Available online via: http://dx.doi.org/10.1016/j.patcog.2005.07.001

On Transforming Statistical Models for Non-Frontal Face Verification

Conrad Sanderson [*] [a] [b]

and

Samy Bengio [c]

and

Yongsheng Gao [d]

Version: 20 Aug 2005

[*] e-mail: conradsand @ ieee.org, phone: +61 2 6125 8812, fax: +61 2 6125 8645
[a] National ICT Australia (NICTA), Locked Bag 8001, Canberra, ACT 2601, Australia
[b] Australian National University, Canberra, ACT 0200, Australia
[c] IDIAP Research Institute, Rue du Simplon 4, CH-1920 Martigny, Switzerland
[d] School of Microelectronic Engineering, Griffith University, QLD 4111, Australia

Abstract: We address the pose mismatch problem which can occur in face verification systems that have only a single (frontal) face image available for training. In the framework of a Bayesian classifier based on mixtures of gaussians, the problem is tackled through extending each frontal face model with artificially synthesized models for non-frontal views. The synthesis methods are based on several implementations of Maximum Likelihood Linear Regression (MLLR), as well as standard multi-variate linear regression (LinReg). All synthesis techniques rely on prior information and learn how face models for the frontal view are related to face models for non-frontal views. The synthesis and extension approach is evaluated by applying it to two face verification systems: a holistic system (based on PCA-derived features) and a local feature system (based on DCT-derived features). Experiments on the FERET database suggest that for the holistic system, the LinReg based technique is more suited than the MLLR based techniques; for the local feature system, the results show that synthesis via a new MLLR implementation obtains better performance than synthesis based on traditional MLLR. The results further suggest that extending frontal models considerably reduces errors. It is also shown that the local feature system is less affected by view changes than the holistic system; this can be attributed to the parts based representation of the face, and, due to the classifier based on mixtures of gaussians, the lack of constraints on spatial relations between the face parts, allowing for deformations and movements of face areas.

Keywords: biometrics, pose mismatch, face recognition, local features, gaussian mixture model, prior information, model synthesis.

1. INTRODUCTION

Biometric recognition systems based on face images (here we mean both identification and verification systems) have attracted much research interest for quite some time. Applications include surveillance, forensics, transaction authentication, and various forms of access control, such as immigration checkpoints and access to digital information [1, 2, 3, 4].

Contemporary approaches are able to achieve low error rates when dealing with *frontal* faces (see for example [5, 6]). In order to handle *non-frontal* faces, previously proposed extensions to 2D approaches include the use of training images (for the person to be recognized) at multiple views [7, 8, 9]. In some applications, such as surveillance, there may be only one reference image (e.g., a passport photograph) for the person to be spotted. In a surveillance video (e.g. at an airport), the pose of the face is usually uncontrolled, thus causing a problem in the form of a mismatch between the training and the test poses.

While it is possible to use 3D approaches to address the single training pose problem [10, 11], in this paper we concentrate on extending two 2D based techniques. We extend a local feature approach (based on DCT-derived features [12, 13]) and a holistic approach (based on PCA-derived features [14, 15]). In both cases we employ a Bayesian classifier based on Gaussian Mixture Models (GMMs) [16, 17], which is central to our extensions.

The PCA/GMM system is an extreme example of a holistic system where the spatial relations between face characteristics (such as the eyes and nose) are rigidly kept. Contrarily, the DCT/GMM approach is an extreme example of a local feature approach (also known as a *parts based* approach [13]). Here, the spatial relations between face parts are largely not used, resulting in robustness to translations of the face which can be caused by an automatic face localization algorithm [18, 19]. In between the two extremes are systems based on multiple template matching [20], modular PCA [9], Pseudo 2D Hidden Markov Models [21, 22, 23] and approaches based on Elastic Graph Matching [24, 25]. As an in-depth review of face recognition literature is beyond the scope of this paper, the reader is directed to the following review articles [26, 27, 28, 29]. Further introductory and review material about the biometrics field in general can be found in [3, 30, 31, 32].

In general, an appearance based face recognition system can be thought of as being comprised of:

1. Face localization and segmentation
2. Feature extraction and classification

The first stage usually provides a size normalized face image (with eyes at fixed locations). Illumination normalization may also be performed (however, it may not be not necessary if the feature extraction method is robust to illumination changes). In this work we exclusively deal with the classification problem, and postulate that the face localization step has been performed correctly. Recent reviews of face localization algorithms can be found in [33, 34].

There are three distinct configurations of how a classifier can be used: the *closed set identification* task, the *open set identification* task, and the *verification* task[1] . In closed set identification, the job is to assign a given face into one of K face classes (where K is the number of known faces). In open set identification, the task is to assign a given face into one of $K + 1$ classes, where the extra class represents an "unknown" or "previously unseen" face. In the verification task the classifier must assign a given face

[1] *verification* is also known as *authentication*.

into one of two classes: either the face is the one we are looking for, or it isn't. The verification and open set identification tasks represent operation in an uncontrolled environment [35], where any face could be encountered. In contrast, the closed set identification task assumes that all the faces to be encountered are already known.

In this paper, we propose to address the single training pose problem by extending each statistical frontal face model with artificially *synthesized* models for non-frontal views. We propose to synthesize the non-frontal models via methods based on several implementations of Maximum Likelihood Linear Regression (MLLR), as well as standard multi-variate linear regression (LinReg). MLLR was originally developed for tuning speech recognition systems [36], and to our knowledge this is the first time it is being adapted for face verification.

In the proposed MLLR-based approach, prior information is used to construct *generic* face models for different views. A generic GMM does not represent a specific person's face - instead it represents a population of faces, or interpreted alternatively, a "generic" face. In the field of speech based identity verification, an analogous generic model is known as a world model and as a Universal Background Model [17, 37]. Each non-frontal generic model is constructed by *learning* and *applying* a MLLR-based transformation to the frontal generic model. When we wish to obtain a person's non-frontal model, we first obtain the person's frontal model via adapting [17] the frontal generic model; a non-frontal face model is then synthesized by applying the previously learned transformation to the person's frontal model. In order for the system to automatically handle the two views, a person's frontal model is extended by concatenating it with the newly synthesized model. The procedure is then repeated for other views. An interpretation of this procedure is shown in Fig. 1.

The LinReg approach is similar to the MLLR-based approach described above. The main difference is that it learns a common relation between two sets of feature vectors, instead of learning the transformation between generic models. In our case the LinReg technique is applicable only to the holistic system, while the MLLR-based methods are applicable to both holistic and local feature based systems.

Previous approaches to addressing single view problems include the synthesis of new *images* at previously unseen views; some examples are optical flow based methods [38, 39], and linear object classes [40]. To handle views for which there are no training images, an appearance based face recognition system could then utilize the synthesized images. The proposed model synthesis and extension approach is inherently more efficient, as the intermediary steps of image synthesis and feature extraction (from synthesized images) are omitted.

The model extension part of the proposed approach is somewhat similar to [8], where features from many real images were used to extend a person's face model. This is in contrast to the proposed approach, where the models are synthesized to represent the face of a person for various non-frontal views, *without* having access to the person's real images. The synthesis part is somewhat related to [41] where the "jets" in the nodes an elastic graph are transformed according to a geometric framework. Apart from the inherent differences in the structure of classifiers (i.e. Elastic Graph Matching compared to a Bayesian classifier), the proposed synthesis approach differs in that it is based on a statistical framework.

The rest of this paper is structured as follows. In Section 2 we briefly describe the database used in the experiments and the pre-processing of images. In Section 3 we overview the DCT and PCA based feature extraction techniques. Section 4 provides a concise description of the GMM based classifier and the different training strategies used when dealing with DCT and PCA derived features. In Section 5 we summarize MLLR, while in Section 6 we describe model synthesis techniques based on MLLR and standard multi-variate linear regression. Section 7 details the process of extending a frontal model with synthesized non-frontal models. Section 8 is devoted to experiments evaluating the proposed synthesis techniques and the use of extended models. Conclusions and future areas of research are given in Section 9.

2. DATABASE SETUP AND PRE-PROCESSING

In our experiments we utilized a subset of face images from the FERET database [42]. Specifically, we used images from the *ba*, *bb*, *bc*, *bd*, *be*, *bf*, *bg*, *bh* and *bi* portions, which represent views of 200 persons for approximately 0^o (frontal), +60^o, +40^o, +25^o, +15^o, -15^o, -25^o, -40^o and -60^o, respectively.

The 200 persons were split into three groups: group A, group B and an impostor group. There are 90 people each in group A and B, and 20 people in the impostor group. The class IDs for each group are given in Appendix A. Example images are shown in Fig. 2. Throughout the experiments, group A is used as a source of prior information while the impostor group and group B are used for verification tests. For most experiments there are 90 true claimant accesses and 90×20=1800 impostor attacks per angle (with the view of impostor faces matching the testing view). This restriction is relaxed in later experiments.

To reduce the effects of facial expressions and hair styles, closely cropped faces are used [43]; face windows, with a size of 56 rows and 64 columns, are extracted based on manually found eye locations. As in this paper we are proposing extensions to existing 2D approaches, we obtain normalized face windows for non-frontal views in the same way as for the frontal view (i.e. the location of the eyes is the same in each face window). This has a significant side effect: for large deviations from the frontal view (such as -60^o and +60^o) the effective size of facial characteristics is significantly larger than for the frontal view. The non-frontal face windows thus differ from the frontal face windows due to out-of-plane rotation of the face and scale. Example face windows are shown in Fig. 3.

3. FEATURE EXTRACTION

3.1. DCT Based System

In this work we utilize the DCTmod2 feature extraction technique [12], which is a modified form of DCT based feature extraction. First, a given face image is analyzed on a block by block basis; each block is $N_P \times N_P$ (here we use N_P=8) and overlaps neighbouring blocks by N_O pixels. Each block is decomposed in terms of orthogonal 2D Discrete Cosine Transform (DCT) basis functions [44]. A feature vector for a given block is then constructed as:

$$\mathbf{x} = \left[\, \Delta^h c_0 \;\; \Delta^v c_0 \;\; \Delta^h c_1 \;\; \Delta^v c_1 \;\; \Delta^h c_2 \;\; \Delta^v c_2 \;\;\; c_3 \;\; c_4 \; ... \; c_{M-1} \, \right]^T \tag{1}$$

where c_n represents the n-th DCT coefficient, while $\Delta^h c_n$ and $\Delta^v c_n$ represent the horizontal and vertical delta coefficients respectively. The deltas are computed using DCT coefficients extracted from neighbouring blocks. Compared to standard DCT feature extraction [22], the first three DCT coefficients are replaced by their respective horizontal and vertical deltas as a way of preserving discriminative information while alleviating the effects of illumination changes. Note that this feature extraction is only possible when a given block has vertical and horizontal neighbours. In this study we use M=15 (choice based on [12]), resulting in an 18 dimensional feature vector for each block. A further study of this feature extraction technique is given in [45].

The degree of overlap (N_O) has two effects: the first is that as overlap is increased the spatial area used to derive one feature vector is decreased (see Fig. 4 for an example); the second is that as the overlap is increased the number of feature vectors extracted from an image grows in a quadratic manner. Table 1 shows the amount of feature vectors extracted from a 56×64 face window using our implementation of the DCTmod2 extractor. As will be shown later, the larger the overlap (and hence the smaller the spatial area for each feature vector), the more the system is robust to view changes.

3.2. PCA Based System

In PCA based feature extraction [14, 15], a given face image is represented by a matrix containing grey level pixel values. The matrix is then converted to a face vector, $\mathbf{f}$, by concatenating all the columns. A D-dimensional feature vector, $\mathbf{x}$, is then obtained by:

$$\mathbf{x} = \mathbf{U}^T(\mathbf{f} - \mathbf{f}_\mu) \tag{2}$$

where $\mathbf{U}$ contains D eigenvectors (corresponding to the D largest eigenvalues) of the training data covariance matrix, and $\mathbf{f}_\mu$ is the mean of training face vectors. In our experiments we use frontal faces from group A to find $\mathbf{U}$ and $\mathbf{f}_\mu$.

It must be emphasized that in the PCA based approach, one feature vector represents the entire face (i.e. it is a holistic representation), while in the DCT approach one feature vector represents only a small portion of the face (i.e. it is a local feature representation).

4. GMM BASED CLASSIFIER

The distribution of training feature vectors for each person's face is modeled by a GMM [12, 13, 17]. There is also a secondary model, the generic model, which models the distribution of a population of faces, or interpreted alternatively, it represents "generic" face.

In the verification task we wish to find out whether a set of (test) feature vectors, $X = \{\mathbf{x}_i\}_{i=1}^{N_V}$, extracted from an unknown person's face, belongs to person C (which we will refer to as client C) or someone else (i.e. this is a two class recognition task). We first find the likelihood of set X belonging to client C with

$$P(X|\lambda_C) = \prod_{i=1}^{N_V} P(\mathbf{x}_i|\lambda_C) \tag{3}$$

where $P(\mathbf{x}|\lambda) = \sum_{g=1}^{N_G} w_g \; \mathcal{N}(\mathbf{x}|\mu_g, \mathbf{\Sigma}_g)$ and $\lambda = \{w_g, \mu_g, \mathbf{\Sigma}_g\}_{g=1}^{N_G}$. Here, $\mathcal{N}(\mathbf{x}|\mu, \mathbf{\Sigma})$ is a D-dimensional gaussian function with mean μ and diagonal covariance matrix $\mathbf{\Sigma}$:

$$\mathcal{N}(\mathbf{x}|\mu, \mathbf{\Sigma}) = \frac{\exp\left[-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)\right]}{(2\pi)^{\frac{D}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} \tag{4}$$

λ_C is the parameter set for client C, N_G is the number of gaussians and w_g is the weight for gaussian g (with constraints $\sum_{g=1}^{N_G} w_g = 1$ and $\forall\, g : w_g \geq 0$). Secondly, we obtain $P(X|\lambda_{generic})$, which is the likelihood of set X describing someone else's face (which we shall refer to as an *impostor* face). A log-likelihood ratio is then found using

$$\Lambda(X|\lambda_C, \lambda_{generic}) = \log P(X|\lambda_C) - \log P(X|\lambda_{generic}) \tag{5}$$

The verification decision is reached as follows: given a threshold t, the set X (i.e. the face in question) is classified as belonging to client C when $\Lambda(X|\lambda_C, \lambda_{generic}) \geq t$ or to an impostor when $\Lambda(X|\lambda_C, \lambda_{generic}) < t$. Note that $\Lambda(X|\lambda_C, \lambda_{generic})$ can be interpreted as an opinion of how more likely set X represents client C's face than an impostor's face, and hence can also be used in an open set identification system. Methods for obtaining the parameter set for the generic model and each client model are described in the following sections.

Note that in (3) each vector in the set $X = \{\mathbf{x}_i\}_{i=1}^{N_V}$ was assumed to be independent and identically distributed [16, 46]. When using local features, this results in the spatial relations between face parts to be not used, resulting in robustness to translations of the face [18, 19].

4.1. Classifier Training for the DCT Based System

First, the parameters for the generic model are obtained via the Expectation Maximization (EM) algorithm [16, 17, 47], using all 0^o data from group A. Here, the EM algorithm tunes the model parameters to optimize the maximum likelihood criterion. The parameters (λ) for each client model are then found by using the client's training data and adapting the generic model. The adaptation is traditionally done using a form of maximum *a posteriori* (MAP) estimation [17, 48]. In this work we shall also employ the MLLR model transformation approaches as adaptation methods. The choice of the adaptation technique depends on the non-frontal model synthesis method utilized later (Section 6).

4.2. Classifier Training for the PCA Based System

The subset of the FERET database that is utilized in this work has only one frontal image per person. In PCA-based feature extraction, this results in only one training vector, leading to necessary constraints in the structure of the classifier and the classifier's training paradigm.

The generic model and all client models for frontal faces are constrained to have only one component (i.e. one gaussian), with a diagonal covariance matrix[2]. The mean and the covariance matrix of the generic model are taken to be the mean and the covariance covariance matrix of feature vectors from group A, respectively. Instead of adaptation (as done in the DCT based system), each client model inherits the covariance matrix from the generic model. Moreover, the mean of each client model is taken to be the single training vector for that client.

4.3. Error Measures

There are two types of errors that can occur in a verification system: a false acceptance (FA), which occurs when the system accepts an impostor face, or a false rejection (FR), which occurs when the system refuses

[2]The assumption of a diagonal covariance matrix is supported by the fact that PCA derived feature vectors are decorrelated [16, 46].

a true face. The performance of verification systems is generally measured in terms of False Acceptance Rate (FAR) and False Rejection Rate (FRR), defined as:

$$\text{FAR} = \frac{\text{number of FAs}}{\text{number of impostor face presentations}} \tag{6}$$

$$\text{FRR} = \frac{\text{number of FRs}}{\text{number of true face presentations}} \tag{7}$$

To aid the interpretation of performance, the two error measures are often combined into one measure, called the Half Total Error Rate (HTER), which is defined as HTER = (FAR+FRR)/2. The HTER can be thought of as a particular case of the Decision Cost Function (DCF) [49, 50]:

$$\text{DCF} = \text{cost(FR)} \cdot P(\text{true face}) \cdot \text{FRR} + \text{cost(FA)} \cdot P(\text{impostor face}) \cdot \text{FAR} \tag{8}$$

where $P(\text{true face})$ is the prior probability that a true face will be presented to the system, $P(\text{impostor face})$ is the prior probability that an impostor face will be presented, cost(FR) is the cost of a false rejection and cost(FA) is the cost of a false acceptance. For the HTER, we have $P(\text{true face})$=$P(\text{impostor face})$=0.5 and the costs are set to 1.

A particular case of the HTER, known as the Equal Error Rate (EER), occurs when the system is adjusted (e.g. via tuning the threshold) so that FAR = FRR on a particular dataset. We use a global threshold (common across all clients) tuned to obtain the lowest EER on the test set, following the approach often used in speaker verification [3, 50][3].

5. MAXIMUM LIKELIHOOD LINEAR REGRESSION

In the Maximum Likelihood Linear Regression (MLLR) framework [36, 52], the adaptation of a given model is performed in two steps. In the first step the means are updated while in the second step the covariance matrices are updated, such that:

$$P(X|\widetilde{\lambda}) \geq P(X|\widehat{\lambda}) \geq P(X|\lambda) \tag{9}$$

where $\widetilde{\lambda}$ has both means and covariances updated while $\widehat{\lambda}$ has only means updated. The weights are not adapted as the main differences are assumed to be reflected in the means and covariances.

5.1. Adaptation of Means

Each adapted mean is obtained by applying a transformation matrix $\mathbf{W}_S$ to each original mean:

$$\widehat{\mu}_g = \mathbf{W}_S \nu_g \tag{10}$$

where $\nu_g = [\,1\;\mu_g^T\,]^T$ and W_S is a $D \times (D+1)$ transformation matrix which maximizes the likelihood of given training data. For $\mathbf{W}_S$ shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$ (see Section 5.3 below), the general form for finding $\mathbf{W}_S$ is:

$$\sum_{i=1}^{N_V}\sum_{r=1}^{N_S} P(g_r|\mathbf{x}_i,\lambda)\boldsymbol{\Sigma}_{g_r}^{-1}\mathbf{x}_i\nu_{g_r}^T = \sum_{i=1}^{N_V}\sum_{r=1}^{N_S} P(g_r|\mathbf{x}_i,\lambda)\boldsymbol{\Sigma}_{g_r}^{-1}\mathbf{W}_S\nu_{g_r}\nu_{g_r}^T \tag{11}$$

[3] We note that the posterior selection of the threshold can place an optimistic bias on the results [51].

where

$$P(g|\mathbf{x}_i, \lambda) = \frac{w_g \mathcal{N}(\mathbf{x}_i|\mu_g, \mathbf{\Sigma}_g)}{\sum_{n=1}^{N_G} w_n \mathcal{N}(\mathbf{x}_i|\mu_n, \mathbf{\Sigma}_n)} \tag{12}$$

As further elucidation is quite tedious, the reader is referred to [36] for the full solution of $\mathbf{W}_S$.

Two forms of $\mathbf{W}_S$ were originally proposed: full and "diagonal" [36]. We shall refer to MLLR transformation with a full transformation matrix as *full-MLLR*. When the transformation matrix is forced to be "diagonal", it has the following form:

$$\mathbf{W}_S = \begin{bmatrix} w_{1,1} & w_{1,2} & 0 & \cdots & 0 \\ w_{2,1} & 0 & w_{2,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{D,1} & 0 & 0 & \cdots & w_{D,D+1} \end{bmatrix} \tag{13}$$

We shall refer to MLLR transformation with a "diagonal" transformation matrix as *diag-MLLR*. We propose a third form of MLLR, where the "diagonal" elements are set to one, i.e.:

$$\mathbf{W}_S = \begin{bmatrix} w_{1,1} & 1 & 0 & \cdots & 0 \\ w_{2,1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{D,1} & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{14}$$

In other words, each mean is transformed by adding an offset; thus Eqn. (10) can be rewritten as:

$$\widehat{\mu}_g = \mu_g + \mathbf{\Delta}_S \tag{15}$$

where $\mathbf{\Delta}_S$ maximizes the likelihood of given training data. This leads to the following solution:

$$\mathbf{\Delta}_S = \left[\sum_{r=1}^{N_S} \sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda) \mathbf{\Sigma}_{g_r}^{-1} \right]^{-1} \left[\sum_{r=1}^{N_S} \sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda) \mathbf{\Sigma}_{g_r}^{-1} (\mathbf{x}_i - \mu_{g_r}) \right] \tag{16}$$

The derivation for the above solution is given in Appendix B. We shall refer to this form of MLLR as *offset-MLLR*.

5.2. Adaptation of Covariance Matrices

Once the new means are obtained, each new covariance matrix is found using [52]:

$$\widetilde{\mathbf{\Sigma}}_g = \mathbf{B}_g^T \mathbf{H}_S \mathbf{B}_g \tag{17}$$

where

$$\mathbf{B}_g = \mathbf{C}_g^{-1} \tag{18}$$

$$\mathbf{C}_g \mathbf{C}_g^T = \mathbf{\Sigma}_g^{-1} \tag{19}$$

Here, Eqn. (19) is a form of Cholesky decomposition [53]. $\mathbf{H}_S$, shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$, is found with:

$$\mathbf{H}_S = \frac{\sum_{r=1}^{N_S} \left\{ \mathbf{C}_{g_r}^T \left[\sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda)(\mathbf{x}_i - \widehat{\mu}_{g_r})(\mathbf{x}_i - \widehat{\mu}_{g_r})^T \right] \mathbf{C}_{g_r} \right\}}{\sum_{i=1}^{N_V} \sum_{r=1}^{N_S} P(g_r|\mathbf{x}_i, \lambda)} \tag{20}$$

The covariance transformation may be either full or diagonal. When the full transformation is used, full covariance matrices can be produced even if the original covariances were diagonal to begin with. To avoid this, the off-diagonal elements of $\mathbf{H}_S$ can be set to zero. In this work we restrict ourselves to the use of diagonal covariance matrices to reduce the number of parameters that need to be estimated. For full covariance matrices the dataset may not be large enough to robustly estimate the transformation parameters, which could result in the transformed covariance matrices being ill-conditioned [52].

5.3. Regression Classes

If each gaussian is transformed individually, then for full-MLLR there are $D^2 + 2D$ parameters to estimate per gaussian (i.e. $D \times (D + 1)$ parameters for each mean and D parameters for each covariance matrix); for diag-MLLR, there are $D + D + D = 3D$ parameters and for offset-MLLR there are $D + D = 2D$ parameters. Ideally each gaussian would have its own transform, however in practical applications the training dataset may not be large enough to reliably estimate the required number of parameters. One way of working around the small training dataset problem is to share a transform across two or more gaussians [36, 52]. We define which gaussians are to share a transform by clustering the gaussians based on the distance between their means.

We define a regression class as $\{g_r\}_{r=1}^{N_S}$ where g_r is the r-th gaussian in the class; all gaussians in a regression class share the same mean and covariance transforms. In our experiments we vary the number of regression classes from one (all gaussians share one mean and one covariance transform) to 32 (each gaussian has its own transform). The number of regression classes is denoted as N_R.

6. SYNTHESIZING CLIENT MODELS FOR NON-FRONTAL VIEWS

6.1. DCT Based System

In the MLLR based model synthesis technique, we first transform, using prior information, the frontal generic model into a non-frontal generic model for angle Θ. For full-MLLR and diag-MLLR, the parameters which describe the transformation of the means and covariances are $\Psi = \{\mathbf{W}_g, \mathbf{H}_g\}_{g=1}^{N_G}$, while for offset-MLLR the parameters are $\Psi = \{\mathbf{\Delta}_g, \mathbf{H}_g\}_{g=1}^{N_G}$. $\mathbf{W}_g$, $\mathbf{\Delta}_g$ and $\mathbf{H}_g$ are found as described in Section 5. When several gaussians share the same transformation parameters, the shared parameters are replicated for each gaussian in question. To synthesize a client model for angle Θ, the previously learned transformations are applied to the client's frontal model. The weights are kept the same as for the frontal model. Moreover, each frontal client model is derived from the frontal generic model by MLLR.

6.2. PCA Based System

For the PCA based system, we utilize MLLR based model synthesis in a similar way as described in the previous section. The only difference is that each non-frontal client model inherits the covariance matrix from the corresponding non-frontal generic model. Moreover, as each client model has only one gaussian, we note that the MLLR transformations are "single point to single point" transformations, where the points are the old and new mean vectors.

As described in Section 4.2, the mean of each client model is taken to be the single training vector available. Thus in this case a transformation in the feature domain is equivalent to a transformation in the model domain. It is therefore possible to use transformations which are not of the "single point to single point" type. Let us suppose that we have the following multi-variate linear regression model:

$$\mathbf{B} = \mathbf{A}\mathbf{W} \tag{21}$$

$$\begin{bmatrix} \mathbf{b}_1^T \\ \mathbf{b}_2^T \\ \vdots \\ \mathbf{b}_N^T \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{a}_1^T \\ 1 & \mathbf{a}_2^T \\ & \vdots \\ 1 & \mathbf{a}_N^T \end{bmatrix} \begin{bmatrix} w_{1,1} & \cdots & w_{1,D} \\ w_{2,1} & \cdots & w_{2,D} \\ \vdots & \vdots & \vdots \\ w_{D+1,1} & \cdots & w_{D+1,D} \end{bmatrix} \tag{22}$$

where $N > D + 1$, with D being the dimensionality of $\mathbf{a}$ and $\mathbf{b}$. $\mathbf{W}$ is a matrix of unknown regression parameters. Under the sum-of-least-squares regression criterion, $\mathbf{W}$ can be found using [53]:

$$\mathbf{W} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{B} \tag{23}$$

Compared to MLLR, this type of regression finds a common relation between two *sets* of points; hence it may be more accurate than MLLR. Given a set of PCA-derived feature vectors from group A, representing faces at 0^o and Θ, we find $\mathbf{W}$. We can then synthesize the single mean for Θ from client C's 0^o mean using:

$$\mu^{\Theta} = \left[\, 1 \;\; (\mu^{0^o})^T \,\right] \; \mathbf{W} \tag{24}$$

We shall refer to this PCA-specific linear regression based technique as *LinReg*. We note that for this synthesis technique, $(D{+}1){\times}D{=}D^2{+}D$ parameters need to be estimated.

7. EXTENDING FRONTAL MODELS

In order for the system to automatically handle non-frontal views, each client's frontal model is extended by concatenating it with synthesized non-frontal models. The frontal generic model is also extended with non-frontal generic models. Formally, an extended model is created using:

$$\begin{aligned} \lambda^{extended} &= \lambda^{0^o} \sqcup \lambda^{+60^o} \sqcup \lambda^{+40^o} \cdots \sqcup \lambda^{-40^o} \sqcup \lambda^{-60^o} \\ &= \sqcup_{i \in \Phi} \lambda^i \end{aligned} \tag{25}$$

where λ^{0^o} represents a frontal model, Φ is a set of angles, e.g., $\Phi = \{\, 0^o, +60^o, \cdots, +15^o, -15^o, \cdots, -60^o \,\}$, and $\sqcup$ is an operator for joining GMM parameter sets. Let us suppose we have two GMM parameter sets, λ^x and λ^y, comprised of parameters for N_G^x and N_G^y gaussians, respectively. The $\sqcup$ operator is defined as follows:

$$\begin{aligned} \lambda^z &= \lambda^x \sqcup \lambda^y \\ &= \left\{\alpha w_g^x,\ \mu_g^x,\ \mathbf{\Sigma}_g^x\right\}_{g=1}^{N_G^x} \cup \left\{\beta w_g^y,\ \mu_g^y,\ \mathbf{\Sigma}_g^y\right\}_{g=1}^{N_G^y} \end{aligned} \tag{26}$$

where $\alpha = N_G^x/(N_G^x + N_G^y)$ and $\beta = 1 - \alpha$.

8. EXPERIMENTS AND DISCUSSION

8.1. DCT Based System

In the first experiment we studied how the overlap setting in the DCTmod2 feature extractor and number of gaussians in the classifier affects performance and robustness. Client models were trained on frontal faces and tested on faces at 0^o and +40^o views; impostor faces matched the testing view. Traditional MAP adaptation was used to obtain the client models. Results, in terms of EER (Section 4), are shown in Figs. 5 and 6.

When testing with frontal faces, the overall trend is that as the overlap increases more gaussians are needed to decrease the error rate. This can be interpreted as follows: the smaller the area used in the derivation of each feature vector, the more gaussians are required to adequately model the face. When testing with non-frontal faces, the overall trend is that as the overlap increases, the lower the error rate. There is also a less defined trend when the overlap is 4 pixels or greater: the more gaussians, the lower the error rate[4]. While not shown here, the DCT based system obtained similar trends for non-frontal views other than +40^o. The best performance for +40^o faces is achieved with an overlap of 7 pixels and 32 gaussians, resulting in an EER close to 10%. We chose this configuration for further experiments.

In the second experiment we evaluated the performance of models synthesized via the full-MLLR, diag-MLLR and offset-MLLR techniques, for varying number of regression classes. Results are presented in Tables 2 to 5. As can be observed, the full-MLLR technique falls apart when there are two or more regression classes. Its best results (obtained for one regression class) are in some cases worse than for standard frontal models. Frontal client models, obtained by using full-MLLR as an adaptation method, resulted in an EER of 0% for frontal faces for all configurations of regression classes. Thus while the full-MLLR transformation is adequate for adapting the frontal generic model to frontal client models, the synthesis results suggest that the transformation is only reliable when applied to the specific model it was trained to transform. Further investigation of the sensitivity of the full-MLLR transform, presented in Appendix C, shows that the full-MLLR transform is easily affected by the starting point. We conjecture that this is probably due to the training dataset being too small to robustly estimate the transformation parameters.

Compared to full-MLLR, the diag-MLLR technique obtains lower EERs (Table 3). We note that the number of transformation parameters for diag-MLLR is significantly less than for full-MLLR. The overall error rate (across all angles) decreases as the number of regression classes increases from one to eight; the performance then deteriorates for higher numbers of regression classes. The results are consistent with the scenario that once the number of regression classes reaches a certain point, the training dataset is too small to obtain robust transformation parameters. The best performance, obtained at eight regression classes, is for all angles better than the performance of standard frontal models.

The offset-MLLR technique (Table 4) has the lowest EERs when compared to full-MLLR and diag-MLLR. It must be noted that it also has the least number of transformation parameters. The overall error rate consistently decreases as the number of regression classes increases from one to 32. The best

[4]This is true up to a point: eventually the error rate will go up as there will be too many gaussians to train adequately with the small size of the training dataset. Preliminary experiments showed that there was little performance gain when using more than 32 gaussians.

performance, obtained at 32 regression classes, is for all angles better than the performance of standard frontal models.

8.2. PCA Based System

In the first experiment we studied how the dimensionality of the feature vectors used in the PCA based system affects robustness to varying pose. Client models were trained on frontal faces and tested on faces from -60^o to +60^o views; impostor faces matched the testing view. Results for -60^o to 0^o are shown in Fig. 7 (results for +15^o to +60^o, not shown here, have very similar trends).

As can be observed, a dimensionality of at least 40 is required to achieve perfect verification on frontal faces (this is consistent with the results presented in [23]). For non-frontal faces at $\pm 60^o$ and $\pm 40^o$, the error rate generally increases as the dimensionality increases, and saturates when the dimensionality is about 15. Hence there is somewhat of a trade-off between the error rates on frontal faces and non-frontal faces, controlled by the dimensionality. Since in this work we are pursuing extensions to standard 2D approaches, the dimensionality has been fixed at 40 for further experiments. Using a lower dimensionality of, say 4, offers better performance for non-frontal faces, however it comes at the cost of an EER of about 10% on frontal faces.

We note that the PCA based system (which is holistic in nature) is much more affected by view changes than the DCT based system. This can be attributed to the rigid preservation of spatial relations between face areas, which is in contrast to the DCT/GMM based approach, where the spatial relations between face parts are very loose. The loose spatial relations allow for the deformations and movements of face areas, which can occur due to view changes. Interestingly, recent empirical evidence suggests that humans recognize faces by parts rather than in a holistic manner [54].

In the second experiment we evaluated the performance of models synthesized using LinReg and MLLR-based techniques. As there is only one gaussian per client model, there was only one regression class for MLLR techniques. Results in Table 6 show that model synthesis with full-MLLR and diag-MLLR was unsuccessful. Since the LinReg technique works quite well and has a similar number of free parameters as full-MLLR, we attribute the failure of full-MLLR and diag-MLLR to their sensitivity to the starting point, which is described in Appendix C. While models synthesized by offset-MLLR exhibit better performance than standard frontal models, they are easily outperformed by models synthesized via the LinReg technique. This supports the view that “single point to single point” type transformations (such as MLLR) are less useful for a system utilizing PCA derived features.

8.3. Performance of Extended Frontal Models

In the experiments described in Sections 8.1 and 8.2, it was assumed that the angle of the face is known. In this section we progressively remove this constraint and propose to handle varying pose by extending each client’s frontal model with the client’s synthesized non-frontal models.

In the first experiment we compared the performance of extended models to frontal models and models synthesized for a specific angle; impostor faces matched the test view. For the DCT based system, each client’s frontal model was extended with models synthesized by the offset-MLLR technique (with 32 regression classes) for the following angles: $\pm 60^o$, $\pm 40^o$ and $\pm 25^o$. Synthesized models for $\pm 15^o$ were not used since they provided little performance benefit over the 0^o model (see Table 5). The frontal generic model was also extended with non-frontal generic models. Since each frontal model had 32 gaussians, each extended model had 224 gaussians. Following the offset-MLLR based model synthesis paradigm, each frontal client model was derived from the frontal generic model using offset-MLLR.

For the PCA based system, model synthesis was accomplished using LinReg. Each client's frontal model was extended for the following angles: $\pm 60^o$, $\pm 40^o$, $\pm 25^o$ and $\pm 15^o$. The frontal generic model was also extended with non-frontal generic models. Since each frontal model had one gaussian, each extended model had nine gaussians.

As can be seen in Tables 7 and 8, for most angles only a small reduction in performance is observed when compared to models synthesized for a specific angle. These results suggest that the model extension approach could be used instead of selecting the most appropriate synthesized model (via detection of the face angle), thus reducing the complexity of a multi-view face verification system.

In the first experiment impostor attacks and true claims were evaluated for each angle separately. In the second experiment we relaxed this restriction and allowed true claims and impostor attacks to come from all angles, resulting in $90 \times 9 = 810$ true claims and $90 \times 20 \times 9 = 16200$ impostor attacks; an overall EER was then found. For both DCT and PCA based systems two types of models were used: frontal and extended. For the DCT based system, frontal models were derived from the generic model using offset-MLLR. From the results presented in Table 9, it can be observed that model extension reduces the error rate in both PCA and DCT based systems, with the DCT based system achieving the lowest EER. The largest error reduction is present in the PCA based system, where the EER is reduced by approximately 58%; for the DCT based system, the EER is reduced by approximately 26%.

9. CONCLUSIONS AND FUTURE WORK

In this paper we addressed the pose mismatch problem which can occur in face verification systems that have only a single (frontal) face image available for training. In the framework of a Bayesian classifier based on mixtures of gaussians, the problem was tackled through extending each frontal face model with artificially synthesized models for non-frontal views. The synthesis was accomplished via methods based on several implementations of Maximum Likelihood Linear Regression (MLLR) (originally developed for tuning speech recognition systems), and standard multi-variate linear regression (LinReg). To our knowledge this is the first time MLLR has been adapted for face verification.

All synthesis techniques rely on prior information and learn how face models for the frontal view are related to face models at non-frontal views. The synthesis and extension approach was evaluated by applying it to two face verification systems: a holistic system (utilizing PCA derived features) and a local feature system (using DCT derived features).

Experiments on the FERET database suggest that for the PCA based system, the LinReg technique (which is based on a common relation between two *sets* of points) is more suited than the MLLR based techniques (which are "single point to single point" transforms in the PCA based system). For the DCT based system, the results show that synthesis via a new MLLR implementation obtains better performance than synthesis based on traditional MLLR (mainly due to a lower number of free parameters). The results further suggest that extending frontal models considerably reduces errors in both systems.

The results also show that the standard DCT based system (trained on frontal faces) is less affected by view changes than the PCA based system. This can be attributed to the parts based representation of the face (via local features) and, due to the classifier based on mixtures of gaussians, the lack of constraints on spatial relations between face parts. The lack of constraints allows for deformations and movements of face areas, which can occur due to view changes. This is in contrast to the PCA based system, where, due to the holistic representation, the spatial relations are rigidly kept. Interestingly, recent empirical evidence suggests that humans recognize faces by parts rather than in a holistic manner [54].

Future areas of research include whether it is possible to interpolate between two synthesized models to generate a third model for a view for which there is no prior information. A related question is how many discrete views are necessary to adequately cover a wide range of poses. The dimensionality reduction matrix $\mathbf{U}$ in the PCA approach was defined using only frontal faces; higher performance may be obtained by incorporating non-frontal faces. The local feature/GMM approach can be extended by embedding position information into each feature vector [19, 21], thus placing a weak constraint on the face areas each gaussian can model (as opposed to the current absence of constraints). This in turn could make the transformation of frontal models to non-frontal models more accurate, as different face areas effectively "move" in different ways when there is a view change. Alternatively, the GMM based classifier can be replaced with a (more complex) pseudo-2D Hidden Markov Model based classifier [19, 21, 22], where there is a more stringent constraint on the face areas modeled by each gaussian. Lastly, it would be useful to evaluate alternative size normalization approaches in order to address the scaling problem mentioned in Section 2.

ACKNOWLEDGEMENTS

National ICT Australia is funded by the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council. The Swiss National Science Foundation supported this work through the National Center of Competence in Research (NCCR) on Interactive Multimodal Information Management (IM2). The authors are indebted to Ronan Collobert, Alexei Pozdnoukhov and the anonymous reviewers for useful suggestions.

APPENDIX A: CLASS IDS FOR GROUP A, B AND THE IMPOSTOR GROUP.

Classes for group A: 00019, 00029, 00268, 00647, 00700, 00761, 01013 to 01018, 01020 to 01032, 01034 to 01048, 01050, 01052, 01054 to 01066, 01068 to 01076, 01078 to 01081, 01083, 01084, 01085, 01086, 01088 to 01092, 01094, 01098, 01101, 01103, 01106, 01108, 01111, 01117, 01124, 01125, 01156, 01162, 01172.

Classes for group B: 01095 to 01097, 01099, 01100, 01102, 01104, 01105, 01107, 01109, 01110, 01112 to 01116, 01118 to 01120, 01122, 01127 to 01136, 01138 to 01142, 01144, 01146 to 01150, 01152 to 01155, 01157 to 01161, 01163 to 01168, 01170, 01171, 01173 to 01178, 01180 to 01202, 01204 to 01206.

Classes for impostor group: 01019, 01033, 01049, 01051, 01053, 01067, 01077, 01082, 01087, 01093, 01121, 01123, 01126, 01137, 01143, 01145, 01151, 01169, 01179, 01203.

APPENDIX B: DERIVATION OF OFFSET-MLLR

In the offset-MLLR approach, each mean is redefined as [c.f. Eqn. (10)]:

$$\widehat{\mu}_g = \mu_g + \mathbf{\Delta}_g \tag{27}$$

where $\mathbf{\Delta}_g$ maximizes the likelihood of given training data. Substituting (27) into (4) results in:

$$P(\mathbf{x}|\widehat{\mu}_g, \mathbf{\Sigma}_g) = \frac{\exp\left[-\frac{1}{2}(\mathbf{x} - \{\mu_g + \mathbf{\Delta}_g\})^T \mathbf{\Sigma}_g^{-1}(\mathbf{x} - \{\mu_g + \mathbf{\Delta}_g\})\right]}{(2\pi)^{\frac{D}{2}}|\mathbf{\Sigma}_g|^{\frac{1}{2}}} \tag{28}$$

In the framework of the Expectation Maximization (EM) algorithm, we assume that our training data X is incomplete and assume the existence of missing data $Y = \{y_i\}_{i=1}^{N_V}$, where the values of y_i indicate the mixture component (i.e. the gaussian) that "generated" $\mathbf{x}_i$. Thus $y_i \in [1, N_G]\ \forall\ i$ and $y_i = m$ if the i-th feature vector ($\mathbf{x}_i$) was "generated" by the m-th gaussian. An auxiliary function is defined as follows:

$$Q(\lambda, \lambda^{\mathtt{old}}) = \mathrm{E}_Y\left[\log P(X, Y|\lambda) \mid X, \lambda^{\mathtt{old}}\right] \tag{29}$$

It can be shown [47], that maximizing $Q(\lambda, \lambda^{\mathtt{old}})$, i.e.:

$$\lambda^{\mathtt{new}} = \arg\max_{\lambda} Q(\lambda, \lambda^{\mathtt{old}}) \tag{30}$$

results in $P(X|\lambda^{\mathtt{new}}) \geq P(X|\lambda^{\mathtt{old}})$ (i.e. the likelihood of the training data X increases). Evaluating the expectation in Eqn. (29) results in [55]:

$$\begin{aligned} Q(\lambda, \lambda^{\mathtt{old}}) &= \sum_{g=1}^{N_G}\sum_{i=1}^{N_V} \log[w_g]\ P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) + \sum_{g=1}^{N_G}\sum_{i=1}^{N_V} \log[P(\mathbf{x}_i|\mu_g, \mathbf{\Sigma}_g)]\ P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) && (31) \\ &= Q_1 \quad + \quad Q_2 && (32) \end{aligned}$$

where

$$P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) = \frac{w_g^{\mathtt{old}} \mathcal{N}(\mathbf{x}_i|\mu_g^{\mathtt{old}}, \Sigma_g^{\mathtt{old}})}{\sum_{n=1}^{N_G} w_n^{\mathtt{old}} \mathcal{N}(\mathbf{x}_i|\mu_n^{\mathtt{old}}, \Sigma_n^{\mathtt{old}})} \tag{33}$$

A common maximization technique is to take the derivative of $Q(\lambda, \lambda^{\mathtt{old}})$ with respect to the parameter to be maximized and set the result to zero. Since we are interested in finding $\mathbf{\Delta}_g$, we only need to take the derivate of Q_2:

$$\begin{aligned} 0 &= \frac{\partial}{\partial \mathbf{\Delta}_g} \sum_{g=1}^{N_G}\sum_{i=1}^{N_V} \log[P(\mathbf{x}_i|\mu_g, \mathbf{\Sigma}_g)]\ P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) && (34) \\ &= \frac{\partial}{\partial \mathbf{\Delta}_g} \sum_{g=1}^{N_G}\sum_{i=1}^{N_V} \left[-\frac{1}{2}(\mathbf{x}_i - \{\mu_g + \mathbf{\Delta}_g\})^T \mathbf{\Sigma}_g^{-1}(\mathbf{x}_i - \{\mu_g + \mathbf{\Delta}_g\})\right] P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) && (35) \\ &= \sum_{i=1}^{N_V} P(g|\mathbf{x}_i, \lambda^{\mathtt{old}}) \mathbf{\Sigma}_g^{-1}(\mathbf{x}_i - \{\mu_g + \mathbf{\Delta}_g\}) && (36) \end{aligned}$$

where $-\frac{D}{2}\log(2\pi)$ and $-\frac{1}{2}\log(|\mathbf{\Sigma}_g|)$ were omitted in Eqn. (35) since they vanish when taking the derivative. Re-arranging Eqn. (36) yields:

$$\boldsymbol{\Delta}_g = \frac{\sum_{i=1}^{N_V} P(g|\mathbf{x}_i, \lambda^{\texttt{old}})\mathbf{x}_i}{\sum_{i=1}^{N_V} P(g|\mathbf{x}_i, \lambda^{\texttt{old}})} - \mu_g \tag{37}$$

Substituting Eqn. (37) into Eqn. (27) yields:

$$\widehat{\mu}_g = \frac{\sum_{i=1}^{N_V} P(g|\mathbf{x}_i, \lambda^{\texttt{old}})\mathbf{x}_i}{\sum_{i=1}^{N_V} P(g|\mathbf{x}_i, \lambda^{\texttt{old}})} \tag{38}$$

which is the standard maximum likelihood re-estimation formula for the mean. Following [36], we modify the re-estimation formula for tied transformation parameters (e.g. a single $\boldsymbol{\Delta}$ shared by all means). If $\boldsymbol{\Delta}_S$ is shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$, Eqn. (35) is modified to:

$$0 = \frac{\partial}{\partial \boldsymbol{\Delta}_S} \sum_{r=1}^{N_S} \sum_{i=1}^{N_V} \left[-\frac{1}{2}(\mathbf{x}_i - \{\mu_{g_r} + \boldsymbol{\Delta}_S\})^T \boldsymbol{\Sigma}_{g_r}^{-1} (\mathbf{x}_i - \{\mu_{g_r} + \boldsymbol{\Delta}_S\}) \right] P(g_r|\mathbf{x}_i, \lambda^{\texttt{old}}) \tag{39}$$

$$= \sum_{r=1}^{N_S} \sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda^{\texttt{old}}) \boldsymbol{\Sigma}_{g_r}^{-1} (\mathbf{x}_i - \{\mu_{g_r} + \boldsymbol{\Delta}_S\}) \tag{40}$$

which leads to:

$$\boldsymbol{\Delta}_S = \left[\sum_{r=1}^{N_S} \sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda^{\texttt{old}}) \boldsymbol{\Sigma}_{g_r}^{-1} \right]^{-1} \left[\sum_{r=1}^{N_S} \sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i, \lambda^{\texttt{old}}) \boldsymbol{\Sigma}_{g_r}^{-1} (\mathbf{x}_i - \mu_{g_r}) \right] \tag{41}$$

APPENDIX C: ANALYSIS OF MLLR SENSITIVITY

The results presented in Section 8.1 show that the full-MLLR technique is only reliable when applied directly to the specific model it was trained to transform, making the full-MLLR transform unsuitable for model synthesis (where a related model is transformed, instead of the model for which the transformation was learned). In this section we explore this observation further by measuring how sensitive the full-MLLR, diag-MLLR and offset-MLLR transforms are to perturbations of the model they were trained to transform.

The sensitivity is measured as follows. The transformation of the frontal generic model to a +60^o generic model is learned (using 32 regression classes) and the average log-likelihood of +60^o data from group A is found:

$$\mathcal{A}(X|\lambda_{generic}^{+60^o}) = \frac{1}{N_V} \log P(X|\lambda_{generic}^{+60^o}) \tag{42}$$

The mean vectors of the frontal generic model are then "corrupted" by adding gaussian noise with zero mean and various levels of variance. Formally:

$$\left[\mu_g^{corrupted}\right]^T = \left[\mu_{g,d}^{original} + \mathcal{R}(0, \sigma^2)\right]_{d=1}^{D} \tag{43}$$

where $\mu_{g,d}$ is the d-th element of μ_g and $\mathcal{R}(0, \sigma)$ is a gaussian distributed random variable with zero mean and variance σ^2. The previously learned transformation is applied to the "corrupted" frontal generic model to obtain a "corrupted" +60^o generic model. The average log-likelihood of +60^o data from group A is then found as per Eqn. (42). This process is repeated ten times for each variance setting and the mean of the average log-likelihood is taken. The mean value represents how well the transformed model represents the +60^o data; the lower the value, the worse the representation. Results are presented in Table 10.

By treating the mean vectors of frontal client models as noisy instances of the frontal generic model mean vectors (where the frontal client models were derived from the original frontal generic model), it is possible to measure the overall "variance" of the frontal mean vectors; this is the variance that a synthesis technique must handle. While the frontal client models also differ from the frontal generic model in their covariance matrices, we believe this approach nevertheless provides suggestive results.

The full-MLLR, diag-MLLR and offset-MLLR approaches for deriving frontal client models (from the original frontal generic model) obtained similar overall "variance" of frontal client means of around 90. From the results shown in Table 10 it can be observed that the full-MLLR transformation is easily affected by small perturbations of the frontal generic model. Close to level of the required variance (i.e. at 100), the full-MLLR approach produces a +60^o generic model which very poorly represents the data on which the transform was originally trained. In comparison, the diag-MLLR and offset-MLLR transforms are largely robust to perturbations of the frontal generic model, with the offset-MLLR approach the most stable.

REFERENCES

[1] M. Lockie (editor), Facial verification bureau launched by police IT group, Biometric Technology Today 10 (3) (2002) 3-4.

[2] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, A. Senior, C.-F. Shu, Y.L. Tian, Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking, IEEE Signal Processing Magazine 22 (2) (2005) 38-51.

[3] J. Ortega-Garcia, J. Bigun, D. Reynolds, J. Gonzales-Rodriguez, Authentication gets personal with biometrics, IEEE Signal Processing Magazine 21 (2) (2004) 50-62.

[4] J.D. Woodward, Biometrics: privacy's foe or privacy's friend?, Proc. IEEE, 85 (9) (1997) 1480-1492.

[5] J. Czyz, J. Kittler, L. Vandendorpe, Multiple classifer combination for face-based identity verifcation, Pattern Recognition, 37 (7) (2004) 1459-1469.

[6] K. Messer, J. Kittler, M. Sadeghi, S. Marcel, C. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, J. Czyz, L. Vandendorpe, S. Srisuk, M. Petrou, W. Kurutach, A. Kadyrov, R. Paredes, B. Kepenekci, F.B. Tek, G.B. Akar, F. Deravi, N. Mavity, Face verification competition on the XM2VTS database, in: Proc. 4th Int. Conf. Audio- and Video-Based Biometric Person Authentication (AVBPA), Guildford, 2003, pp. 964-974.

[7] D. Graham, N. Allinson, Face recognition from unfamiliar views: subspace methods and pose dependency, in: Proc. 3rd IEEE Int. Conf. Automatic Face and Gesture Recognition (AFGR), Nara, 1998, pp. 348-353.

[8] R. Gross, J. Yang, A. Waibel, Growing gaussian mixture models for pose invariant face recognition, in: Proc. 15th Int. Conf. Pattern Recognition (ICPR), Barcelona, 2000, Vol. 1, pp. 1088-1091.

[9] A. Pentland, B. Moghaddam, T. Starner, View-based and modular eigenspaces for face recognition, in: Proc. Int. Conf. Computer Vision and Pattern Recognition (CVPR), Seattle, 1994, pp. 84-91.

[10] J.J. Atick, P.A. Griffin, A.N. Redlich, Statistical approach to shape from shading: reconstruction of three-dimensional face surfaces from single two-dimensional images, Neural Computation 8 (1996) 1321-1340.

[11] V. Blanz, S. Romdhani, T. Vetter, Face identification across different poses and illuminations with a 3D morphable model, in: Proc. 5th IEEE Int. Conf. Automatic Face and Gesture Recognition (AFGR), Washington, D.C., 2002, pp. 192-197.

[12] C. Sanderson, K.K. Paliwal, Fast features for face authentication under illumination direction changes, Pattern Recognition Letters 24 (14) (2003) 2409-2419. http://dx.doi.org/10.1016/S0167-8655(03)00070-9

[13] S. Lucey, T. Chen, A GMM parts based face representation for improved verification through relevance adaptation, in: Proc. Computer Vision and Pattern Recognition (CVPR), Washington, D.C., 2004, Vol. 2, pp. 855-861.

[14] M. Kirby, L. Sirovich, Application of the Karhunen-Loève procedure for the characterization of human faces, IEEE Trans. Pattern Analysis and Machine Intelligence 12 (1) (1990) 103-108.

[15] M. Turk, A. Pentland, Eigenfaces for recognition, J. Cognitive Neuroscience 3 (1) (1991) 71-86.

[16] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, John Wiley & Sons, USA, 2001.

[17] D. Reynolds, T. Quatieri, R. Dunn, Speaker verification using adapted gaussian mixture models, Digital Signal Processing 10 (1-3) (2000) 19-41.

[18] F. Cardinaux, C. Sanderson, S. Marcel, Comparison of MLP and GMM classifiers for face verification on XM2VTS, in: Proc. 4th Int. Conf. Audio- and Video-Based Biometric Person Authentication (AVBPA), Guildford, 2003, pp. 911-920. http://www.springerlink.com/link.asp?id=bhgqn3nca3nad29c

[19] C. Sanderson, F. Cardinaux, S. Bengio, On Accuracy/Robustness/Complexity Trade-Offs in Face Verification, in: Proc. 3rd Int. Conf. Information Technology and Applications, Sydney, 2005, Vol. 1, pp. 650-655. http://ieeexplore.ieee.org/xpl/abs_free.jsp?arNumber=1488880

[20] R. Brunelli, T. Poggio, Face recognition: features versus templates, IEEE Trans. Pattern Analysis and Machine Intelligence 15 (10) (1998) 1042-1052.

[21] F. Cardinaux, C. Sanderson, S. Bengio, Face verification using adapted generative models, in: Proc. 6th IEEE Int. Conf. Automatic Face and Gesture Recognition (AFGR), Seoul, 2004, pp. 825-830. http://ieeexplore.ieee.org/xpl/abs_free.jsp?arNumber=1301636

[22] S. Eickeler, S. Müller, G. Rigoll, Recognition of JPEG compressed face images based on statistical methods, Image and Vision Computing 18 (4) (2000) 279-287.

[23] F. Samaria, Face Recognition Using Hidden Markov Models, PhD Thesis, University of Cambridge, UK, 1994.

[24] B. Duc, S. Fischer, J. Bigün, Face authentication with Gabor information on deformable graphs, IEEE Trans. Image Processing 8 (4) (1999) 504-516.

[25] M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C. v.d. Malsburg, R.P. Würtz, W. Konen, Distortion invariant object recognition in the dynamic link architecture, IEEE Trans. Computers 42 (3) (1993) 300-311.

[26] R. Chellappa, C.L. Wilson, S. Sirohey, Human and machine recognition of faces: a survey, Proc. IEEE 83 (5) (1995) 705-740.

[27] M.A. Grudin, On internal representations in face recognition systems, Pattern Recognition 33 (7) (2000) 1161-1177.

[28] S.G. Kong, J. Heo, B.R. Abidi, J. Paik, M.A. Abidi, Recent advances in visual and infrared face recognition - a review, Computer Vision and Image Understanding 97 (1) (2005) 103-135.

[29] J. Zhang, Y. Yan, M. Lades, Face recognition: eigenfaces, elastic matching, and neural nets, Proc. IEEE, 85 (9) (1997) 1422-1435.

[30] J.-L. Dugelay, J.-C. Junqua, C. Kotropoulos, R. Kuhn, F. Perronnin, I. Pitas, Recent advances in biometric person authentication, in: Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Orlando, 2002, Vol. IV, pp. 4060-4063.

[31] C. Sanderson, K.K. Paliwal, Identity verification using speech and face information, Digital Signal Processing, 14 (5) (2004) 449-480. http://dx.doi.org/10.1016/j.dsp.2004.05.001

[32] J.L. Wayman, Digital signal processing in biometric identification: a review, in: Proc. IEEE Int. Conf. Image Processing (ICIP), Rochester, 2002, Vol. 1, pp. 37-40.

[33] E. Hjelmås, B.K. Low, Face detection: a survey, Computer Vision and Image Understanding 83 (3) (2001) 236-274.

[34] M-H. Yang, D.J. Kriegman, N. Ahuja, Detecting faces in images: a survey, IEEE Trans. Pattern Analysis and Machine Intelligence, 24 (1) (2002) 34-58.

[35] Behrooz Kamgar-Parsi, Behzad Kamgar-Parsi, A. Jain, J. Dayhoff, Aircraft detection: a case study in using human similarity measure, IEEE Trans. Pattern Analysis and Machine Intelligence 23 (12) (2001) 1404-1414.

[36] C.J. Leggetter, P.C. Woodland, Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models, Computer Speech and Language 9 (2) (1995) 171-185.

[37] J. Mariéthoz, S. Bengio, A comparative study of adaptation methods for speaker verification, in: Proc. Int. Conf. Spoken Language Processing (ICSLP), Denver, 2002, pp. 581-584.

[38] D. Beymer, T. Poggio, Face recognition from one example view, in: Proc. 5th Int. Conf. Computer Vision (ICCV), Cambridge, 1995, pp. 500-507.

[39] P. Niyogi, F. Girosi, T. Poggio, Incorporating prior information in machine learning by creating virtual examples, Proc. IEEE 86 (11) (1998) 2196-2209.

[40] T. Vetter, T. Poggio, Linear object classes and image synthesis from a single example image, IEEE Trans. Pattern Analysis and Machine Intelligence 19 (7) (1997) 733-742.

[41] T. Maurer, C. v.d. Malsburg, Learning feature transformations to recognize faces rotated in depth, in: Proc. Int. Conf. Artificial Neural Networks (ICANN), Paris, 1995, pp. 353-358.

[42] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss, The FERET evaluation methodology for face-recognition algorithms, IEEE Trans. Pattern Analysis and Machine Intelligence 22 (10) (2000) 1090-1104.

[43] L-F. Chen, H-Y. Liao, J-C. Lin, C-C. Han, Why recognition in a statistics-based face recognition system should be based on the pure face portion: a probabilistic decision-based proof, Pattern Recognition 34 (7) (2001) 1393-1403.

[44] R.C. Gonzales, R.E. Woods, Digital Image Processing, Addison-Wesley, Reading, Massachusetts, 1992.

[45] C. Sanderson, M. Saban, Y. Gao, On local features for GMM based face verification, in: Proc. 3rd Int. Conf. Information Technology and Applications, Sydney, 2005, Vol. 1, pp. 638-643. http://ieeexplore.ieee.org/xpl/abs_free.jsp?arNumber=1488882

[46] A. Webb, Statistical Pattern Recognition, John Wiley & Sons, UK, 2002.

[47] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Royal Statistical Soc., Ser. B 39 (1) (1977) 1-38.

[48] J.-L. Gauvain, C.-H. Lee, Maximum *a posteriori* estimation for multivariate gaussian mixture observations of Markov chains, IEEE Trans. Speech and Audio Processing 2 (2) (1994) 291-298.

[49] S. Bengio, J. Mariethoz, S. Marcel, Evaluation of biometric technology on XM2VTS, IDIAP Research Report 01-21, Martigny, Switzerland, 2001. http://www.idiap.ch

[50] G.R. Doddington, M.A. Przybycki, A.F. Martin, D.A. Reynolds, The NIST speaker recognition evaluation - Overview, methodology, systems, results, perspective, Speech Communication 31 (2-3) (2000) 225-254.

[51] S. Bengio, J. Mariéthoz, The Expected Performance Curve: a New Assessment Measure for Person Authentication, in: Proc. Odyssey 2004: The Speaker and Language Recognition Workshop, Toledo, 2004, pp. 279-284.

[52] M.J.F. Gales, P.C. Woodland, Variance compensation within the MLLR framework, Technical Report 242, Cambridge University Engineering Department, UK, 1996.

[53] J.A. Rice, Mathematical Statistics and Data Analysis, 2nd ed., Duxbury Press, 1995.

[54] M. Martelli, N.J. Majaj, D.G. Pelli, Are faces processed like words? A diagnostic test for recognition by parts, Journal of Vision 5 (1) (2005) 58-70.

[55] J.A. Bilmes, A gentle tutorial of the EM algorithm and its applications to parameter estimation for gaussian mixture and hidden Markov models, Technical Report TR-97-021, International Computer Science Institute, Berkeley, California, 1998.

About the Authors

Conrad Sanderson received the Bachelor of Engineering (Hons) degree in 1996 and the PhD degree in 2003 from Griffith University, Queensland, Australia. He has worked at the Advanced Telecommunication Research (ATR) Laboratories (Japan), IDIAP Research Institute (Switzerland) and the University of Adelaide node of the Centre for Sensor Signal and Information Processing (CSSIP). He is presently with National ICT Australia (NICTA) and an adjunct fellow at the Australian National University. His current research interests include application and theoretical areas of biometrics and machine learning.

Samy Bengio obtained his PhD in computer science from Universite de Montreal (1993), and spent three post-doctoral years at CNET, the research center of France Telecom, and INRS-Telecommunications (Montreal). He then worked as a researcher for CIRANO, an economic and financial academic research center, applying learning algorithms to finance; he was then a research director at Microcell Labs, a private research center in mobile telecommunications. Since 1999 he is with the IDIAP Research Institute, as a senior researcher in machine learning. His current interests include all theoretical and applied aspects of learning algorithms.

Yongsheng Gao received the BSc and MSc degrees in electronic engineering from Zhejiang University, China, in 1985 and 1988 respectively, and the PhD degree in computer engineering from Nanyang Technological University, Singapore. Currently he is a senior lecturer with the School of Microelectronic Engineering, Griffith University, Australia. His research interests include face recognition, biometrics, image retrieval, computer vision and pattern recognition. He is a member of the IEEE.

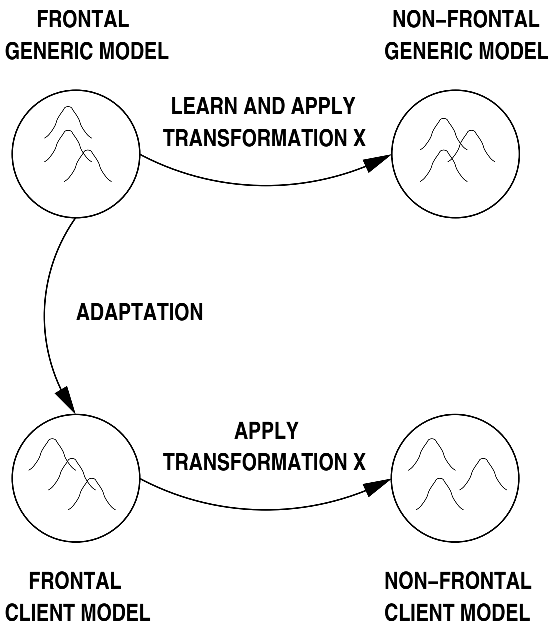

FIG. 1 An interpretation of synthesizing a non-frontal client model based on how the frontal generic model is transformed to a non-frontal generic model.

FIG. 2 Example images from the FERET database for 0^o (frontal), +25^o and +60^o views; note that the angles are approximate.

FIG. 3 Extracted face windows from images in Fig. 2.

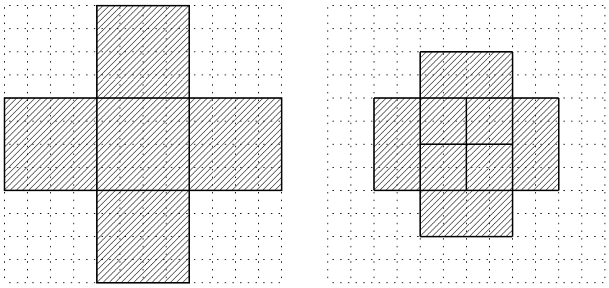

FIG. 4 Graphical example of the spatial area (shaded) used in DCTmod2 feature extraction for N_P=4; left: N_O=0; right: N_O=2.

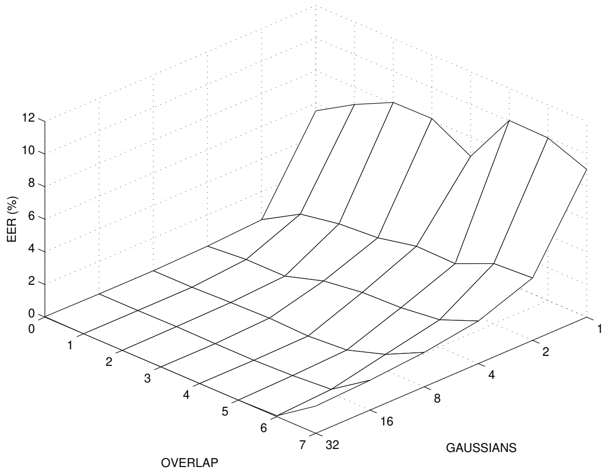

FIG. 5 Performance of the DCT based system trained and tested on frontal faces, for varying degrees of overlap and number of gaussians. Traditional MAP based training was used.

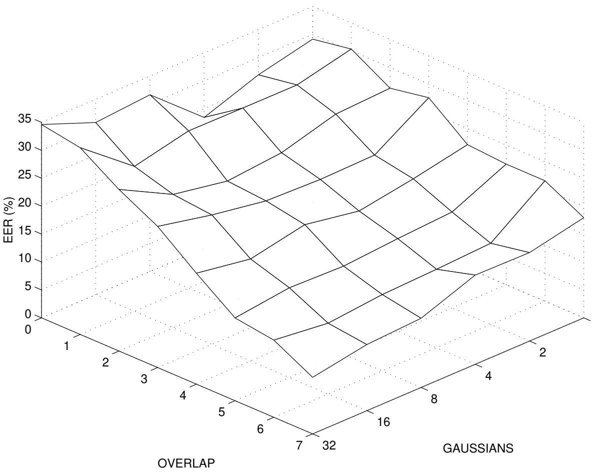

FIG. 6 Performance of the DCT based system trained on frontal faces and tested on +40^o faces, for varying degrees of overlap and number of gaussians. Traditional MAP based training was used.

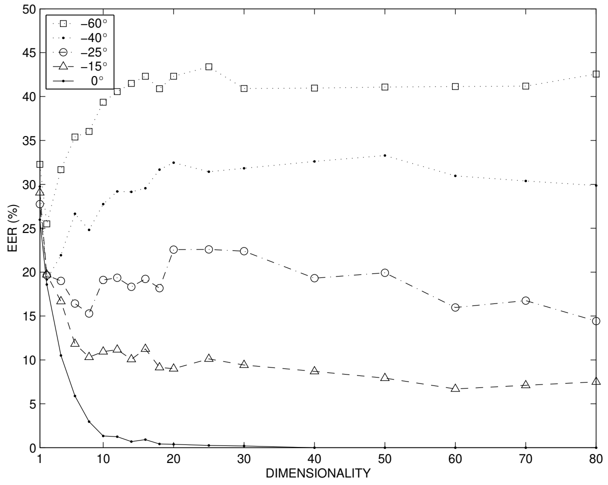

FIG. 7 Performance of PCA based system (trained on frontal faces) for increasing dimensionality and the following angles: -60^o, -40^o, -25^o, -15^o and 0^o (frontal).

Overlap (N_O)	**Vectors** (N_V)	**Spatial width**
0	30	24
1	35	22
2	56	20
3	80	18
4	143	16
5	255	14
6	621	12
7	2585	10

TABLE 1
Number of DCTmod2 feature vectors extracted from a 56×64 face using N_P=8 and varying overlap. It also shows the effective spatial width (& height) in pixels for each feature vector.

Angle	N_R=1	N_R=2	N_R=4	N_R=8	N_R=16	N_R=32
-60^o	23.58	48.83	49.50	49.56	49.94	49.81
-40^o	13.11	49.61	49.58	49.50	49.47	49.56
-25^o	5.81	50.39	49.56	49.56	49.97	49.64
-15^o	1.58	49.83	49.47	49.67	49.75	49.69
$+15^o$	1.28	50.19	49.58	49.61	49.81	49.58
$+25^o$	4.69	50.17	49.67	49.69	49.97	49.56
$+40^o$	9.39	49.25	49.67	49.67	49.64	49.53
$+60^o$	19.53	49.81	49.64	49.81	49.75	49.64

TABLE 2
EER performance of full-MLLR synthesis technique for varying number of regression classes.

Angle	N_R=1	N_R=2	N_R=4	N_R=8	N_R=16	N_R=32
-60^o	23.56	22.69	22.11	18.33	23.67	32.61
-40^o	11.86	11.97	11.14	11.19	15.28	25.17
-25^o	5.25	5.72	4.75	3.86	8.06	16.75
-15^o	1.64	1.58	1.56	1.50	3.53	16.81
$+15^o$	1.36	1.36	1.33	1.36	2.50	15.67
$+25^o$	4.97	4.42	4.36	3.69	5.92	20.72
$+40^o$	8.97	8.33	7.86	8.78	17.14	29.28
$+60^o$	19.81	16.97	16.86	15.31	31.22	31.25

TABLE 3

EER performance of diag-MLLR synthesis technique for varying number of regression classes.

Angle	N_R=1	N_R=2	N_R=4	N_R=8	N_R=16	N_R=32
-60^o	23.31	22.78	22.47	19.67	16.97	17.94
-40^o	12.28	11.00	10.06	10.83	9.25	7.94
-25^o	4.89	5.31	4.64	3.72	3.33	3.44
-15^o	1.58	1.58	1.56	1.53	1.44	1.44
$+15^o$	1.36	1.36	1.33	1.33	1.42	1.42
$+25^o$	4.94	4.67	4.42	3.33	3.08	3.28
$+40^o$	9.00	7.42	7.08	7.42	6.81	6.67
$+60^o$	19.86	18.94	18.81	17.11	15.44	14.33

TABLE 4

EER performance of offset-MLLR synthesis technique for varying number of regression classes.

Angle	**standard** (frontal models)	**full-MLLR** (N_R=1)	**diag-MLLR** (N_R=8)	**offset-MLLR** (N_R=32)
-60^o	22.72	23.58	18.33	$*$ 17.94
-40^o	11.47	13.11	11.19	$*$ 7.94
-25^o	5.72	5.81	3.86	$*$ 3.44
-15^o	2.83	1.58	1.50	$*$ 1.44
$+15^o$	2.64	$*$ 1.28	1.36	1.42
$+25^o$	5.94	4.69	3.69	$*$ 3.28
$+40^o$	10.11	9.39	8.78	$*$ 6.67
$+60^o$	24.72	19.53	15.31	$*$ 14.33

TABLE 5
EER performance for standard frontal models (obtained via traditional MAP based training) and models synthesized for non-frontal angles via MLLR based techniques. Best result for a given angle is indicated by an asterix.

Angle	**frontal**	**full-MLLR**	**diag-MLLR**	**offset-MLLR**	**LinReg**
-60^o	40.97	49.67	50.00	38.56	$*$ 14.92
-40^o	32.61	50.00	49.97	25.75	$*$ 17.19
-25^o	19.31	49.69	49.75	$*$ 13.81	15.78
-15^o	8.69	49.58	49.72	6.86	$*$ 6.44
$+15^o$	10.39	49.67	49.69	8.36	$*$ 5.72
$+25^o$	20.83	49.58	49.97	14.00	$*$ 7.78
$+40^o$	34.36	49.78	50.00	28.97	$*$ 15.00
$+60^o$	44.92	49.83	49.47	38.44	$*$ 14.89

TABLE 6
EER performance comparison between frontal models and synthesized non-frontal models for the PCA based system. Best result for a given angle is indicated by an asterix.

Angle	Frontal	Synth.	Ext.
-60^o	28.22	17.94	18.25
-40^o	15.17	7.94	9.36
-25^o	6.06	3.44	3.28
-15^o	1.61	1.44	1.64
$+15^o$	1.44	1.42	1.67
$+25^o$	5.67	3.28	3.53
$+40^o$	9.39	6.67	5.94
$+60^o$	23.75	14.33	16.56

TABLE 7
EER performance of frontal, synthesized and extended frontal models, DCT-derived features; offset-MLLR based training (frontal models) and synthesis (non-frontal models) was used.

Angle	Frontal	Synth.	Ext.
-60^o	40.97	14.92	15.33
-40^o	32.61	17.19	17.56
-25^o	19.31	15.78	14.94
-15^o	8.69	6.44	9.17
$+15^o$	10.39	5.72	3.67
$+25^o$	20.83	7.78	8.11
$+40^o$	34.36	15.00	15.67
$+60^o$	44.92	14.89	16.08

TABLE 8
EER performance of frontal, synthesized and extended frontal models, PCA features; LinReg model synthesis was used.

Feature type	Model type	
	frontal	extended
PCA	27.34	11.51
DCT	14.82	10.96

TABLE 9
Overall EER performance of frontal and extended frontal models.

noise variance	full-MLLR	diag-MLLR	offset-MLLR
0	-74.81	-74.81	-74.81
1×10^{-7}	-76.51	-74.81	-74.81
1×10^{-6}	-78.76	-74.81	-74.81
1×10^{-5}	-83.34	-74.81	-74.81
1×10^{-4}	-91.63	-74.82	-74.81
1×10^{-3}	-119.95	-74.85	-74.81
1×10^{-2}	-367.01	-75.14	-74.81
1×10^{-1}	-246.57×10^{1}	-75.55	-74.82
1	-313.49×10^{2}	-76.80	-74.92
$1 \times 10^{+1}$	-205.79×10^{3}	-78.29	-75.96
$1 \times 10^{+2}$	-172.71×10^{4}	-84.32	-81.59
$1 \times 10^{+3}$	-283.12×10^{5}	-104.29	-95.81

TABLE 10
Mean of the average log-likelihood [Eqn. (42)] computed using +60^{o} generic model; the +60^{o} generic model was derived from a noise corrupted frontal generic model using a fixed transform (either full-MLLR, diag-MLLR or offset-MLLR).