Non-Linear Variance Reduction Techniques in Biometric Authentication

Norman Poh and Samy Bengio
IDIAP, Rue du Simplon 4, Martigny, CH-1920 Martigny, Switzerland
Email: {norman,bengio}@idiap.ch
Telephone: (41) 27–721.77.53 Fax: (41) 27–721.77.12

Abstract— **In this paper, several approaches that can be used to improve biometric authentication applications are proposed. The idea is inspired by the ensemble approach, i.e., the use of several classifiers to solve a problem. Compared to using only one classifier, the ensemble of classifiers has the advantage of reducing the overall variance of the system. Instead of using multiple classifiers, we propose here to examine other possible means of variance reduction (VR), namely through the use of multiple synthetic samples, different extractors (features) and biometric modalities. The scores are combined using the average operator, Multi-Layer Perceptron and Support Vector Machines. It is found empirically that VR via modalities is the best technique, followed by VR via extractors, VR via classifiers and VR via synthetic samples. This order of effectiveness is due to the corresponding degree of independence of the combined objects (in decreasing order). The theoretical and empirical findings show that the combined experts via VR techniques *always* perform better than the average of their participating experts. Furthermore, in practice, *most* combined experts perform better than any of their participating experts.**

I. Introduction

Biometric authentication (BA) is the problem of verifying an identity claim using a person's behavioural and physiological characteristics. BA is becoming an important alternative to traditional authentication methods such as keys ("something one has", i.e., by possession) or PIN numbers ("something one knows", i.e., by knowledge) because it is essentially "who one is", i.e., by biometric information. Therefore, it is not susceptible to misplacement, forgetfulness or reproduction. Examples of biometric modalities are fingerprint, face, voice, hand-geometry and retina scans [1].

However, to date, biometric-based security systems (devices, algorithms, architectures) still have room for improvement, particularly in their accuracy, tolerance to various noisy environments and scalability as the number of individuals increases. The focus of this study is to improve the system accuracy by directly minimising the noise via various variance reduction techniques.

Biometric data is often noisy because of deformable templates, corruption by environmental noise, variability over time and occlusion by the user's accessories. The higher the noise, the less reliable the biometric system becomes.

Advancements in biometrics show two emerging solutions: combining several biometric modalities [2], [3] (often called multi-modal biometrics) and combining several samples of a single biometric modality [4]. These techniques are related to *variance reduction* (VR). This is a phenomenon originated from combining classifier scores. Specifically, by combining the outputs of N classifier scores using an average operator (in the simplest case), one can reduce the variance of the combined score, with respect to the target score, by a factor of N [5, Chap. 9], if the classifier scores are not correlated (or independent from each another). On the other hand, in the extreme case, when they are completely correlated (dependent on each other), there will be no reduction in variance at all [6].

In the context of BA, when one combines several biometric modalities or several samples, one indeed exploits the independence of each modality and sample, respectively. In this work, we examine several other ways to exploit this (often partial) independence, namely via extractors, classifiers and synthetic samples. In short, all these methods can be termed as follows: Variance Reduction (VR) via classifiers, VR via extractors, VR via samples and VR via (biometric) modalities.

To our opinion, VR techniques are potential to improve the accuracy of BA systems because better classifiers or ensemble methods, feature extraction algorithms and biometric-enabled sensors are emerging. Instead of choosing one best technique (best features, classifiers, etc), VR techniques propose to combine these new algorithms with existing techniques (features, classifiers) to obtain improved results, whenever this is feasible. The added overhead cost will be computation time and possibly hardware cost in the case of adding new sensors (as opposed to other VR techniques which *do not require* any extra hardware).

II. Variance Reduction in Biometric Authentication

A. Variance Reduction

This section presents a brief findings on the theory of variance reduction (VR). Details can be found in [6].

A person requesting an access can be measured by his or her biometric data. Let this biometric data be $\mathbf{x}$. This measurement can be done in several methods, to be explored later. Let i denote the i-th extract of $\mathbf{x}$ by a given method. For the sake of comprehension, one method to do so is to use multiple samples. Thus, in this case, i denotes the i-th sample. If the chosen method uses multiple biometric modalities, then i refers to the i-th biometric modality. Let the measured relationship be denoted as $y_i(\mathbf{x})$. It can be thought as the i-th response (of the sample or modality, for instance) given by a biometric system. Typically, this output (e.g. score) is used to make the accept/reject decision. $y_i(\mathbf{x})$ can be decomposed into two components, as follows:

$$y_i(\mathbf{x}) = h(\mathbf{x}) + \eta_i(\mathbf{x}), \tag{1}$$

where $h(\mathbf{x})$ is the "target" function that one wishes to estimate and $\eta_i(\mathbf{x})$ is a random additive noise with zero mean, also dependent on $\mathbf{x}$.

Let N be the number of trials, (e.g., the number of samples, assuming that the chosen method uses multiple samples hereinafter). The mean of y over N trials, denoted as $\bar{y}(\mathbf{x})$ is:

$$\bar{y}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} y_i(\mathbf{x}). \tag{2}$$

When N samples are available and they are used separately, the *average of variance* made by each sample, independently, is:

$$\mathrm{VAR}_{AV}(\mathbf{x}) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} \mathrm{VAR}[y_i(\mathbf{x})], \tag{3}$$

where VAR[$\cdot$] is the variance of $\cdot$.

On the other hand, the variance as a result of averaging (or *variance of average*) due to Eq. 2 is defined as:

$$\mathrm{VAR}_{COM}(\mathbf{x}) \quad = \quad E[(\bar{y}(\mathbf{x}) - h(\mathbf{x})])^2], \tag{4}$$

where E[$\cdot$] is the expectation of $\cdot$. In our previous work [6], it has been shown that:

$$\frac{1}{N}\mathrm{VAR}_{AV}(\mathbf{x}) \leq \mathrm{VAR}_{COM}(\mathbf{x}) \leq \mathrm{VAR}_{AV}(\mathbf{x}). \tag{5}$$

This equation shows that when scores $y_i, i = 1, \ldots, N$ are not correlated, the variance of average is reduced by a factor of $1/N$ with respect to the average of variance. On the other hand, when the scores are totally correlated, there is no reduction of variance, with respect to the average of variance.

To measure *explicitly* the factor of reduction, we introduce α, which can be defined as follows:

$$\alpha = \frac{\mathrm{VAR}_{AV}(\mathbf{x})}{\mathrm{VAR}_{COM}(\mathbf{x})}. \tag{6}$$

By dividing Equation 5 by VAR_{COM} and rearranging it, we can deduce that $1 \leq \alpha \leq N$.

B. Variance Reduction and Classification Reduction

Figure 1 illustrates the effect of averaging scores in a two-class problem, such as in BA where an identity claim could belong either to a client or an impostor. Let us assume that the genuine user scores in a situation where 3 samples are available but are used separately, follow a normal distribution of mean 1.0 and variance ($\mathrm{VAR}_{AV}(\mathbf{x})$ of genuine users) 0.9, denoted as $\mathcal{N}(1, \sqrt{0.9})$, and that the impostor scores (in the mentioned situation) follow a normal distribution of $\mathcal{N}(-1, \sqrt{0.6})$ (both graphs are plotted with "+"). If for each access, the 3 scores are used, according to Equation 6, the variance of the resulting distribution will be reduced by a factor (which is the value α defined in Equation 6) of 3 or less. Both resulting distributions are plotted with "o". Note the area where both the distributions cross before and after. The later area is shaded in Figure 1. This area corresponds to the zone where minimum amount of mistakes will be committed given that the threshold is optimal[1]. Decreasing this area implies an improvement in the performance of the system.

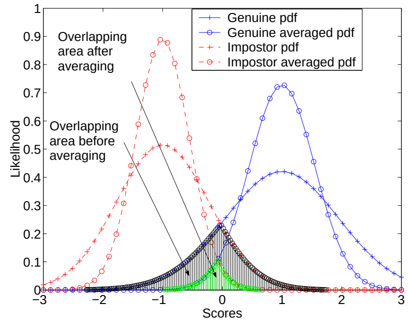

Fig. 1. Averaging score distributions in a two-class problem

C. Variance Reduction and Correlation in the Input Score Space

From the previous section, it is obvious that by reducing the variance, the classification results should be improved. How much variance can be reduced depends on how correlated the input scores are. The correlation between scores of two experts can be examined by plotting their scores on a 2D-plan, with one axis for each expert. This is shown in Fig. 2 and 3. The first figure shows a scatter-plot of scores taken from two experts working on the *same* features. The second figure shows a scatter-plot of scores taken from two experts working on *different biometric modalities*. Details of the experts are explained in Sec. IV. As can be seen, the scores of the former overlaps more than the latter, i.e., if a boundary is to be drawn between clients and impostors scores, it would be more difficult for the former problem than the latter problem. Note that overlapping occurs when both experts make the same errors. Thus, there will be more classification errors in the former problem than in the latter.

D. Exploring Various Variance Reduction Techniques

This section explores various variance reduction (VR) techniques that can be applied to the BA problem.

A BA system can be viewed as a system consisted of sensors, extractors, classifiers and a supervisor. Sensors such as cameras are responsible to capture a person's biometric traits. Extractors are responsible to compress and detect salient features that are useful for discriminating a person from the others. Classifiers are responsible for matching the extracted features from previously stored features that are known to belong to the person. Finally, in the context of multiple modalities, features, classifiers or samples, a supervisor is needed to merge all the results.

This serial concatenation process of sensors, extractors, classifiers and a supervisor shows that error may accumulate along the chain because each module depends on its previous module. An important finding in Sec. II-A [6] is that it is beneficial to increase the number of processes. For instance, one can use more samples or more biometric modalities. In these two cases, N will be the number of samples and

[1]Optimal in the Bayes sense, when (1) the cost and (2) probability of both types of errors are equal.

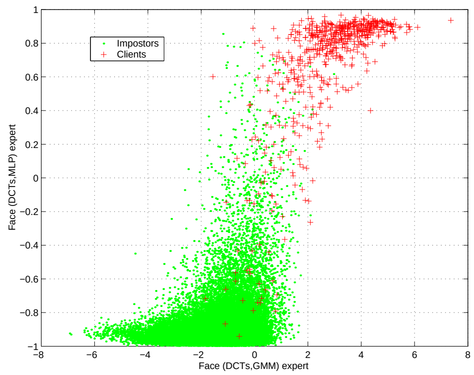

Fig. 2. Scores from experts of different features

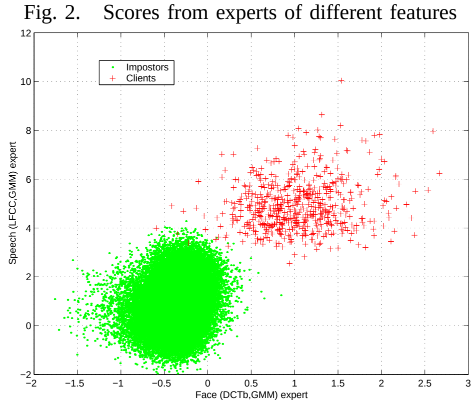

Fig. 3. Scores from experts of different biometric modalities

modalities, respectively. This is because by increasing N, one can decrease the variance further, regardless of how correlated the scores obtained from these N experts are. Note that in the work of Kittler *et al* [4], they showed that by increasing N samples up to a limit, there is no more gain in accuracy. When this happens, they said that the system is "saturated". In our context, we expand N through different methods, as follow:

- **Multiple Biometric Modalities**. Each modality has its own feature set and classifiers. In other words, they operate independently of each other [7]–[9].
- **Multiple Samples**. Samples could be real [4] or virtually generated [10].
- **Multiple Extractors**. Each feature is classified by a classifier independently of other features [11], [12].
- **Multiple Classifiers**. All classifiers receive the same input features. Heterogeneous types of classifiers can be used. Unstable homegenous classifiers such as MLP trained by bagging or with different hidden units can also be used. In general, many ensemble methods such as bagging, boosting, via Error-Correcting Output-Coding fall in this category [13].

For each method mentioned above, the problem now is to combine these N scores. This is treated in the next subsection.

E. Fusions in Variance Reduction Techniques

In Sec. II-A, it has been illustrated that correlation of scores in the input space plays a vital role in determining the success of the resultant combined system. Furthermore, by simple averaging of N scores, it has been shown that the variance of the resultant combined score can be reduced by a factor between 1 and N with respect to the average of variance.

Instead of using simple averaging, one could have used weighted average, or even non-linear techniques such as Multi-Layer Perceptrons (MLPs) and Support Vector Machines (SVMs) [5]. In the latter two cases however, one needs to select carefully the various hyper-parameters of these models (such as the number of hidden units of the MLPs or the kernel parameter of the SVMs). According to the Statistical Learning Theory [14], the expected performance of a model such as an MLP or an SVM on new data depends on the *capacity* of the set of functions the model can approximate. If the capacity is too small, the desired function might not be in the set of functions, while if it too high, several apparently good functions could be approximated, with the risk of selecting a bad one. This phenomenon is often called *over-training*. Although this capacity cannot unfortunately be explicitely estimated for complex set of functions such as MLPs and SVMs, its ordering can be used to select efficiently the corresponding hyper-parameters using some sort of validation technique. One such method is the K-fold cross-validation.

Algorithm 1 shows how K-fold cross-validation can be used to estimate the correct value of the hyper-parameters of our fusion model, as well as the decision threshold used in the case of authentication. The basic framework of the algorithm is as follows: first perform K-fold cross-validation on the training set by varying the capacity parameter, and for each capacity parameter, select the corresponding decision threshold that minimizes HTER; then choose the best hyper-parameter according to this criterion and perform a normal training with the best hyper-parameter on the whole training set; finally test the resultant classifier on the test set [8] with HTER evaluated on the previously found decision threshold.

There are several points to note concerning Algorithm 1: $\mathcal{Z}$ is a set of labelled examples of the form $(\mathcal{X}, \mathcal{Y})$, where the first term is a set of patterns and the second term is a set of corresponding labels. The "train" function receives a hyper-parameter θ and a training set, and outputs an optimal classifier $\hat{F}$ by minimising the HTER on the training set. The "test" function receives a classifier $\hat{F}$ and a set of examples, and outputs a set of scores for each associated example. The "thrd$_{HTER}$" function returns a *decision threshold* that minimizes HTER by minimising $|\text{FAR}(\Delta) - \text{FRR}(\Delta)|$ with respect to the threshold Δ (FAR(Δ) and FRR(Δ) are false acceptance and false rejection rates, as a function of Δ) while L_{HTER} returns the HTER *value* for a particular decision threshold. What makes this cross-validation different from classical cross-validation is that there is only one single decision threshold and the corresponding HTER value for all the held-out folds and for a given hyper-parameter θ. This is because it is logical to union scores of all held-out folds into one single set of scores to select the decision threshold (and obtain the corresponding HTER).

F. Fusions For VR via Samples

All the VR techniques discussed earlier can be treated in a general manner, except VR via samples. This is because

Algorithm 1 Risk Estimation $(\Theta, K, \mathcal{Z}^{train}, \mathcal{Z}^{test})$

REM: Risk Estimation with K-fold Validation. See [8].
Θ : a set of values for a given hyper-parameter
$\mathcal{Z}^i$: a tuple $(\mathcal{X}^i, \mathcal{Y}^i)$, for $i \in \{train, test\}$ where
$\mathcal{X}$: a set of patterns. Each pattern contains scores/hypothesis from base experts
$\mathcal{Y}$: a set of labels $\in \{client, impostor\}$
Let $\cup_{k=1}^{K} \mathcal{Z}^k = \mathcal{Z}^{train}$
for each hyper-parameter $\theta \in \Theta$ **do**
  for each $k = 1, \ldots, K$ **do**
    $\hat{F}_\theta$ = train(θ, $\cup_{j=1, j\neq k}^{K} \mathcal{Z}^j$)
    $\hat{\mathcal{Y}}_\theta^k$ = test($\hat{F}_\theta$, $\mathcal{X}^k$)
  end for
  $\Delta_\theta = \text{thrd}_{HTER}\left(\{\hat{\mathcal{Y}}_\theta^k\}_{k=1}^K, \{\mathcal{Y}^k\}_{k=1}^K\right)$
end for
$\theta^* = \arg\min_\theta \left(L_{HTER}\left(\Delta_\theta, \{\hat{\mathcal{Y}}_\theta^k\}_{k=1}^K, \{\mathcal{Y}^k\}_{k=1}^K\right)\right)$
$\hat{F}_{\theta^*}$ = train(θ^*, $\mathcal{Z}^{train}$)
$\hat{\mathcal{Y}}_{\theta^*}^{test}$ = test($\hat{F}_{\theta^*}$, $\mathcal{X}^{test}$)
return $L_{HTER}(\Delta_{\theta^*}, \hat{\mathcal{Y}}_{\theta^*}^{test}, \mathcal{Y}^{test})$

the ordering of scores induced by samples are not important. Simply concatenating the scores and feeding them to a classifier may not be an optimal solution. Another problem that might arise is that when there are many scores, possibly in the range of hundreds, (one can generate as many virtual scores as one wishes), matching should be done in terms of their distribution instead. We hence propose two solutions to handle this: 1) estimate the likelihood of the set of virtual scores when coming from either a client or an impostor distribution; 2) estimate the distribution of the scores so that matching will be performed between a competing client and an impostor distribution. Both approaches assume that the scores are generated independently from some unknown distributions. Of course this independence assumption is not true, but it is good enough for most practical problems.

The first approach is carried out using Gaussian Mixture Models (GMMs) to model the scores. First estimate the client and impostor distributions using GMMs by separately maximizing the likelihood of the client and impostor scores using the Expectation-Maximization algorithm [5]. During an access request with one real biometric sample, a set of synthetic samples and hence a set of scores are generated. These scores will be fed to the client and an impostor GMM score distribution. Let $\log p(\mathbf{x}|\theta_C)$ be the log likelihood of the set of scores $\mathbf{x}$ given the client GMM model θ_C and $\log p(\mathbf{x}|\theta_I)$ be the same term but for the impostor model. The decision is often taken using the so-called log-likelihood ratio: $s = \log p(\mathbf{x}|\theta_C) - \log p(\mathbf{x}|\theta_I)$

As for the second approach, we propose to first model the distribution of these synthetic scores using a Parzen window non parametric density model [5, Chap. 2] and then compute the relative entropy of each distribution, which is defined as follows:

$$L(p, q) = -\sum_i p(y_i) \log \frac{q(y_i)}{p(y_i)}, \tag{7}$$

where q and p are two distributions. Entropy can be regarded as a distortion of $q(y)$ from $p(y)$ This alone does not give discriminative information. To do so, entropies of a client and an impostor distribution should be used together. Let $L(p_C, q)$ be the entropy of $q(y)$ with respect to a client distribution and $L(p_I, q)$ be that of $q(y)$ with respect to an impostor distribution. Then the difference between these two entropies, can be defined as: $s = L(p_I, q) - L(p_C, q)$.

When $s > 0$, the distortion of $q(y)$ from an impostor distribution is greater than that of a client distribution, which reflects how likely a set of synthetic scores belong to a client. In fact, for both approaches, $s > \triangle$ is used instead, where $\triangle$ is a threshold chosen *a priori* according to the HTER criterion.

III. Experimental Settings

A. XM2VTS Database Description

The XM2VTS database [15] contains synchronised video and speech data from 295 subjects, recorded during four sessions taken at one month intervals. On each session, two recordings were made, each consisting of a speech shot and a head shot. The speech shot consisted of frontal face and speech recordings of each subject during the pronunciation of a sentence.

The database is divided into three sets: a training set, an evaluation set and a test set. The training set was used to build client models, while the evaluation set (Eval) was used to compute the decision thresholds (as well as other hyper-parameters) used by classifiers. Finally, the test set (Test) was used to estimate the performance.

The 295 subjects were divided into a set of 200 clients, 25 evaluation impostors and 70 test impostors. There exists two configurations or two different partitioning of the training and evaluation sets. They are called Lausanne Protocol I and II, denoted as **LP1** and **LP2** in this paper. Thus, besides the data for training the model, the following four data sets are available for evaluating the performance: LP1 Eval, LP1 Test, LP2 Eval and LP2 Test. Note that LP1 Eval and LP2 Eval are used to calculate the optimal thresholds that will be used in LP1 Test and LP2 Test, respectively. Results are reported only for the test sets, in order to be as unbiased as possible (using an *a priori* selected threshold). Table I is the summary of the data. In both configurations, the test set remains the same.

TABLE I
THE LAUSANNE PROTOCOLS OF XM2VTS DATABASE

Data sets	Lausanne Protocols	
	LP1	LP2
Training client accesses	3	4
Evaluation client accesses	600 (3×200)	400 (2×200)
Evaluation impostor accesses	40,000 ($25 \times 8 \times 200$)	
Test client accesses	400 (2×200)	
Test impostor accesses	112,000 ($70 \times 8 \times 200$)	

However, there are three training data per client for LP1 and four training data per client for LP2. More details can be found in [16].

B. Features Used for the XM2VTS Database

For the face data, a bounding box is placed on a face according to eyes coordinates located manually. This assumes a perfect face detection[2]. The face is cropped and the extracted sub-image is down-sized to a 30×40 image. After enhancement and smoothing, the face image has a feature vector of dimension 1200.

In addition to these normalised features, RGB (Red-Green-Blue) histogram features are used. For each colour channel, a histogram is built using 32 discrete bins. Hence, the histograms of three channels, when concatenated, form a feature vector of 96 elements. More details about this method, including experiments, can be obtained from [17].

Another feature set derived from Discrete Cosine Transform (DCT) coefficients [18], [19] has also given good performance. The idea is to divide images into overlapping blocks. For each block, a subset of DCT coefficients are computed. The horizontal, vertical and diagonal (with respect to a reference block of) DCT coefficients can also be derived. It has been shown that these features are comparable (in terms of performance in the context of BA) to features derived from Principal Component Analysis [18].

For the speech data, the feature sets used in the experiments are Linear Filter-bank Cepstral Coefficients (LFCC) [20], Phase Auto-correlation derived Mel-scale Frequency Cepstrum Coefficients (PAC) [21] and Mean-Subtracted Spectral Sub-band Centroids (SSC) [22]. The speech/silence segmentation is done using two competing Gaussians trained in an unsupervised way by maximising the likelihood of the data given a mixture of the 2 Gaussians. One Gaussian will end up modelling the speech and the other will end up modelling the non-speech feature frames [23]. In general, the segmentation given by this technique is satisfactory.

IV. Results

In order to analyse the effects due to VR techniques, we first present the baseline experimental results. This is followed by results obtained by various VR techniques. Note that all results reported here are in terms of **percentage of HTER**, the thresholds are all selected **a priori** (i.e., tuned on the training set, hence the threshold is *completely independent* of the test set and is thus unbias), and for the combination strategy, **only two experts are used** each time.

A. Baseline Performance on The XM2VTS Database

The face baseline experts are based on the following features:

1) **FH**: It is a normalised **f**ace image concatenated with its RGB **H**istogram (thus the abbreviation **FH**)
2) **DCTs**: It is a set of face features derived from a subset of DCT-derived coefficients. The DCT algorithm used overlapping windows (block of sub-image) having the size of 40×32 pixels. (**s** indicates the use of this small image comparing to the bigger size image with the abbreviation **b**).
3) **DCTb**: Similar to DCTs except that it uses overlapping windows having the size of 80×64.

The speech baseline experts are based on the following features:

1) **LFCC**: The Linear Filter-bank Cepstral Coefficient (LFCC) speech features were computed with 24 linearly-spaced filters on each frame of Fourier coefficients sampled with a window length of 20 milliseconds and each window moved at a rate of 10 milliseconds. 16 DCT coefficients are computed to decorrelate the 24 coefficients (log of power spectrum) obtained from the linear filter-bank. The first temporal derivatives are added to the feature set.
2) **PAC**: The PAC-MFCC features are derived with a window length of 20 miliseconds and each window moves at a rate of 10 miliseconds. 20 DCT coefficients are computed to decorrelate the 30 coefficients obtained from the Mel-scale filter-bank. The first temporal derivatives are added to the feature set.
3) **SSC**: The mean-subtracted SSCs are obtained from 16 coefficients. The γ parameter, which is a parameter that raises the power spectrum and controls how much influence the centroid, is set to 0.7. Also The first temporal derivatives are added to the feature set.

Two different types of classifiers were used for these experiments: a Multi-Layer Perceptron (MLP) [5] and a Bayes Classifier using Gaussian Mixture Models (GMMs) to estimate the class distributions [5]. While in theory both classifiers could be trained using any of the previously defined feature sets, in practice only some specific combinations appear to yield reasonable performance.

Whatever the classifier is, the hyper-parameters (e.g. the number of hidden units for MLPs or the number of Gaussian components for GMMs) are tuned on the evaluation set LP1 Eval. The same set of hyper-parameters are used in both LP1 and LP2 configurations of the XM2VTS database.

For each client-specific MLP, the samples associated to the client are treated as positive patterns while all other samples *not* associated to the client are treated as negative patterns. All MLPs reported here were trained using the stochastic version of the error-backpropagation training algorithm [5].

For the GMMs, two competing models are often needed: a world and a client-dependent model. Initially, a world model is first trained from an external database (or a sufficiently large data set) using the standard Expectation-Maximisation algorithm [5]. The world model is then adapted for each client to the corresponding client data of the training set of the XM2VTS database using the Maximum-A-Posteriori adaptation [24] algorithm.

The baseline experiments based on DCT coefficients were reported in [19] while those based on normalised face images and RGB histograms (FH features) were reported in [17]. Details of the experiments, coded in the pair **(feature, classifier)**, for the face experts, are as follows:

[2]Hence, even if this is often done in the literature, the final results using face scores could be optimistically biased due to this manual detection step. Note on the other hand that due to the clean and controlled quality of XM2VTS, automatic detectors often yield detection rates around 99%.

1) **(FH,MLP)** Features are normalised **F**ace concatenated with **H**istogram features. The client-dependent classifier used is an MLP with 20 hidden units. The MLP is trained with geometrically transformed images [17].
2) **(DCTs,GMM)** The face features are DCT-derived coefficients with each overlapping window (block of sub-image) having the size of 40×32 pixels There are 64 Gaussian components in the GMM. The world model is trained using *all the clients* in the training set [19].
3) **(DCTb,GMM)** Similar to (DCTs,GMM), except that the features used are DCT-derived coefficients with the overlapping window-size of 80×64. The corresponding GMM has 512 Gaussian components [19].
4) **(DCTs,MLP)** Features are the same as those in (DCTs,GMM) except that an MLP is used in place of a GMM. The MLP has 32 hidden units [19].
5) **(DCTb,MLP)** The features are the same as those in (DCTb,GMM) except that an MLP with 128 hidden units is used [19].

and for the speech experts:

1) **(LFCC,GMM)** This is the Linear Filter-bank Cepstral Coefficients (LFCC) obtained from the speech data of the XM2VTS database. The GMM has 200 Gaussian components, with the minimum relative variance of each Gaussian fixed to 0.5, and the MAP adaptation weight equals 0.1. This is the best known model currently available.
2) **(PAC,GMM)** The same GMM configuration as in LFCC is used. Note that in general, 200-300 Gaussian components would give about 1% of difference of HTER.
3) **(SSC,GMM)** The same GMM configuration as in LFCC is used.

The baseline performances are shown in Table II.

TABLE II

BASELINE PERFORMANCE IN HTER(%) OF DIFFERENT MODALITIES EVALUATED ON XM2VTS BASED ON *a priori* SELECTED THRESHOLDS

Data sets	(Features, classifiers)	FAR	FRR	HTER
Face LP1 Test	(FH,MLP)	1.751	2.000	1.875
Face LP1 Test	(DCTs,GMM)	4.454	4.000	4.227
Face LP1 Test	(DCTb,GMM)	1.840	1.500	1.670
Face LP1 Test	(DCTs,MLP)	3.219	3.500	3.359
Face LP1 Test	(DCTb,MLP)	4.443	8.000	6.221
Speech LP1 Test	(LFCC,GMM)	1.029	1.250	1.139
Speech LP1 Test	(PAC,GMM)	4.608	8.000	6.304
Speech LP1 Test	(SSC,GMM)	2.374	2.500	2.437
Face LP2 Test	(FH,MLP)	1.469	2.250	1.860
Face LP2 Test	(DCTb,GMM)	1.039	0.250	0.644
SpeechLP2 Test	(LFCC,GMM)	1.349	1.250	1.300
Speech LP2 Test	(PAC,GMM)	5.283	8.000	6.642
Speech LP2 Test	(SSC,GMM)	2.276	1.750	2.013

As can be seen, among the face experiments, (DCTb,GMM) performs the best across all configurations while (DCTb,MLP) performs the worst. In the speech experiments, LFCC features are the best features, followed by SSC and PAC, in decreasing order of accuracy. Regardless of strong or weak classifiers, as long as their correlation is weak, they can be used in the VR techniques.

B. VR via Different Modalities, Extractors, Classifiers

Table III shows the results combining scores of two modalities, two extractors and two classifiers (working on the same feature space). The second to last column shows the mean HTER of each of the two underlying experts while the last column shows the minimum HTER of the two experts. The three sub-columns under the heading "joint HTER" are the HTERs of the combined experts via the mean operator, MLP and SVM. Numbers in bold are the best HTER among the three fusion methods. A quick examination of this table reveals that all combined experts via modalities are better than the best underlying expert (compare min HTER with the scores in the joint HTER). However, the combined experts via extractors and classifiers, as shown in Table IV, are not always better than their participating experts.

TABLE III

PERFORMANCE IN (%) OF HTER OF VR VIA MODALITIES ON XM2VTS BASED ON *a priori* SELECTED THRESHOLDS

(a) Face experts and (LFCC,GMM) expert

Data sets	Face, Experts	Joint HTER			mean HTER	min HTER
		mean	MLP	SVM		
LP1 Test	(FH,MLP)	0.399	**0.366**	0.381	1.507	1.139
LP1 Test	(DCTs,GMM)	**0.537**	0.576	0.613	2.683	1.139
LP1 Test	(DCTb,GMM)	0.520	0.483	**0.475**	1.405	1.139
LP1 Test	(DCTs,MLP)	0.591	0.611	**0.587**	2.249	1.139
LP1 Test	(DCTb,MLP)	0.497	0.489	**0.485**	3.680	1.139
LP2 Test	(FH,MLP)	0.151	**0.150**	0.389	1.580	1.300
LP2 Test	(DCTb,GMM)	0.147	**0.130**	0.252	0.972	0.644

(b) Face experts and (PAC,GMM) expert

Data sets	Face, Experts	Joint HTER			mean HTER	min HTER
		mean	MLP	SVM		
LP1 Test	(FH,MLP)	1.114	**0.856**	0.970	4.090	1.875
LP1 Test	(DCTs,GMM)	1.407	1.425	**1.402**	5.266	4.227
LP1 Test	(DCTb,GMM)	**0.899**	0.900	0.923	3.987	1.670
LP1 Test	(DCTs,MLP)	1.248	1.056	**1.009**	4.832	3.359
LP1 Test	(DCTb,MLP)	3.978	**2.455**	2.664	6.263	6.221
LP2 Test	(FH,MLP)	1.282	**0.765**	0.855	4.251	1.860
LP2 Test	(DCTb,GMM)	0.243	**0.222**	0.431	3.643	0.644

(c) Face experts and (SSC,GMM) expert

Data sets	Face, Experts	Joint HTER			mean HTER	min HTER
		mean	MLP	SVM		
LP1 Test	(FH,MLP)	0.972	0.786	**0.742**	2.156	1.875
LP1 Test	(DCTs,GMM)	**1.028**	1.175	1.213	3.332	2.437
LP1 Test	(DCTb,GMM)	0.756	**0.704**	0.742	2.053	1.670
LP1 Test	(DCTs,MLP)	1.167	**0.829**	0.850	2.898	2.437
LP1 Test	(DCTb,MLP)	2.986	1.176	**1.121**	4.329	2.437
LP2 Test	(FH,MLP)	0.901	**0.302**	0.404	1.937	1.860
LP2 Test	(DCTb,GMM)	**0.049**	0.162	0.383	1.329	0.644

C. VR via Virtual Samples

The experiments on VR via samples are presented differently than the rest because they cannot be evaluated using the mean HTER and min HTER. Instead, the combined experts are compared to the original baseline experts (compare the first row of Table V against the other rows). The two numbers in bold are the best fusion technique for LP1 and LP2 configurations, respectively. The Entropy and GMM approaches are discussed in Sec. II-F. The median technique refers to combining synthetic scores using the median operator which is known to be robust to outlier scores. We note that the best fusion technique on these datasets are the entropy approach and the

TABLE IV

PERFORMANCE IN (%) OF HTER OF VR VIA EXTRACTORS AND CLASSIFIERS ON XM2VTS BASED ON *a priori* SELECTED THRESHOLDS

Data sets	(Features, classifiers)	Joint HTER			mean	min
		mean	MLP	SVM	HTER	HTER
LP1 Test	(FH,MLP) (DCTs,GMM)	1.641	**1.379**	1.393	3.051	1.875
LP1 Test	(FH,MLP) (DCTb,GMM)	**1.123**	1.151	1.528	1.772	1.670
LP1 Test	(FH,MLP) (DCTs,MLP)	**1.475**	1.667	1.476	2.617	1.875
LP1 Test	(FH,MLP) (DCTb,MLP)	1.948	**1.933**	1.938	4.048	1.875
LP1 Test	(LFCC,GMM) (SSC,GMM)	1.296	1.444	**1.142**	1.788	1.139
LP1 Test	(PAC,GMM) (SSC,GMM)	3.594	2.954	**2.663**	4.370	2.437
LP2 Test	(FH,MLP) (DCTb,GMM)	0.896	0.670	**0.488**	1.252	0.644
LP2 Test	(LFCC,GMM) (SSC,GMM)	1.107	**1.034**	1.063	1.656	1.300
LP2 Test	(PAC,GMM) (SSC,GMM)	2.614	2.316	**2.125**	4.328	2.013
LP1 Test	(DCTs,GMM) (DCTs,MLP)	2.873	**2.486**	2.697	3.793	3.359
LP1 Test	(DCTb,GMM) (DCTb,MLP)	2.898	1.532	**1.471**	3.946	1.670

GMM approach for LP1 and LP2, respectively. For LP1, the entropy approach is *significantly better* with 90% confidence level than the mean operator according to the McNemar's Test[3] [25] (i.e., with a difference of 0.006 HTER% between the two approaches). For LP2, the GMM approach is *significantly better* than the mean operator with 99% confidence level. This shows that exploiting the distribution of scores *is better* than using the simple mean operator.

TABLE V

PERFORMANCE IN (%) OF HTER OF DIFFERENT COMBINATION METHODS OF SYNTHETIC SCORES.

Method	HTER	
	LP1	LP2
Original	1.875	1.737
Mean	1.612	1.518
Median	1.667	1.547
GMM	1.709	**1.493**
Entropy	**1.606**	1.559

D. Evaluation of Experiments

Let us define two measures of gain so as to draw a summary of the experiments carried out above, as below:

$$\beta_{mean} = \frac{\text{mean}_i(\text{HTER}_i)}{\text{HTER}_c}$$

$$\beta_{min} = \frac{\text{min}_i(\text{HTER}_i)}{\text{HTER}_c},$$

where β_{mean} and β_{min} measure how many times the HTER of the combined expert c is smaller than the mean and the min HTER of the underlying experts $i = 1, \ldots, N$. β_{mean} is designed to verify Eq. 6, which is somewhat akin to α. According to the theoretical analysis presented in Sec. II-A, $\alpha \geq 1$ should be satisfied. The β_{min}, on the other hand, is a more realistic criterion, i.e., one wishes to obtain better performance than the underlying experts, but there is no analytical proof that $\beta_{min} \geq 1$.

The β_{mean} for each experiment are shown in Table VI(a) for VR via modalities, extractors and classifiers, (b) for VR via synthetic samples and (c) for the gain ratio β_{min}. Note that VR via synthetic samples cannot be evaluated with the β_{min} criterion. It can only be compared to its original method (i.e., with real samples). This gain ratio can be defined as $\beta_{real} = HTER_{real}/HTER_c$, where $real$ is the expert that takes real samples and c is the expert that combines scores obtained from synthetic samples.

Note that the β_{mean} for VR via modalites are sub-divided into 3 parts according to the 3 baseline speech experts (LFCC,GMM), (SSC,GMM) and (PAC,GMM) in a *significantly* decreasing order of accuracy. In such situations, the β_{mean} for these 3 baselines still have comparable range of values, which are bigger than other VR techniques. One possible conclusion is that regardless of the degree of accuracy of participating experts, as long as they are weakly correlated, high β_{mean} can be achieved. Although the mean operator seems to perform the best in the overall VR via modalities based on β_{mean}, it should be noted that out of the 27 experiments in Table III, 4 experiments are best combined with the mean operator, while there are 10 and 7 best results for MLPs and SVMs, respectively. Moreover, the standard deviation of the mean operator is much larger than that of MLPs and SVMs. In these experiments, MLP turns out to be a good candidate for fusion in most situations for VR via modalities. It should be emphasized that the success application of MLPs or SVMs in this fusion problem depends largely on the correct capacity estimate of cross-validation.

Note that Table VI(a) shows that $\beta_{mean} \geq 1$ for all fusion techniques but in (c), $\beta_{min} \geq 1$ is only true for MLPs and SVMs, but not for the mean operator, which we cannot guarantee. According to β_{mean} *on the mean operator*, VR via modalities achieves the highest gain, followed by VR via extractors, VR via classifiers and VR via synthetic samples. A similar trend is observed in (c) according to β_{min}. Such ordering is not a coincidence. It reveals that the correlation is greater and greater in the list just mentioned. In other words, β_{mean} is inversely proportional to the correlation of the underlying experts. However, with MLP and SVM as non-linear fusion techniques, this ordering is slightly perturbed because both the β_{mean} and β_{min} show that VR via classifiers are *better* than VR via extractors. Clearly, in highly correlated problem such as these, non-linear fusion techniques are better than the simple mean operator.

V. CONCLUSIONS

Variance reduction (VR) is an important technique to increase accuracy in regression and classification problems. In this study, several approaches are explored to improve Biometric Authentication systems, namely VR via modalities, VR via extractors, VR via classifiers and VR via synthetic samples. The experiments carried out on the XM2VTS database show

[3]This is done by calculating $((n_{01} - n_{10})^2 - 1)/(n_{01} + n_{10}) > p$ where p is the inverse function of $\mathcal{X}^2$ distribution (with 1 degree of freedom) at a desired confidence interval (i.e., 90%), and n_{01} and n_{10} are the number of *different* mistakes done by the two systems on the *same* accesses

TABLE VI

COMPARISON OF VARIOUS VR TECHNIQUES BASED ON ALL EXPERIMENTS CARRIED OUT USING β_{mean}, β_{min} AND β_{real}

(a) β_{mean} of all experiments

VR techniques	Table	No. of exp.	Joint HTER		
			mean	MLP	SVM
Modalities	III(a) (all)	21	5.559 ±5.879	5.390 ±3.287	4.164 ±1.474
	III(a) (LFCC)	7	5.680 ±2.683	5.843 ±2.744	4.375 ±1.482
	III(a) (PAC)	7	5.086 ±4.459	5.999 ±4.686	4.694 ±1.869
	III(a) (SSC)	7	5.910 ±9.365	4.326 ±2.128	3.422 ±0.733
Extractors	IV	9	1.604 ±0.269	1.719 ±0.313	1.842 ±0.420
Classifiers	IV	2	1.341 ±0.029	2.051 ±0.742	2.044 ±0.902
Synthetic samples	V	2	1.154 ±0.0002	MLP and SVM not used; see (b)	

(b) β_{real} of VR via synthetic samples

Methods	Gain ratio
Mean	1.154 ± 0.000178
Median	1.124 ± 0.000002
GMM	1.130 ± 0.002198
Global Entropy	1.141 ± 0.001422
Local Entropy	0.854 ± 0.000028

(c) β_{min} of all VR techniques except synthetic samples

VR techniques	Table	No. of exp.	Joint HTER		
			mean	MLP	SVM
Modalities	III(a)	21	3.043	3.109	2.459
Extractors	III(b)	9	1.009	1.067	1.120
Classifiers	III(c)	2	0.873	1.221	1.190

that the combined experts due to VR techniques *always* perform better than the average of their participating experts, which can be explained by VR using the mean operator. Furthermore, all combined experts via modalities outperform the best participating expert based on the HTER. By means of non-linear variance reduction techniques, i.e., the use of MLPs and SVMs for combing scores obtained from participating experts, empirical study shows that, in average, these techniques could produce better results than their participating experts, in the context of VR via extractors and classifiers. In the context of VR via samples, exploiting the distribution of synthetic scores using GMM or Parzen-windows is better than the mean operator. In short, this study shows that non-linear fusion techniques using MLPs and SVMs, and incorporating other *a priori* information (i.e., distribution of synthetic score in the case of synthetic samples) are vital to achieve high gain of fusion. In highly correlated situations (i.e, VR via extractors and classifiers), non-linear fusion techniques are very useful. In weakly correalted situations (i.e., VR via modalities), the mean operator could be feasible but non-linear fusion techniques are still useful if the capacity search using cross-validation is reliable. As new and more powerful extraction and classification algorithms become available, they can all be integrated into the VR framework. Therefore, VR techniques are potentially very useful for biometric authentication.

ACKNOWLEDGEMENT

The authors wish to thank the Swiss National Science Foundation through the National Centre of Competence in Research on "Interactive Multimodal Information Management (IM2)".

REFERENCES

[1] A. Jain, R. Bolle, and S. Pankanti, *Biometrics: Person Identification in a Networked Society*. Kluwer Publications, 1999.

[2] L. Hong, A. Jain, and S. Pankanti, "Can Multibiometrics Improve Performance?" Computer Science and Engineering, Michigan State University, East Lansing, Michigan, Tech. Rep. MSU-CSE-99-39, December 1999.

[3] N. Poh and J. Korczak, "Hybrid Biometric Authentication System Using Face and Voice Features," in *The 3rd Int. Conf. on Audio- and Video-Based Biometric Person Authentication, AVBPA'01*, 2001, pp. 348–353.

[4] J. Kittler, G. Matas, K. Jonsson, and M. Sanchez, "Combining Evidence in Personal Identity Verification Systems," *Pattern Recognition Letters*, vol. 18, no. 9, pp. 845–852, September 1997.

[5] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1999.

[6] N. Poh and S. Bengio, "Variance Reduction Techniques in Biometric Authentication," IDIAP, Martigny, Switzerland, IDIAP-RR 03-17, 2003.

[7] C. Sanderson and K. K. Paliwal, "Information Fusion and Person Verification Using Speech and Face Information," IDIAP, Martigny, Switzerland, IDIAP-RR 02-33, 2002.

[8] S. Bengio, C. Marcel, S. Marcel, and J. Mariéthoz, "Confidence Measures for Multimodal Identity Verification," *Information Fusion*, vol. 3, no. 04, pp. 267–276, 2002.

[9] B. Duc, E. S. Bigun, J. Bigun, G. Maitre, and S. Fischer, "Fusion of Audio and Video Information for Multi Modal Person Authentication," *Pattern Recognition Letters*, vol. 18, pp. 835–843, 1997.

[10] N. Poh, S. Marcel, and S. Bengio, "Improving Face Authetication Using Virtual Samples," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 233–236, 2003.

[11] R. Brunelli and D. Falavigna, "Personal Identification Using Multiple Cues," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 955–966, 1995.

[12] F. Smeraldi, N. Capdevielle, and J. Bigun, "Face Authentication by Retinotopic Sampling of the Gabor Decomposition and Support Vector Machines," in *Proceedings of the 2nd International Conference on Audio and Video Based Biometric Person Authentication (AVBPA'99)*, vol. I, Washington DC, USA, March 1999, pp. 125–129.

[13] T. Dietterich, "Ensemble Methods in Machine Learning," in *Multiple Classifier Systems*, 2000, pp. 1–15.

[14] V. N. Vapnik, *Statistical Learning Theory*. Springer, 1998.

[15] J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H. Yan, F. Smeraldi, J. Begun, N. Capdevielle, W. Gerstner, S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz, "Comparison of Face Verification Results on the XM2VTS Database," in *The Proceedings of the 15th ICPR*, vol. 4, 2000, pp. 858–863.

[16] J. Lüttin, "Evaluation Protocol for the XM2FDB Database (Lausanne Protocol)," IDIAP, Martigny, Switzerland, IDIAP-COM 98-05, 1998.

[17] S. Marcel and S. Bengio, "Improving Face Verification Using Skin Color Information," in *Proceedings of the 16th Int. Conf. on Pattern Recognition*. IEEE Computer Society Press, 2002.

[18] C. Sanderson and K. Paliwal, "Polynomial Features for Robust Face Authentication," *Proceedings of International Conference on Image Processing*, vol. 3, pp. 997–1000, 2002.

[19] F. Cardinaux, C. Sanderson, and S. Marcel, "Comparison of MLP and GMM Classifiers for Face Verification on XM2VTS," IDIAP, IDIAP-RR 03-10, 2003.

[20] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Oxford University Press, 1993.

[21] S. Ikbal, H. Misra, and H. Bourlard, "Phase AutoCorrelation (PAC) derived Robust Speech Features," in *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-03)*, Hong Kong, April 2003.

[22] K. K. Paliwal, "Spectral Subband Centroids as Features for Speech Recognition," in *IEEE International Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 1997, pp. 124–13.

[23] J. Mariéthoz and S. Bengio, "A Comparative Study of Adaptation Methods for Speaker Verification," in *International Conference on Spoken Language Processing ICSLP*, Denver, CO, USA, September 2002, pp. 581–584, IDIAP-RR 01-34.

[24] J. Gauvain and C.-H. Lee, "Maximum A posteriori estimation for multivariate gaussian mixture obervation of markov chains," in *IEEE Transactions on Speech Audio Processing*, April 1994, pp. 290–298.

[25] T. G. Dietterich, "Approximate Statistical Test for Comparing Supervised Classification Learning Algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.