15. Springer, 2004.