STATISTICAL TRANSFORMATIONS OF FRONTAL MODELS FOR NON-FRONTAL FACE VERIFICATION

Conrad Sanderson [(a)(b)(c)] *and Samy Bengio* [(a)]

[(a)] IDIAP Research Institute, Rue du Simplon 4, CH-1920 Martigny, Switzerland
[(b)] Electrical and Electronic Engineering, University of Adelaide, SA 5005, Australia
[(c)] CRC for Sensor Signal and Information Processing, Mawson Lakes, SA 5095, Australia
conradsand @ ieee.org, bengio @ idiap.ch

ABSTRACT

In the framework of a face verification system using local features and a Gaussian Mixture Model based classifier, we address the problem of non-frontal face verification (when only a single (frontal) training image is available) by *extending* each client's frontal face model with artificially *synthesized* models for non-frontal views. Furthermore, we propose the Maximum Likelihood Shift (MLS) synthesis technique and compare its performance against a Maximum Likelihood Linear Regression (MLLR) based technique (originally developed for adapting speech recognition systems) and the recently proposed "difference between two Universal Background Models" (UBMdiff) technique. All techniques rely on prior information and learn how a generic face model for the frontal view is related to generic models at non-frontal views. Experiments on the FERET database suggest that that the proposed MLS technique is more suitable than MLLR (due to a lower number of free parameters) and UBMdiff (due to lack of heuristics). The results further suggest that extending frontal models considerably reduces errors.

1. INTRODUCTION

Contemporary approaches to face recognition (here we mean both identification and verification) are able to achieve quite low error rates when dealing with *frontal* faces (see for example [12]). A more realistic and challenging task is to verify a face at a non-frontal view when only one (frontal) training image is available (e.g. a passport photograph).

Whereas the task of view-independent recognition has been addressed through the use of training images (for the person to be recognized) at multiple views [10, 14], the much harder task of using only one training image has received relatively little attention. While it is possible to use 3D approaches to address the single training image problem [1, 3], here we concentrate on extending a local feature approach which utilizes a Gaussian Mixture Model (GMM) based classifier [16] (an instance of a Bayesian classifier [7]); we shall utilize DCTmod2 features [18].

It has been previously shown that compared to systems based on holistic face representations (such as Principal Component Analysis (PCA) derived features [7]), the DCTmod2/GMM approach is more robust to out-of-plane rotations [19] and to imperfectly located faces [4]; this is mainly due to the spatial relations between facial characteristics being inherently less constrained, permitting faces to undergo a degree of translation and deformation.

Generally speaking, an appearance based face recognition system can be thought of as being comprised of:

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

The authors thank the Swiss National Science Foundation for supporting this work through the National Center of Competence in Research (NCCR) on Interactive Multimodal Information Management (IM2).

The first stage usually provides a size normalized face image (with eyes at fixed locations); illumination normalization may also be performed. In this work we shall concentrate on the last stage (and thus postulate that the preceding steps have been performed correctly).

Some approaches to addressing the single training image problem involve the synthesis of new face *images* (at various angles) based on prior information [2, 13]. In these approaches, the image synthesis comes before the usual step of feature extraction. A question thus arises: if we are only interested in recognition (and hence we are going to extract features from synthesized images), why not synthesize the features instead? Following this line of thinking, the next question is: instead of synthesizing features with which we are going to train a classifier, why not directly synthesize the classifier's parameters? This is the central idea of our proposed extensions, sketched below.

Using prior information in the form of a set of faces at different views (faces which will never be used during testing), we construct *generic* (non-person specific) face GMMs for specific views; these generic models are referred to as Universal Background Models (UBMs) in the speaker verification field [16]. Each *non-frontal* UBM is constructed by learning and applying a transformation to the *frontal* UBM. Let us now suppose that we wish to enroll a new client in our face verification system and we only have their frontal view; a frontal client model is obtained via the usual approach of adapting the frontal UBM [16]; a non-frontal client model is then synthesized by applying the previously learned UBM transformation to the client's frontal model. Fig. 1 shows a graphical interpretation of this procedure. In order for the system to automatically handle the two views, we then extend the client's frontal model by concatenating it with the newly synthesized model; the procedure is then repeated for other views. The frontal UBM is also extended with non-frontal UBMs.

Under this general synthesis and model extension framework, we propose the Maximum Likelihood Shift (MLS) model transformation technique; we also adapt Maximum Likelihood Linear Regression (MLLR), which was previously used for transforming speech recognition systems to be more user specific [11]; to our knowledge this is the first time MLLR is being utilized for face verification. The MLS and MLLR based techniques are compared against a previously proposed technique referred to as UBMdiff [19].

The rest of the paper is organized as follows. In Sec. 2 we briefly describe the database used in the experiments and the pre-processing of images. In Sec. 3 we overview the DCTmod2 feature extraction technique. Sec. 4 provides a concise description of the GMM based classifier. In Sec. 5 we summarize MLLR, while in Section 6 we propose the MLS technique. In Sec. 7 we describe model synthesis techniques based on MLLR, MLS and UBMdiff. Sec. 8 details the process of extending a frontal model with synthesized non-frontal models. Sec. 9 is devoted to experiments evaluating the synthesis techniques and the use of extended models. The paper is concluded and future work is suggested in Sec. 10.

Appears in: *Proc. IEEE Int. Conf. Image Processing*, Singapore, 2004.

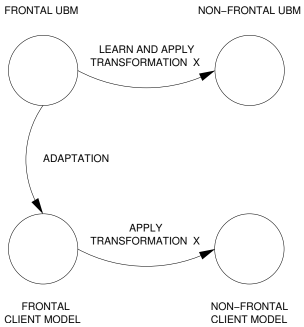

Fig. 1. Graphical interpretation of synthesizing a non-frontal client model based on how the frontal UBM is transformed to a non-frontal UBM.

Fig. 2. Images of subject 01169 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.

2. FERET DATABASE: SETUP & PRE-PROCESSING

In our experiments we used images from the *ba*, *bb*, *bc*, *bd*, *be*, *bf*, *bg*, *bh* and *bi* subsets of the FERET database [15], 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 disjoint groups: group A, group B and impostor group; the impostor group is comprised of 20 persons, resulting in 90 persons 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 variations possible in real life (such as facial expressions and hair styles) closely cropped faces are used [6]. Since in this paper we are proposing extensions to an existing 2D approach, 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.

3. FEATURE EXTRACTION

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 neighboring blocks by N_O pixels; each block is decomposed in terms of 2D Discrete Cosine Transform (DCT) basis functions [18]. A feature vector for each block is then constructed as:

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

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 neighboring 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 [18]), resulting in an 18 dimensional feature vector for each block. The overlap is set to N_O=7 resulting in 2585 vectors[1] for each 56×64 (rows $\times$ columns) face window; the choice of overlap is based on [20], where it was shown that the larger the overlap, the more robust the system is to out-of-plane rotations.

[1] in contrast to PCA based feature extraction, which results in 1 vector for each 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:

$$P(X|\lambda_C) = \prod\nolimits_{i=1}^{N_V} P(\mathbf{x}_i|\lambda_C) \tag{2}$$

where $P(\mathbf{x}|\lambda) = \sum_{g=1}^{N_G} w_g \, \mathcal{N}(\mathbf{x}|\mu_g, \mathbf{\Sigma}_g)$ and $\lambda = \{w_g, \mu_g, \mathbf{\Sigma}_g\}_{g=1}^{N_G}$. Furthermore, $\mathcal{N}(\mathbf{x}|\mu, \mathbf{\Sigma})$ is a D-dimensional gaussian function with mean μ and diagonal covariance matrix $\mathbf{\Sigma}$ [7, 16]; 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 $\forall \, g : w_g \geq 0$).

Given the likelihood of the claimant being an impostor, $P(X|\lambda_{\overline{C}})$, an opinion on the claim is found using:

$$\mathcal{O}(X) = \log P(X|\lambda_C) - \log P(X|\lambda_{\overline{C}}) \tag{3}$$

The verification decision is reached as follows: given a threshold t, the claim is accepted when $\mathcal{O}(X) \geq t$ and rejected when $\mathcal{O}(X) < t$. In our experiments we use a global threshold to obtain performance as close as possible to the Equal Error Rate (EER) (i.e. where the false rejection rate equals the false acceptance rate), following the popular practice used in the speaker verification field [8].

4.1. Classifier Training

First, a Universal Background Model (UBM) is trained using the Expectation Maximization (EM) algorithm [7], 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.:

$$P(X|\lambda_{\overline{C}}) = P(X|\lambda_{ubm}) \tag{4}$$

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, 16]. In this work we shall also utilize two other adaptation techniques, namely MLLR and the proposed MLS, described in Sections 5 and 6, respectively. The choice of the adaptation technique depends on the non-frontal model synthesis method utilized later (Section 7).

5. MAXIMUM LIKELIHOOD LINEAR REGRESSION

In the MLLR framework [9, 11], 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:

$$P(X|\widetilde{\lambda}) \geq P(X|\widehat{\lambda}) \geq P(X|\lambda) \tag{5}$$

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.

5.1. Adaptation of Means

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

$$\widehat{\mu}_g = \mathbf{W}_S \, \nu_g \tag{6}$$

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 5.3 below), the general form for finding $\mathbf{W}_S$ is:

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

where

$$P(g|\mathbf{x}_i, \lambda) = \{w_g \mathcal{N}(\mathbf{x}_i|\mu_g, \mathbf{\Sigma}_g)\} \ / \ \sum\nolimits_{n=1}^{N_G} w_n \mathcal{N}(\mathbf{x}_i|\mu_n, \mathbf{\Sigma}_n) \tag{7}$$

As further elucidation is quite tedious, the reader is referred to [11] for the full solution of $\mathbf{W}_S$. There are two possible forms of $\mathbf{W}_S$: full or "diagonal" [11], which we shall refer to as full-MLLR and diag-MLLR, respectively.

5.2. Adaptation of Covariance Matrices

Once the new means are obtained, each new covariance matrix is found using [9]:

$$\widetilde{\mathbf{\Sigma}}_g = \mathbf{B}_g^T \mathbf{H}_S \mathbf{B}_g \tag{8}$$

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 [17]. $\mathbf{H}_S$, shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$, is found with:

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

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; since in this work we restrict ourselves to the use of diagonal covariance matrices, we force the transform to be diagonal by setting the off-diagonal elements of $\mathbf{H}_S$ to zero.

5.3. Regression Classes

If each gaussian has its own mean and covariance transformation matrices, then for full-MLLR there is $D \times (D+1) + D = D^2 + 2D$ parameters to estimate per gaussian (where D is the dimensionality); for diag-MLLR, there is $D + D + D = 3D$ parameters to estimate 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 Mahalanobis distance [7] 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); we denote the number of regression classes as N_R.

6. PROPOSED MAXIMUM LIKELIHOOD SHIFT

In the Maximum Likelihood Shift (MLS) approach, we re-define each mean as:

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

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 [20]):

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

where $\mathbf{\Delta}_S$ is shared by N_S gaussians $\{g_r\}_{r=1}^{N_S}$. The covariance matrices are then updated as described in Section 5.2. We note that for each regression class $D + D = 2D$ parameters need to be estimated (compared to $D^2 + 2D$ for full-MLLR and $3D$ for diag-MLLR).

7. SYNTHESIZING NON-FRONTAL MODELS

In the MLLR based model synthesis technique, we first transform, using prior data, the frontal UBM into a non-frontal UBM for angle Θ. The parameters which describe the transformation of the means and covariances are $\Psi_{\text{MLLR}} = \{\mathbf{W}_g, \mathbf{H}_g\}_{g=1}^{N_G}$, where $\mathbf{W}_g$ 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. Specifically, the means and covariances of the client's model for Θ are synthesized using Eqns. (6) and (8), respectively; the weights are kept the same as for the 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.

Angle	standard (frontal)	UBMdiff	full-MLLR (N_R=1)	diag-MLLR (N_R=8)	MLS (N_R=32)
-60^o	22.72	18.19	23.58	18.33	∗ 17.94
-40^o	11.47	∗ 6.00	13.11	11.19	7.94
-25^o	5.72	∗ 3.44	5.81	3.86	∗ 3.44
-15^o	2.83	1.83	1.58	1.50	∗ 1.44
0^o	1.67	0.08	∗ 0.00	∗ 0.00	∗ 0.00
$+15^o$	2.64	2.39	∗ 1.28	1.36	1.42
$+25^o$	5.94	∗ 3.28	4.69	3.69	∗ 3.28
$+40^o$	10.11	6.86	9.39	8.78	∗ 6.67
$+60^o$	24.72	20.33	19.53	15.31	∗ 14.33

Table 1. EER performance of standard (frontal) and synthesized models; standard models used traditional MAP training; best result for a given angle is indicated by an asterix.

The MLS based synthesis technique is similar to the MLLR based technique; the main difference being that instead of using $\mathbf{W}_g$ to transform the mean of the g-th gaussian, $\mathbf{\Delta}_g$ is used. Specifically, the parameters which describe the transformation of the frontal UBM to a non-frontal UBM are: $\Psi_{\text{MLS}} = \{\mathbf{\Delta}_g, \mathbf{H}_g\}_{g=1}^{N_G}$ where $\mathbf{\Delta}_g$ and $\mathbf{H}_g$ are found as described in Sections 6 and 5.2, respectively. A client's model for angle Θ is then synthesized by transforming the client's frontal model using Eqns. (9) and (8).

In the previously proposed UBMdiff technique [19], the differences between the frontal UBM and a non-frontal UBM are found (i.e. how the means have changed and how the diagonal covariance matrix elements have scaled); a non-frontal client model was then created by applying the differences to the corresponding frontal client model. While effective, the method requires a heuristic modification to the MAP training algorithm to ensure correspondence between the UBMs and the frontal client models.

8. EXTENDING FRONTAL MODELS

In order for the system to automatically handle non-frontal views, we propose to extend each client's frontal model by synthesized non-frontal models (Section 7); such an extended model for client C is created as follows:

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

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

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

where $\alpha = N_G^x / (N_G^x + N_G^y)$ and $\beta = 1 - \alpha$. It must be noted that the frontal UBM is also extended with non-frontal UBMs.

9. EXPERIMENTS AND DISCUSSION

9.1. Performance of Synthesized Non-Frontal Models

Based on [19], the number of gaussians for each client model was set to 32. The performance of non-frontal models synthesized via the UBMdiff, full-MLLR, diag-MLLR and MLS 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 MLS). The full-MLLR transformation is adequate for adapting the

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 2. EER performance of frontal, synthesized and extended frontal models; MLS-based training and synthesis was used.

Model type	EER
frontal	14.82
extended	10.96

Table 3. Overall EER performance of frontal and extended frontal models; MLS-based training and synthesis was used.

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 [20].

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 MLS technique has the best performance characteristics 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 performance, obtained at 32 regression classes, is for all angles better than the performance of standard frontal models.

By comparing all four synthesis techniques it can be observed that the MLS technique obtains the best overall performance. It is also arguably the easiest to implement and avoids the heuristics (modified MAP estimation) required in the UBMdiff technique.

9.2. Performance of Extended Frontal Models

In the experiments described in Section 9.1, 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 frontal models with models synthesized for a specific angle; impostor faces matched the test view. Each client's frontal model was extended with models synthesized by the MLS 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 extended model had 224 gaussians. Following the MLS-based model synthesis paradigm (Section 7), each frontal client model was derived from the frontal UBM using MLS. As can be seen in Table 2, for most angles extended frontal models have only a small reduction in performance when compared to models synthesized for a specific angle (implying that pose detection is not 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. From the results presented in Table 3, it can be observed that model extension considerably reduces the EER; these results thus also support the use of extended frontal models.

10. CONCLUSIONS AND FUTURE WORK

We addressed the problem of non-frontal face verification (when only a frontal training image is available) by extending each client's frontal face model with artificially synthesized models for non-frontal views. In the framework of a system using local features and a Gaussian Mixture Model based classifier, we proposed the Maximum Likelihood Shift (MLS) synthesis technique and compared its performance against a Maximum Likelihood Linear Regression (MLLR) based technique and the recently proposed UBMdiff technique. All techniques rely on prior information and learn how a generic face model for the frontal view is related to generic models for non-frontal views. Experiments on the FERET database suggest that the proposed MLS technique (where the shift of the means is found under a maximum likelihood constraint) is more suitable than MLLR (due to a lower number of free parameters) and UBMdiff (due to lack of heuristics). The results further suggest that extending frontal models considerably reduces errors.

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; moreover, it would be useful to evaluate alternative size normalization approaches in order to address the scaling problem mentioned in Section 2.

11. REFERENCES

[1] J.J. Atick et al., "Statistical Approach to Shape from Shading: Reconstruction of Three-Dimensional Face Surfaces from Single Two-Dimensional Images", *Neural Computation* 8, 1996, 1321-1340.

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

[3] V. Blanz et al., "Face Identification across Different Poses and Illuminations with a 3D Morphable Model", *Proc. 5th IEEE Int. Conf. Automatic Face and Gesture Recognition*, Washington, D.C., 2002.

[4] F. Cardinaux et al., "Comparison of MLP and GMM Classifiers for Face Verification on XM2VTS", *Proc. 4th Int. Conf. Audio- and Video-Based Biometric Person Authentication*, Guildford, 2003.

[5] F. Cardinaux, C. Sanderson, S. Bengio, "Face Verification Using Adapted Generative Models", *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* 34 (7), 2001, 1393-1403.

[7] R. Duda et al., *Pattern Classification*, John Wiley & Sons, USA, 2001.

[8] S. Furui, "Recent Advances in Speaker Recognition", *Pattern Recognition Letters* 18 (9), 1997, 859-872.

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

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

[11] 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.

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

[13] P. Niyogi et al., "Incorporating Prior Information in Machine Learning by Creating Virtual Examples", *Proc. IEEE* 86 (11), 1998, 2196-2209.

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

[15] 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.

[16] D. Reynolds et al., "Speaker Verification Using Adapted Gaussian Mixture Models", *Digital Signal Processing* 10 (1-3), 2000, 19-41.

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

[18] C. Sanderson, K.K. Paliwal, "Fast features for face authentication under illumination direction changes", *Pattern Recognition Letters* 24 (14), 2003, 2409-2419.

[19] C. Sanderson, S. Bengio, "Augmenting Frontal Face Models for Non-Frontal Verification", *Proc. Workshop Multimodal User Authentication*, Santa Barbara, 2003, pp. 165-172. *(http://mmua.cs.ucsb.edu)*

[20] C. Sanderson, S. Bengio, "Statistical Transformation Techniques for Face Verification Using Faces Rotated in Depth", IDIAP Research Report 04-04, Martigny, Switzerland, 2004. (see *www.idiap.ch*)