Extrapolating Single View Face Models for Multi-View Recognition

Conrad Sanderson [1, 2, 3] and Samy Bengio [3]
[1] *Electrical and Electronic Engineering, University of Adelaide, SA 5005, Australia*
[2] *CRC for Sensor Signal and Information Processing, Mawson Lakes, SA 5095, Australia*
[3] *IDIAP Research Institute, Rue du Simplon 4, Martigny, CH-1920, Switzerland*
conradsand@ieee.org, bengio@idiap.ch

Abstract

Performance of face recognition systems can be adversely affected by mismatches between training and test poses, especially when there is only one training image available. We address this problem by extending each statistical 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 utilize prior information on 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: PCA based (holistic features) and DCTmod2 based (local 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 in effect are "single point to single point" transforms). For the DCTmod2 based system, the results show that synthesis via a new MLLR implementation obtains better performance than synthesis based on traditional MLLR (due to a lower number of free parameters). The results further show that extending frontal models considerably reduces errors.

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. Contemporary approaches are able to achieve low error rates when dealing with *frontal* faces (see for example [15]). Previously proposed extensions, in order to handle *non-frontal* faces, generally use training images (for the person to be recognized) at multiple views [12], [16]. 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 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 [1], [3], in this paper we concentrate on extending two face verification systems based on 2D feature extraction (Principal Component Analysis (PCA) based [22] and the recently proposed DCTmod2 method [20]). In both systems we employ a Bayesian classifier which models the distributions via Gaussian Mixture Models (GMMs) [18].

The PCA/GMM system is an extreme example of a holistic system where the spatial relation between face characteristics (such as the eyes and nose) is rigidly kept. Contrarily, the DCTmod2/GMM approach is an extreme example of a local feature approach; here, the spatial relations between face characteristics are largely not utilized (which results in robustness to translations [4]). Examples of systems in between the two extremes include modular PCA [16], Elastic Graph Matching [7], and Pseudo 2D Hidden Markov Models [5].

We address the pose mismatch problem by extending person-specific frontal GMMs with artificially *synthesized* GMMs for non-frontal views. The synthesis of non-frontal models is accomplished via methods based on several implementations of Maximum Likelihood Linear Regression (MLLR) [13] as well as standard multi-variate linear regression (LinReg). MLLR was originally developed for tuning speech recognition systems, and, to our knowledge, this is the first time it is being adapted for face verification.

In the proposed MLLR-based approach, the synthesis and extension is accomplished as follows. Prior information is used to construct several *generic* face models for several views; a generic GMM represents a population of faces and hence does not represent a specific person; in the speaker verification field such generic models are referred to as Universal Background Models (UBMs) [18]. Each non-frontal UBM is obtained by *learning* and *applying* a MLLR-based transformation to the frontal UBM. A person-specific frontal model is obtained via adapting the frontal UBM [18]; a non-frontal person-specific model is then synthesized by applying the previously learned UBM transformation to the person's frontal model. In order for the system to automatically handle the two views, each person's frontal model is then extended by concatenating it with the newly synthesized model; the procedure is then repeated for other views. A graphical interpretation of this procedure shown in Fig. 1.

The LinReg approach is similar to the MLLR-based approach described above. The main difference being that instead of learning the transformation between UBMs, it learns a common relation between two *sets* of feature vectors. The LinReg technique is only utilized for the PCA based system, while the MLLR-based methods are utilized for both PCA and DCTmod2 based systems.

0-7803-8894-1/04/$20.00 © 2004 IEEE

ISSNIP 2004

Previous approaches to addressing single view problems include the synthesis of new *images* at previously unseen views; some examples are optical flow based methods [2], and linear object classes [23]. To handle views for which there is no training data, 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 [12], 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 at various non-frontal views, *without* having access to the person's real images. The synthesis part is somewhat related to [14] where the "jets" in an elastic graph are transformed according to a geometrical 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 the paper is organized as follows. In Section 2 we briefly describe the database for experiments and the preprocessing of images. In Section 3 we overview the DCTmod2 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 DCTmod2 and PCA based 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 the proposed synthesis techniques and the use of extended models. The paper is concluded and future work is suggested in Section 9.

2. SETUP OF THE FACE DATABASE

In our experiments we used images from the *ba*, *bb*, *bc*, *bd*, *be*, *bf*, *bg*, *bh* and *bi* subsets of the FERET database [17], 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; thus for each person there are nine images (see Fig. 2 for examples). The 200 persons were split into three groups: group A, group B and impostor group; the impostor group is comprised of 20 persons, resulting in 90 persons each in groups A and B.

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 (i.e. clients come from group B). Thus in each verification trial there are 90 true claimant accesses and 90×20=1800 impostor attacks; moreover, in each verification trial the view of impostor faces matched the testing view (this restriction is relaxed later).

In order to reduce the effects of facial expressions and hair styles, closely cropped faces are used [6]; 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 exactly 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 not only in terms of out-of-plane rotation of the face, but also scale. Example face windows are shown in Fig. 3.

3. FEATURE EXTRACTION

A. DCTmod2 Based System

In DCTmod2 feature extraction 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 2D Discrete Cosine Transform (DCT) basis functions [20]. A feature vector for each block is then constructed as:

$$\mathbf{x} = \left[\begin{array}{cccccccccc} \Delta^h c_0 & \Delta^v c_0 & \Delta^h c_1 & \Delta^v c_1 & \Delta^h c_2 & \Delta^v c_2 & c_3 & c_4 & \ldots & c_{M-1} \end{array} \right]^T \quad (1)$$

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 traditional DCT feature extraction, the first three DCT coefficients are replaced by their respective deltas in order to reduce the effects of illumination changes, without losing discriminative information. In this study we use M=15 (based on [20]), resulting in an 18 dimensional feature vector for each block. The overlap is set to N_O=7 resulting in 2585 vectors for each 56×64 face window; the choice of overlap is based on [21], where it was shown that the larger the overlap, the more robust the system is to out-of-plane rotations.

B. PCA Based System

In PCA based feature extraction [8], [22], 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:

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

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, while in the DCTmod2 approach one feature vector represents only a small portion of the face.

4. GMM BASED CLASSIFIER

The distribution of training feature vectors for each person is modeled by a GMM. Given a claim for client C's identity and a set of (test) feature vectors $X = \{\mathbf{x}_i\}_{i=1}^{N_V}$ supporting the claim, the likelihood of the claimant being the true claimant is found with:

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

where $P(\mathbf{x}|\lambda) = \sum_{g=1}^{N_G} w_g \, \mathcal{N}(\mathbf{x}|\mu_g, \mathbf{\Sigma}_g)$ and $\lambda = \{w_g, \mu_g, \mathbf{\Sigma}_g\}_{g=1}^{N_G}$. $\mathcal{N}(\mathbf{x}|\mu, \mathbf{\Sigma})$ is in turn a D-dimensional gaussian function with mean μ and diagonal covariance matrix $\mathbf{\Sigma}$ [8], [18]. 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 $\sum_{g=1}^{N_G} w_g = 1$ and

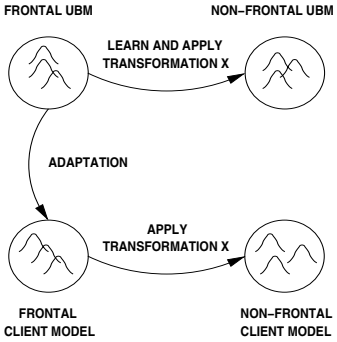

Fig. 1: Graphical interpretation of synthesizing a non-frontal client model based on how the frontal UBM is transformed to a non-frontal UBM.

Fig. 2: Example images from the FERET database for 0^o, $+25^o$ and $+60^o$ views (left to right); note that the angles are approximate.

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

$\forall\ g : w_g \geq 0$). Note that in (3) each feature vector is treated as being independent and identically distributed (iid).

Given the likelihood of the claimant being an impostor, $P(X|\lambda_I)$, an opinion on the claim is found using:

$$\mathcal{O}(X) = \log P(X|\lambda_C) - \log P(X|\lambda_I) \tag{4}$$

The verification decision is reached as follows: given a threshold t, the claim is accepted when $\mathcal{O}(X) \geq t$ and rejected when $\mathcal{O}(X) < t$. We use a global threshold (common across all clients) tuned to obtain the lowest Equal Error Rate (EER) (where the false rejection rate equals the false acceptance rate) on test data, following the approach often used in speaker verification [9], [11].

Methods for obtaining the parameter set for the impostor model (λ_I) and each client model are described in the following sections.

A. Classifier Training: DCTmod2 Based System

First, a Universal Background Model (UBM) is trained using the Expectation Maximization (EM) algorithm [8], using all 0^o data from group A. Since the UBM is a good representation of a general face, it is also used to find the likelihood of the claimant being an impostor, i.e.:

$$P(X|\lambda_I) = P(X|\lambda_{\texttt{UBM}}) \tag{5}$$

The parameters (λ) for each client model are then found by using the client's training data and adapting the UBM; the adaptation is usually done using Maximum *a Posteriori* (MAP) estimation [5], [18]. In this work we shall also utilize three other adaptation techniques, all based on MLLR (described in Section 5). The choice of the adaptation technique depends on the non-frontal model synthesis method utilized later (Section 6).

B. Classifier Training: PCA Based System

The image subset from 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 UBM and all client models (for frontal faces) are constrained to have only one component (i.e. one gaussian), with a diagonal covariance matrix. The mean and covariance matrix of the UBM is taken to be the mean and covariance matrix of feature vectors from group A. Instead of adaptation (as done in the DCTmod2 based system, above), each client model inherits the covariance matrix from the UBM; moreover, the mean of each client model is taken to be the single training vector for that client.

5. MAXIMUM LIKELIHOOD LINEAR REGRESSION

In the MLLR framework [10], [13], the adaptation of a given model is performed in two steps; first the means are updated followed by an update of the covariance matrices, such that:

$$P(X|\widetilde{\lambda}) \geq P(X|\widehat{\lambda}) \geq P(X|\lambda) \tag{6}$$

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

A. Adaptation of Means

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

$$\widehat{\mu}_g = \mathbf{W}_S\ \nu_g \tag{7}$$

where $\nu_g = [\ 1\ \mu_g^T\ ]^T$ and $\mathbf{W}_S$ is a $D \times (D+1)$ 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-C below), the general form for finding $\mathbf{W}_S$ is:

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

where

$$P(g|\mathbf{x}_i,\lambda) = \{w_g\mathcal{N}(\mathbf{x}_i|\mu_g,\mathbf{\Sigma}_g)\}\ /\ \textstyle\sum_{n=1}^{N_G} w_n\mathcal{N}(\mathbf{x}_i|\mu_n,\mathbf{\Sigma}_n) \tag{8}$$

As further elucidation is quite tedious, the reader is referred to [13] for the full solution of $\mathbf{W}_S$. Two forms of $\mathbf{W}_S$ were originally proposed: full or "diagonal" [13], which we shall refer to as full-MLLR and diag-MLLR, respectively. We propose a third form of MLLR, where the transformation matrix is modified so that transforming each mean is equivalent to:

$$\widehat{\mu}_g = \mu_g + \mathbf{\Delta}_S \tag{9}$$

where $\mathbf{\Delta}_S$ maximizes the likelihood of given training data. Using the EM framework leads to the following solution (we provide the derivation in [21]):

$$\mathbf{\Delta}_S = \left[\sum_{r=1}^{N_S}\sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i,\lambda)\mathbf{\Sigma}_{g_r}^{-1}\right]^{-1}\left[\sum_{r=1}^{N_S}\sum_{i=1}^{N_V} P(g_r|\mathbf{x}_i,\lambda)\mathbf{\Sigma}_{g_r}^{-1}(\mathbf{x}_i - \mu_{g_r})\right] \tag{10}$$

where $\mathbf{\Delta}_S$ is shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$. We shall refer to this form of MLLR as *offset-MLLR*.

B. Adaptation of Covariance Matrices

Once the new means are obtained, each new covariance matrix is found using [10]:

$$\widetilde{\mathbf{\Sigma}}_g = \mathbf{B}_g^T\mathbf{H}_S\mathbf{B}_g \tag{11}$$

where $\mathbf{B}_g = \mathbf{C}_g^{-1}$ and $\mathbf{C}_g\mathbf{C}_g^T = \mathbf{\Sigma}_g^{-1}$; the latter equation is a form of Cholesky decomposition [19]. $\mathbf{H}_S$, shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$, is found with:

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

The covariance transformation may be either full or diagonal. When full transformation is used, full covariance matrices are produced even if the original covariances were diagonal to begin with. To avoid this, the off-diagonal elements of $\mathbf{H}_S$ are set to zero.

C. Regression Classes

If each gaussian has its own mean and covariance transformation matrices, then for full-MLLR there are $D\times(D+1)+D=D^2+2D$ parameters to estimate per gaussian (where D is the dimensionality); for diag-MLLR there are $D+D+D=3D$ parameters per gaussian, and for offset-MLLR there are $D+D=2D$ parameters per gaussian.

Ideally each mean and covariance matrix in a GMM will have its own transform, however in practical applications there may not be enough training data to reliably estimate the required number of parameters. One way of working around the lack of data is to share a transform across two or more gaussians. We define which gaussians are to share a transform by clustering the gaussians based on the distance between their means.

Let us 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 non-frontal Models

A. DCTmod Based System

In the MLLR based model synthesis technique, we first transform, using prior data, the frontal UBM into a non-frontal UBM for angle Θ. To synthesize a client model for angle Θ, the previously learned transformations are applied to the client's frontal model. Moreover, each frontal client model is derived from the frontal UBM by MLLR (to ensure correspondence between models), instead of using MAP adaptation.

B. PCA Based System

For the PCA based system, we shall 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 appropriate non-frontal UBMs Moreover, as each client model has only one gaussian, we note that the MLLR based transformations are "single point to single point" transformations, where the points are the old and new mean vectors.

As described in Section 4-B, the mean of each client model is taken to be the single training vector available; thus in this case transformation in the feature domain is equivalent to 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:

$$\mathbf{B} = \mathbf{AW} \tag{12}$$

$$\begin{bmatrix} \mathbf{b}_1^T \\ \mathbf{b}_2^T \\ \vdots \\ \mathbf{b}_N^T \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{a}_1^T \\ 1 & \mathbf{a}_2^T \\ \vdots & \vdots \\ 1 & \mathbf{a}_N^T \end{bmatrix} \begin{bmatrix} w_{1,1} & \cdots & w_{1,D} \\ w_{2,1} & \cdots & w_{2,D} \\ \vdots & \vdots & \vdots \\ w_{D+1,1} & \cdots & w_{D+1,D} \end{bmatrix} \tag{13}$$

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 [19]: $\mathbf{W} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{B}$. 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:

$$\mu^{\Theta} = \begin{bmatrix} 1 & (\mu^{0^o})^T \end{bmatrix} \mathbf{W} \tag{14}$$

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 UBM is also extended with non-frontal UBMs. Formally, an extended model is created using:

$$\lambda_C^{\texttt{extended}} = \lambda_C^{0^o} \sqcup \lambda_C^{+60^o} \cdots \sqcup \lambda_C^{-40^o} \sqcup \lambda_C^{-60^o} = \sqcup_{i\in\Phi} \lambda_C^i$$

where $\lambda_C^{0^o}$ is the client's frontal model and Φ is a set of angles, e.g. $\Phi = \{\ 0^o, +60^o, +40^o, +25^o, +15^o, -15^o, -25^o, -40^o, -60^o\ \}$. $\sqcup$ 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:

$$\lambda^z = \lambda^x \sqcup \lambda^y = \{\alpha w_g^x,\ \mu_g^x,\ \mathbf{\Sigma}_g^x\}_{g=1}^{N_G^x} \cup \{\beta w_g^y,\ \mu_g^y,\ \mathbf{\Sigma}_g^y\}_{g=1}^{N_G^y}$$

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

8. Experiments and Discussion

A. DCTmod2 Based System

Based on [21], the number of gaussians for each client model was set to 32. The performance of non-frontal models synthesized via the full-MLLR, diag-MLLR and offset-MLLR techniques is shown in Table 1; for the latter three methods varying number of regression classes was used, however due to lack of space only the results for the optimal number of regression classes are shown.

The full-MLLR technique falls apart when there is two or more regression classes; its best results (obtained for one regression class) are in some cases worse than for standard frontal models. We believe the poor results are due to not enough training data available to properly estimate the transformation matrices (recall that the full-MLLR technique has more free parameters than diag-MLLR and offset-MLLR). The full-MLLR transformation is adequate for adapting the frontal UBM to frontal client models (as evidenced by the 0% EER), suggesting that the transformation is only reliable when applied to the specific model it was trained to transform. A further investigation of the sensitivity of the MLLR transform is presented in [21].

Compared to full-MLLR, the diag-MLLR technique has better performance characteristics; this is expected, as the number of transformation parameters is significantly less than 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 number of regression classes. The results are consistent with the scenario that once the number of regression classes reaches a certain threshold, there is not enough training data 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 has the best performance characteristics when compared to full-MLLR and diag-MLLR; it

TABLE 1: EER PERFORMANCE OF STANDARD (FRONTAL) AND SYNTHESIZED MODELS (**DCTmod2 features**); STANDARD MODELS USED TRADITIONAL MAP TRAINING.

Angle	standard (frontal)	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
0^o	1.67	$*$ 0.00	$*$ 0.00	$*$ 0.00
$+15^o$	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 2: EER PERFORMANCE COMPARISON BETWEEN FRONTAL MODELS AND SYNTHESIZED NON-FRONTAL MODELS FOR **PCA based system**.

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
0^o	0.00	0.00	0.00	0.00	0.00
$+15^o$	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

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

B. PCA Based System

Based on [21], the dimensionality for PCA derived feature vectors was set to 40. The performance of models synthesized using full-MLLR, diag-MLLR, offset-MLLR and LinReg techniques is shown in Table 2. As there is only one gaussian per client model, there was only one regression class for MLLR based techniques.

Results in Table 2 further 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 to full-MLLR, we attribute this to the sensitivity of the transformation techniques, described in [21]. The best results were obtained with the LinReg technique, supporting the view that "single point to single point" type transformations (such as MLLR) are not suitable for a system utilizing PCA derived features.

Lastly, we note that the standard PCA based system is significantly more affected by view changes than the standard DCTmod2 based system. This can be attributed to the rigid preservation of spatial relations between face areas, which is in contrast to the local feature approach, where each feature describes only a part of the face. The combination of local features and the GMM classifier causes most of the spatial relation information to be lost; this in effect allows for movement of facial areas (which occur due out-of-plane rotations).

C. Performance of Extended Frontal Models

In the experiments described in Sections 8-A and 8-B, 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 augmenting 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 DCTmod2 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 no significant performance benefit over the 0^o model; the frontal UBM was also extended with non-frontal UBMs. Since each frontal model had 32 gaussians, each resulting extended model had 224 gaussians. Following the offset-MLLR based model synthesis paradigm, each frontal client model was derived from the frontal UBM 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 UBM was also extended with non-frontal UBMs. Since each frontal model had one gaussian, each resulting extended model had nine gaussians. As can be seen in Tables 3 and 4, for most angles only a small reduction in performance is observed when compared to models synthesized for a specific angle (implying that pose detection may not be necessary).

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 DCTmod2 and PCA based systems two types of models were used: frontal and extended. For the DCTmod2 based system frontal models were derived from the UBM using offset-MLLR.

From the results presented in Table 5, it can be observed that model extension reduces the error rate in both PCA and DCTmod2 based systems, with the DCTmod2 system achieving the lowest EER. The largest error reduction is present in the PCA based system, where the EER is reduced by 58%; for the DCTmod2 based system, the EER is reduced by 26%. These results thus support the use of extended frontal models.

TABLE 3: EER PERFORMANCE OF FRONTAL, SYNTHESIZED AND EXTENDED FRONTAL MODELS, DCTMOD2 FEATURES; OFFSET-MLLR BASED TRAINING AND SYNTHESIS WAS USED.

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
0^o	0.00	0.00	0.00
$+15^o$	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 4: EER PERFORMANCE OF FRONTAL, SYNTHESIZED AND EXTENDED FRONTAL MODELS, PCA FEATURES; LINREG MODEL SYNTHESIS 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
0^o	0.00	0.00	0.28
$+15^o$	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 5: OVERALL EER PERFORMANCE OF FRONTAL AND EXTENDED FRONTAL MODELS.

Feature type	Model type	
	frontal	extended
PCA	27.34	11.51
DCTmod2	14.82	10.96

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) and standard multi-variate linear regression (LinReg). The 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 was evaluated on two face verification systems: PCA based (holistic features) and DCTmod2 based (local 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 in effect are "single point to single point" transforms in the PCA based system); for the DCTmod2 based system, the results show that synthesis via a new MLLR implementation obtains better performance than synthesis based on traditional MLLR (due to a lower number of free parameters). The results further suggest that extending frontal models considerably reduces errors.

The results also show that the standard DCTmod2 based system (trained on frontal faces) is less affected by out-of-plane rotations than the corresponding 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 movement of facial areas (which occur due out-of-plane rotations). This is in contrast to the PCA based system, where, due to the holistic representation, the spatial relations are rigidly kept.

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 data. A related question is how many discrete views are necessary to adequately cover all poses. The dimensionality reduction matrix $\mathbf{U}$ in the PCA approach was defined using only frontal faces; higher performance may be obtained by also incorporating non-frontal faces. The DCTmod2/GMM approach can be extended by embedding positional information into each feature vector [5], 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 an out-of-plane rotation. Lastly, it would be useful to evaluate alternative size normalization approaches in order to address the scaling problem mentioned in Section 2.

ACKNOWLEDGEMENTS

The authors thank the Swiss National Science Foundation (SNSF) for supporting this work via the National Center of Competence in Research on Interactive Multimodal Information Management (IM2). The authors also thank Yongsheng Gao (Griffith University) for useful suggestions.

REFERENCES

[1] 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*, Vol. 8, 1996, pp. 1321-1340.

[2] D. Beymer, T. Poggio, "Face Recognition From One Example View", In: *Proc. 5th Int. Conf. Computer Vision*, Cambridge, 1995, pp. 500-507.

[3] 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*, Washington, D.C., 2002.

[4] 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*, Guildford, 2003, pp. 911-920.

[5] F. Cardinaux, C. Sanderson, S. Bengio, "Face Verification Using Adapted Generative Models", In: *Proc. 6th IEEE Int. Conf. Automatic Face and Gesture Recognition*, Seoul, 2004, pp. 825-830.

[6] L-F. Chen et al., "Why recognition in a statistics-based face recognition system should be based on the pure face portion: a probabilistic decision-based proof", *Pattern Recognition*, Vol. 34, No. 7, 2001, pp. 1393-1403.

[7] B. Duc, S. Fischer, J. Bigün, "Face Authentication with Gabor Information on Deformable Graphs", *IEEE Trans. Image Processing*, Vol. 8, No. 4, 1999, pp. 504-516.

[8] R. Duda, P. Hart, D. Stork, *Pattern Classification*, John Wiley & Sons, USA, 2001.

[9] S. Furui, "Recent Advances in Speaker Recognition", *Pattern Recognition Letters*, Vol. 18, No. 9, 1997, pp. 859-872.

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

[11] J. Ortega-Garcia, J. Bigun, D. Reynolds, J. Gonzales-Rodriguez, "Authentication gets personal with biometrics", *IEEE Signal Processing Magazine* Vol. 21, No. 2, 2004, pp. 50-62.

[12] R. Gross, J. Yang, A. Waibel, "Growing Gaussian Mixture Models for Pose Invariant Face Recognition", In: *Proc. 15th Int. Conf. Pattern Recognition*, Barcelona, 2000, Vol. 1, pp. 1088-1091.

[13] C.J. Leggetter, P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models", *Computer Speech and Language*, Vol. 9, No. 2, 1995, pp. 171-185.

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

[15] K. Messer et al., "Face Verification Competition on the XM2VTS Database", In: *Proc. 4th Int. Conf. Audio- and Video-Based Biometric Person Authentication*, Guildford, 2003, pp. 964-974.

[16] A. Pentland, B. Moghaddam, T. Starner, "View-Based and Modular Eigenspaces for Face Recognition", In: *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Seattle, 1994, pp. 84-91.

[17] 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*, Vol. 22, No. 10, 2000, pp. 1090-1104.

[18] D. Reynolds, T. Quatieri, R. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models", *Digital Signal Processing*, Vol. 10, No. 1-3, 2000, pp. 19-41.

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

[20] C. Sanderson, K.K. Paliwal, "Fast features for face authentication under illumination direction changes", *Pattern Recognition Letters*, Vol. 24, No. 14, 2003, pp. 2409-2419.

[21] C. Sanderson, S. Bengio, "Statistical Transformation Techniques for Face Verification Using Faces Rotated in Depth", IDIAP Research Report 04-04, Martigny, Switzerland, 2004.

[22] M. Turk, A. Pentland, "Eigenfaces for Recognition", *J. Cognitive Neuroscience*, Vol. 3, No. 1, 1991, pp. 71-86.

[23] T. Vetter, T. Poggio, "Linear object classes and image synthesis from a single example image", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, 1997, pp. 733-742.