IDIAP RESEARCH REPORT

Improving Face Authentication Using Virtual Samples

Norman Poh Hoon Thian [a]
Sébastien Marcel [a] Samy Bengio [a]

IDIAP–RR 02-40

October 2002

to appear in
2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03), April, Hong Kong.

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 – 27 – 721 77 11
fax +41 – 27 – 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

[a] IDIAP, CP 592, 1920 Martigny, Switzerland

IDIAP Research Report 02-40

Improving Face Authentication Using Virtual Samples

Norman Poh Hoon Thian Sébastien Marcel Samy Bengio

October 2002

to appear in
2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03),
April, Hong Kong.

Abstract. In this paper, we present a simple yet effective way to improve a face verification system by generating multiple virtual samples from the unique image corresponding to an access request. These images are generated using simple geometric transformations. This method is often used during training to improve accuracy of a neural network model by making it robust against minor translation, scale and orientation change. The main contribution of this paper is to introduce such method during testing. By generating N images from one single image and propagating them to a trained network model, one obtains N scores. By merging these scores using a simple mean operator, we show that the variance of merged scores is decreased by a factor between 1 and N. An experiment is carried out on the XM2VTS database which achieves new state-of-the-art performances.

1 INTRODUCTION

1.1 Problem Definition

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 sources are fingerprint, face, voice, hand-geometry and retina scans. General introduction of biometrics can be found in [5].

Biometric data is often noisy because of the failure of biometric devices to capture the plastic nature of biometric traits (e.g. deformed fingerprint due to different pressures), 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. Current 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 by using multiple virtual samples, when multiple real samples are not available.

1.2 Related work in the literature

In the literature, to the best of our knowledge, the closest work to ours is the one reported by Kittler *et al* [1]. The fundamental difference is that they assume that multiple samples are available. In real-life situation, where a face image is scanned and transfered over a communication line, obtaining multiple face images for each access may not be feasible. In this case, "virtual" samples could be used. Although there is no gain in information, in this paper, it is shown that accuracy can still be exploited by reducing variance of the virtual samples. Moreover, this approach can be easily generalised to other pattern recognition problems.

An alternative approach to creating variations due to geometric transformation is to synthesize virtual images from an approximated user-customized 3D model. This approach, although maybe more effective than the proposed method, is not considered here due to the possible inaccuracy of approximating the model in the first place. Our approach does not require such an estimation. The rest of this paper is organised as follows: Section 2 explains the theoretical bounds in the expected gain coming from averaging scores; a description of the experiment can be found in Section 3; this is followed by conclusions.

2 VARIANCE REDUCTION VIA AVERAGING

2.1 Variance reduction

Let us assume that the measured relationship between a feature vector $\mathbf{x}_i$ and its associated score y_i can be written as:

$$y_i = f(\mathbf{x}_i) + \eta_i. \tag{1}$$

where $f(\cdot)$ is the true relation and η_i is a random additive noise with zero mean. The mean of y over N trials, denoted as $\bar{y}$ is:

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

With enough samples, the expected value of y, denoted as $E[y]$, which is estimated by the mean of y, approximates the "true" measure:

$$E[y] \quad = \quad E[f(\mathbf{x})] + E[\eta] \tag{3}$$

$$= f(\mathbf{x}). \tag{4}$$

Moreover, the variance of y can be written as:

$$\mathrm{Var}[y] = \frac{1}{N}\mathrm{Var}[\eta] \tag{5}$$

Therefore, it can be concluded that when N scores of a single biometric source are averaged, noise that occurs due to classification can be reduced by a factor of N. The effect of averaging in Equation 2 can best be observed using synthetically generated data in Figure 1. Assume that in the original problem, the genuine user scores follow a normal distribution of mean 1.0 and variance 0.9, denoted as $\mathcal{N}(1, 0.9)$, and that the impostor scores follow a normal distribution of $\mathcal{N}(-1, 0.6)$ (both graphs are plotted with '+'). If for each access, three confidence scores are available, according to Equation 5, the variance of the resulting distribution will be reduced by a factor of three. Both resulting distributions are plotted with 'o'. Note the area where both the distributions cross before and after. This area corresponds to the zone where minimum amount of mistakes will be committed given that the threshold is optimal [1]. The decrease in this area means an improvement in the recognition rate. In general, the more samples

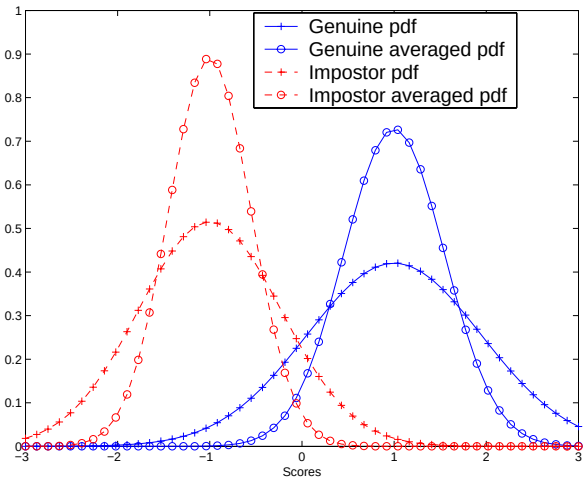


Figure 1: Averaging scores distribution in a two-class problem

are used, the sharper (taller and with shorter tails at both ends) both the impostors' and the clients' score distributions become. The sharper they are, the lower the area where these two distributions overlap. The lower this area is, the lower the number of mistakes committed.

2.2 Error reduction

The above discussion is only true when scores are corrupted by noise with zero-mean and uncorrelated. In reality, one knows that scores coming from virtual samples are dependent on the original image. What would then be the upper and lower bounds of such a gain? Here, we refer to the work of Bishop [2, Chap. 9] who has shown that by averaging scores of N classifiers, a committee could perform better than a single classifier. The assumptions were that each classifier was not correlated and that the error of each classifier had zero mean. He showed that:

$$\mathrm{err}_c = \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{err}_i \tag{6}$$

$$= \frac{1}{N}\mathrm{mean}(\mathrm{err}_i). \tag{7}$$

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

where err_c is the error of the committee and err_i is the error associated to the i-th classifier. Note that the major difference between Bishop's context and ours is that scores are due to variation of N classifiers. In our context, scores are due to variation in the "virtual" samples obtained from N geometric transformations. The index i is referred to a sample hereinafter.

Due to the false assumption of uncorrelation in scores obtained from virtual samples, the error reduction obtained using the mean operator will not be N as shown in Equation 7 but less. This equation should be rightly written as:

$$\mathrm{err}_c = \frac{1}{\alpha}\mathrm{mean}(\mathrm{err}) \qquad 1 \leq \alpha \leq N. \tag{8}$$

where α can be understood as a "gain" in error reduction. It shows that the maximum gain in averaging scores is N with respect to the average performance of each virtual sample. This is, in practice, not attainable since the scores are correlated. The minimum gain, according to Equation 8 is 1, which means that there is no gain *but one does not loose* in the combination neither. This can be understood as follows: If the errors made by each virtual score are dependent, i.e., they make exactly the same error in the extreme case ($\forall_{i,j}(\mathrm{err}_i = \mathrm{err}_j)$), then $\mathrm{mean}(\mathrm{err}) = \mathrm{err}_i = \mathrm{err}_c$, which implies that $\alpha = 1$.

As in the case of committee of classifiers, by averaging N scores from N transformed images, the gain factor in terms of error reduction with respect to a single input image is in the range $[1, N]$. Therefore, score averaging is a simple yet effective way to increase system accuracy.

3 EXPERIMENT

3.1 Database and Protocols

The XM2VTS face database is used for this purpose because it is a benchmark database with well-defined protocols called the Lausanne Protocols [3]. The XM2VTS database contains synchronized image and speech data recorded on 295 subjects during four sessions taken at one month intervals. On each session, two recordings were made, each consisting of a speech shot and a head rotation shot.

The database was 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 was used to compute the decision (by estimating thresholds for instance, or parameters of a fusion algorithm). Finally, the test set was used only to estimate the performance of the system.

The 295 subjects were divided into a set of 200 clients, 25 evaluation impostors, and 70 test impostors. Two different evaluation configurations were defined. They differ in the distribution of client training and client evaluation data. Both the training client and evaluation client data were drawn from the same recording sessions for configuration I (LP1) which might lead to biased estimation on the evaluation set and hence poor performance on the test set. For configuration II (LP2) on the other hand, the evaluation client and test client sets were drawn from different recording sessions which might lead to more realistic results. More details can be obtained from [3].

In this database, each access is represented by only one face image. We can increase the number of images by using geometric transformations. In this way, we obtain multiple "virtual" samples from a single access. For each virtual image, features will be extracted in the same way as a real face image. Both feature extraction and geometric transformations are explained in sections below.

3.2 Features

In the XM2VTS database, a bounding box is placed on a face according to eyes coordinates located manually. This assumes a perfect face detection. The face is cropped and the extracted sub-image is downsized 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. To construct this additional feature set, a skin colour look-up table must first be constructed using a large number of colour images which contain only skin. In the second step, face images are filtered according to this look-up table. Unavoidably, non-skin pixels are captured as well. This noise will be submitted to a classifier to discriminate its degree of relevance. For each color 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 [4].

3.3 Geometric Transformations

The extended number of patterns is computed such that given an access image, N geometric transformations are performed. This number is calculated as follows: $N = 2 \times A \times B$, which shows the mirrored number of shifted and scaled face patterns. $A =$ number of shifts $\times\, 8 + 1$ is the total number of shifts, in 8 directions, including the original frame, for each scale. $B =$ number of scales $\times\, 2 + 1$ is the total number of scales, in 2 directions (zooming-in and zooming-out), including the original scale. In the experiment, 4 shifts and 2 scales are used. This produces 330 virtual images per original image.

In the following experiments, we compared the system from [4] (denoted "original") to our system (denoted "averaged"). In the original system, geometric transformations were added to the training set only, while in the averaged system, they were also added to the evaluation and test sets.

The training set is used to train an MLP for each client and the evaluation set is used to stop the training using an early-stopping criterion. At the end of training, the trained MLP model is applied on the evaluation set again to estimate the global threshold that optimises the Equal Error Rate (EER). Once all parameters are set, including threshold, the trained MLP model is applied on the test set. Thus the obtained Half Total Error Rate (HTER) on the test set is said to be *a priori*, while if the threshold was optimising EER on the test set, it would be called *a posteriori*. Of course, the *a priori* results are more realistic. In the experiment, the optimised client dependent MLPs had 20 hidden units each.

3.4 Results

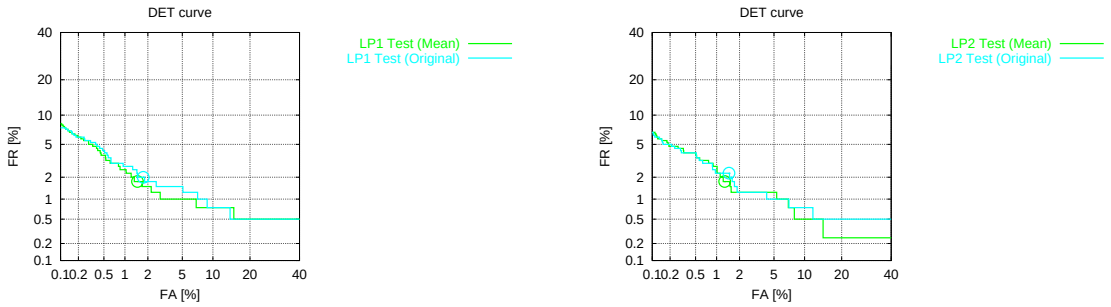
The experiments are carried out on LP1 and LP2 configurations of XM2VTS database. The results are shown in Tables 1 and 2. Odd lines in these tables show the HTERs of the original approach while even lines show the HTERs after averaging virtual scores. In all comparisons, the improvements are obvious. The HTERs in Table 1 are *a posteriori* and thus not realistic, but nevertheless give insights of the expected improvements. The HTERs in Table 2 are *a priori*. The corresponding DET curves of Table 2 are shown in Figure 2. As expected, the performance obtained by averaging is always superior. Moreover, to the best of our knowledge, the newly obtained *a priori* results appear to be the best published ones on this benchmark database.

Table 1: Performace of averaging scores versus original approach based on *a posteriori* selected thresholds

Data sets	Models	FA[%]	FR[%]	HTER[%]
LP1 Eval	Original	1.667	1.667	1.667
LP1 Eval	Averaged	1.333	1.333	1.333
LP2 Eval	Original	1.250	1.250	1.250
LP2 Eval	Averaged	1.107	1.000	1.054
LP1 Test	Original	1.817	1.750	1.783
LP1 Test	Averaged	1.692	1.750	1.721
LP2 Test	Original	1.726	1.750	1.738
LP2 Test	Averaged	1.514	1.500	1.507

Table 2: Performace of averaging scores versus original approach based on *a priori* selected thresholds

Data sets	Models	FA[%]	FR[%]	HTER[%]
LP1 Test	Original	1.230	2.750	1.990
LP1 Test	Averaged	1.474	1.750	1.612
LP2 Test	Original	1.469	2.250	1.860
LP2 Test	Averaged	1.285	1.750	1.518



(a) LP1 configuration (b) LP2 configuration

Figure 2: Test sets on XM2VTS database

3.5 Analysis of virtual distribution scores

One insight to examine the effectiveness of this method is by looking at the probability density function (*pdf*) of the 330 virtual scores with respect to a false rejection and a correct acceptance. This is shown in Figure 3. When given an upright-frontal image of a client within a certain allowed degree of transformation, one obtains a sharply picked *pdf* (with very low variance) around the mean 1. The MLP associated with client 006, in this case, was trained to give a response of 1 for a genuine access and -1 for an impostor access. When the original image is "out" of the allowed transformation range, the *pdf* of virtual scores has a large variance and a mean displaced away from 1. Note that the logarithmic scale for the probability is used in the graph to amplify the changes in distribution accross the score range $[-1, 1]$.

While a single image normally produces only one score, a set of virtual images has the advantage of producing another information: the score distribution. One way to measure this distribution is by its variance. For instance, for the example above, the variance for the correct acceptance case is 1.5670e-05 while the variance for the false rejection case is 0.0181. Clearly, variance of virtual scores can give supplementary information that the original approach cannot. In general, the *pdf* (not just the variance) could probably provide useful insights to improve this method further.

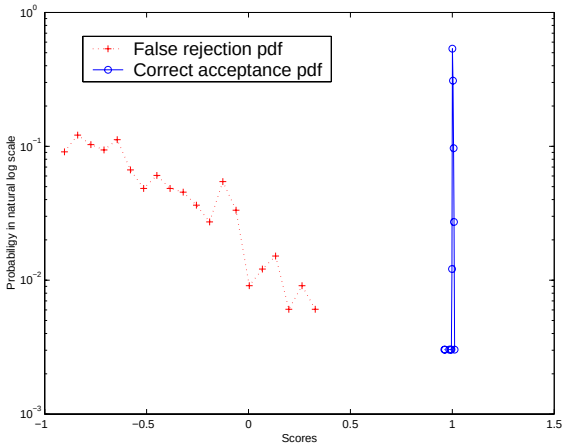
3.6 Variance and error reduction

This section tries to examine the relationship between the reduction of both variance and error. The hypothesis here is that, when N ($N = 330$ in our case) virtual scores are averaged, Equation 5 says that the reduction is by a factor of N, assuming that the scores are independent. They are unfortunately not in our case. To measure the degree of independance, we introduce a *variance reduction ratio*, defined as:

$$\beta = \frac{Var[y_v]}{Var[y]} \tag{9}$$

(a) False rejection

(b) Correct acceptance



(c) Corresponding histograms

Figure 3: Examples of "bad" and "good" photos and their corresponding distribution of virtual scores for client 006

where y are either client or impostor scores from the original method and y_v are either client or impostor virtual scores. These values are shown in Table 3.

Table 3: The gain factor β between the scores of virtual samples and that of original samples.

Data sets	*Pdf*s of access type	Gain factor β	
		LP1	LP2
Eval	Client *pdf*	1.2716	1.2561
Eval	Impostor *pdf*	1.0960	1.0769
Test	Client *pdf*	1.1689	1.2675
Test	Impostor *pdf*	1.1642	1.0507

In all cases, $\beta > 1$. Unfortunately, β is very close to 1 and very far from N. This is expected because of strong depenancy of virtual scores. In all cases, variances of each data set (client and impostor accesses) are reduced systematically.

How about the gain factor of HTER? These values are readily available from Table 1 by dividing the odd lines HTER by the corresponding even line HTER. The definition of α can be derived from Equation 8. The error reduction for each set of Evaluation and Test data in both LP1 and LP2 configurations are shown in Table 4.

Table 4: The gain factor of error reduction according to Table 1

Data sets	Gain factor
LP1 Eval	1.251
LP2 Eval	1.186
LP1 Test	1.010
LP2 Test	1.153

Note that the variance reduction (Table 3) and error reduction (Table 4) are somewhat proportional. In general, if there is a reduction in variance of client or impostor *pdf*, there will be a reduction

in classification error (specifically HTER in our case). To our opinion, it is necessary to investigate this "intuition" further.

As can be observed, these gain factors are very close to the lower bound, i.e., 1, which means that the gain is very little. This may be due to the high correlation among virtual scores. Nevertheless, the fact that improvement is guaranteed makes our approach still very attractive.

Finally, an appropriate question to ask is: by how much the virtual samples approach method wins over the original real samples approach? To answer this question, we literally computed the total error (sum of false acceptance and false rejection errors) for both methods. The difference between these two erros, i.e., $err(\theta) - err_v(\theta)$ are plotted in Figure x.

4 CONCLUSION

By applying N geometric transformations to a given original face image access, it is shown that one could reduce the variance of the original score by a factor of N. Furthermore, by taking into account the assumption that these N image samples are dependent on the original image, the classification error, with respect to the original method is shown to reduce by a factor between 1 and N.

To put in a formal framework, our proposed approach can be summarised as:

$$y = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} f(h(g(\mathbf{x}, t))) \tag{10}$$

instead of $y = f(h(\mathbf{x}))$ for the test set, where, $t \in \mathcal{T}$ is a set of geometric transformation parameters applied by g (the transformation function) on the feature vector $\mathbf{x}$, h is a feature extraction function and f is a trained classifier on $h(f(\mathbf{x}, t))$ over $t \in \mathcal{T}$ with $\mathbf{x}$ sampled from a training set. Equation 10 explains why this method is robust against minor geometric transformations: it is integrated over the space of these transformations and hence achieves invariance over this space.

This method has the advantage of being simple to implement. Furthermore, it does not require multiple real examples. This makes it easily extendable to many general classfication and regression problems. The only added complexity during testing is proportional to the number of artificially generated samples, given that a suitable transformation for a given data set can be defined.

The future work will consist of proposing a theoretical model to understand the necessary criteria and conditions for averaging samples to work. Right now, the relationship between variance reduction and error reduction have not thoroughly been investigated. Such analysis will eventually show the criteria of success or failure of this approach, i.e., when the performance degrades.

References

[1] J. Kittler, G. Matas, K. Jonsson, and M. U. R. Sanchez. *Combining Evidence in Personal Identity Verification Systems.* Pattern Recognition Letters, 18(9):845–852, September 1997.

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

[3] J. Lüttin, *Evaluation protocol for the XM2FDB Database (Lausanne Protocol)*, IDIAP Research Report, COM-05, 1998.

[4] S. Marcel and S. Bengio, *Improving Face Verification using Skin Color Information*, Proceedings of the 16th International Conference on Pattern Recognition, 2002.

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

APPENDIX

This section is a follow-up based on the observation reported before. It describes several attempts to combine virtual scores. These methods includes using median operator and GMM (Gaussian Mixture Model) and entropy methods. The result based on the mean operator is analysed in details, particularly in comparaison with the original approach (using only real samples).

5 Combining virtual scores

It has been shown that the averaging scores from virtual samples can increase the performance. However, it is not clear how the distribution or density of scores from virtual samples can be used. This section proposes several methods to do so:

- Median operator. It is the closest operator to the mean operator and is known to be robust against out-liers.
- Entropy method based on global models. The *pdf* of a global client and impostor scores are first estimated. These *pdf*s are then compared to the *pdf* of a given access. The *pdf* of a given access can be calculated from all the virtual scores associated to an access request. The authentication task then becomes matching of two *pdf*s using the Kullback distance.
- Entropy method based on local models. This is similar to the global models except that local models are used. Local models means models estimated only from client or impostor accesses associated to a given user-specific classifier.
- Gaussian Mixture Model. GMM is very useful for matching sequences which are assumed to have been derived from identically and independently distribution. Virtual scores can be regarded as coming from a certain form of distribution. This distribution can be estimated using a mixture (weighted sum) of Gaussians. During an access, set of virtual scores that is obtained can be regarded as a sequence. This sequence is then matched to the GMM computed *a priori* to obtain the likelihood. Two GMM models are needed: GMMs associated to the client and to the impostor.

The entropy methods and GMM are explained in the following sections.

5.1 Entropy method

The entropy method requires that the density of the data be estimated. There are several ways to estimate the density according to [2, Chap. 2]: histrogram, Parzen window and GMM. These methods receive a set of data and output a density function. Histogram suffers from having the need to define the length of each bin. Larger bins may produce smoother density estimate but does not give accurate estimate on the density of a given value y. Thus, Parzen window and GMM are used. Parzen window is described below and GMM is discussed further.

5.1.1 Density estimation using Parzen window

Given a set of scores $y_i, i = 1, \ldots, N$, its density function can be calculated using:

$$\tilde{p}(y') = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h} H\left(\frac{y' - y_i}{h}\right) \tag{11}$$

where $H(u)$ is a kernel function taking a scalar u. When $H(u)$ is a Gaussian function, Equation 11 becomes:

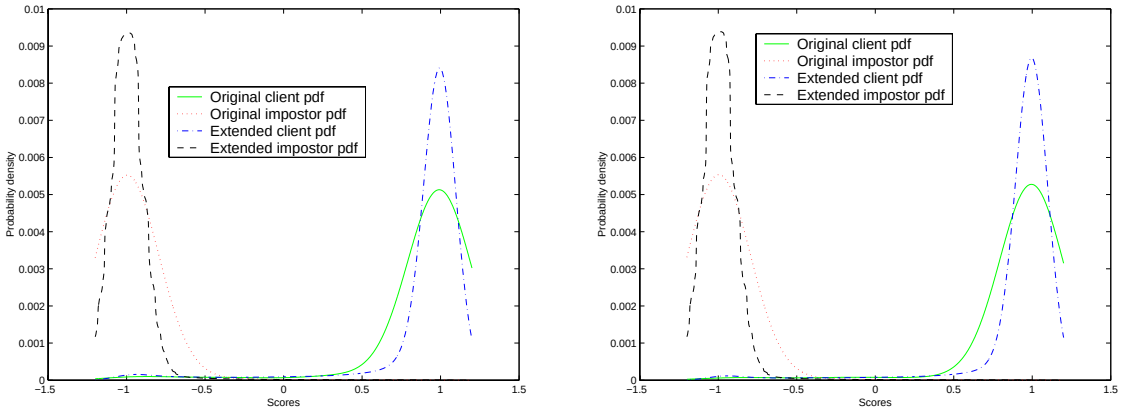
$$\tilde{p}(y') = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{(2\pi h^2)^{1/2}} \exp\left\{\frac{y' - y_i}{h}\right\} \tag{12}$$

In practice, y' is sampled within two bounded values. For our case, y' is bounded within $[-1.2, 1.2]$ because the MLPs used are trained to give scores between -1 and 1. Therefore, values outside the range are not very useful. 1000 samples of y' within the said range are sampled for each client and impostor *pdf* over the evaluation sets of LP1 and LP2. The parameter h, which corresponds to the variance of distribution, controls the smoothness of the resultant *pdf*.

We have attemped to use cross validation to estimate an optimal h value. In each fold of cross-validation, one held-out set is used for test while the rest are used for training. The training set y_i is used to estimate $\prod_j \tilde{p}(y'_j)$ with y'_j as scores from the test set. The goal is to use several h values that minimises the product. Unfortunately, such cross-validation cannot be established when h takes on smaller and smaller values (equivalent to higher capacity) because y and y' are very similar. In fact, these scores (y and y') are extremely concentrated at $+1$ for the client and -1 for the impostor scores. As a consequence, h is fixed arbitrarily to 0.2.

In actual implementation, the negative sum of logairthm is used to overcome the computation precision problem, i.e., $-\sum_j \ln \tilde{p}(y'_j)$. This is because the product of several small values will eventually lead to zero in finite precision.

The larger h is, the smoother the resultant *pdf*. Note that this method is similar to histogram except that it gives a smoother estimation of *pdf*. Furthermore, another major differnce is that the Parzen window method have bins centered around the data point, contrary to histogram which has a fixed bin. The resultant *pdf*s are shown in Figure 4. Note that in both protocol configurations, the



(a) LP1 (b) LP2

Figure 4: The client and impostor *pdf*s of the evaluation sets of LP1 and LP2 configuration

pdf of the original client and that of its extension (with virtual method) are not comparable because the extended method has 330 times more data than the original

5.1.2 Entropy method based on global models

Entropy is used to compare two *pdf*s from a set of virtual scores Y. In our case, one *pdf* comes from a global model (client or impostor), denoted as $p(Y)$, and the other *pdf* comes from another set of virtual scores, denoted as $q(Y)$. Both *pdf*s are estimated by the Parzen window method described earlier. Both *pdf*s are sampled at same i-th location in the score space. This can be denoted as y_i. The entropy of a given access distribution $q(Y)$ can then be defined as:

$$L(p,q) = -\sum_i p(y_i) \ln \frac{q(y_i)}{p(y_i)}. \tag{13}$$

Entropy can be regarded as a distance as to how much $q(y)$ is similar to $p(y)$ but not the other way round, i.e., this distance is not symetric. This alone does not give discriminative information. To

do so, entropy of a client and impostor model should be used together. Let $L(p_{w1}, q)$ be the entropy of $q(y)$ with respect to a client model and $L(p_{w2}, q)$ be that of $q(y)$ with respect to an impostor model. Then the distance between these two entropy can be defined as:

$$\triangle = L(p_{w2}, q) - L(p_{w1}, q) \tag{14}$$

$\triangle > 0$ means that the entropy of an impostor model is more than that of a client. Therefore $\triangle > 0$ reflects how likely a set of virtual scores belong to a client.

5.1.3 Entropy method based on local models

Instead of using two global models to represent a client and impostor score *pdf*, one can also represent a client and impostor score *pdf* for each client. This can be done by replacing p_{w1} and p_{w2} in Equation 14 to two local models as follows:

$$\triangle = L(p^n_{w2}, q) - L(p^n_{w1}, q), \tag{15}$$

where n is an index unique to a client.

One possible problem with this approach is that one does not have enough data to estimate p_{w1} correctly, because client accesses are limited for training.

5.1.4 Gaussian Mixture Model

Given a claim for genuine client w_1's identity and a set of N virtual scores $Y = \{y_i\}_{i=1}^N$ supporting the claim, the average log likelihood of the claimant being the true claimant is calculated using:

$$\mathcal{L}(Y|\lambda_{w1}) = \frac{1}{N}\sum_{i=1}^{N} \log p(y_i|\lambda_{w1}) \tag{16}$$

$$\text{where} \quad p(Y|\lambda) = \sum_{j=1}^{M} m_j \; \mathcal{N}(y; \mu_j, \sigma_j) \tag{17}$$

$$\text{and} \quad \lambda = \{m_j, \mu_j, \sigma_j\}_{j=1}^{M} \tag{18}$$

Here λ_{w1} is the model for person w_1. M is the number of mixtures, m_j is the weight for mixture j (with constraint $\sum_{j=1}^{M} m_j = 1$), and $\mathcal{N}(y; \mu, \sigma)$ is a multi-variate Gaussian function with mean μ and variance σ:

$$\mathcal{N}(y; \vec{\mu}, \sigma) = \frac{1}{\sqrt{(2\pi\sigma^2)}} \exp\left[\frac{-(y-\mu)^2}{2\sigma^2}\right] \tag{19}$$

The number of mixture of Gaussian components are estimated using 5-fold cross-validation from a giving training set.

The impostor model is constructed in a similar way accoring to Equations 16, 17 and 18, with Y as all scores belonging to impostors w_2.

An opinion on the claim is found using:

$$\triangle(Y) = \mathcal{L}(Y|\lambda_{w1}) - \mathcal{L}(Y|\lambda_{w2}) \tag{20}$$

The opinion reflects the likelihood that a given claimant is the true claimant (i.e., a low opinion suggests that the claimant is an impostor, while a high opinion suggests that the claimant is the true claimant).

Table 5: Different combination methods of virtual scores on LP1.

Method	HTER		
	Evaluation	Test *a posteriori*	Test *a priori*
Original	1.667	1.783	1.875
Mean	1.333	1.721	1.612
Median	1.667	1.750	1.667
GMM	1.518	1.741	1.709
Global Entropy	1.333	1.734	1.606
Local Entropy	$*$ 0.499	3.000	2.186

(*) indicates biased estimate.

Table 6: Different combination methods of virtual scores on LP2.

Method	HTER		
	Evaluation	Test *a posteriori*	Test *a priori*
Original	1.250	1.738	1.737
Mean	1.054	1.507	1.518
Median	1.238	1.750	1.547
GMM	1.034	1.500	1.493
Global Entropy	1.218	1.500	1.559
Local Entropy	$*$ 0.251	2.500	2.043

(*) indicates biased estimate.

5.2 Experiment Results

Using the discussed methods, experiments are carrried out on LP1 and LP2 protocols. The results are shown in Table 5

Except the local entropy method, all the methods improve the original approach. GMM seems to perform the best in LP2 protocol but among the worse in the LP1 protocol. It is therefore difficult to judge the best combination method. It is surprsing to see that the mean operator which is a simple method, is among the best way to merge the virtual scores in both protocols.

6 Half total error rate and classification error rate revisited

In biometric authentication, HTER is often used as an important criteria. In this section, we wish to clarify between these two types of errors as evaluation criteria.

Let us define false rejection with respect to a threshold θ as follows:

$$FR(\theta) = \|\{y|y \in w_1 \wedge y < \theta\}\|, \tag{21}$$

where w_1 is client class and $\| \bullet \|$ is the cardinality (number of elements) of $\bullet$. Similaryly, false acceptance with respect to a threshold θ can be defined as:

$$FA(\theta) = \|\{y|y \in w_2 \wedge y \geq \theta\}\|, \tag{22}$$

where w_2 is an impostor class.

In other words, a client score is considered a false rejection case when it is below a threshold. Similarly, an impostor score is considered a false acceptance case when it is above or equal a given threshold.

We now try to relate these two equations to its probability density, assuming that it is known for both the client and impostor classes. $FR(\theta)$ can be written as:

$$FR(\theta) = \|w_1\| \int_{-\infty}^{\theta} p(y|w_1)dy. \tag{23}$$

Similarly, $FA(\theta)$ can be written as:

$$FA(\theta) = \|w_2\| \int_{\theta}^{\infty} p(y|w_2)dy. \tag{24}$$

According to Bayesian rule, the probability of commiting error, (denoted as ERR hereinafter), by taking into account of class prior, can be written as:

$$ERR(\theta) = \int_{-\infty}^{\theta} p(y|w_1)dy \times P(w_1) + \int_{\theta}^{\infty} p(y|w_2)dy \times P(w_2). \tag{25}$$

An intuitive way to understand the probability of error is that the density of false rejection and that of false acceptance are weighted by $P(w_1)$ and $P(w_2)$, where the class priors (weights) sum to one, i.e., $P(w_1) + P(w_2) = 1$.

By making use of Equations 23 and 24, we can rewrite Equation 25 as:

$$ERR(\theta) = \frac{FR(\theta)}{\|w_1\|} \times P(w_1) + \frac{FA(\theta)}{\|w_2\|} \times P(w_2). \tag{26}$$

In biometric authentication, $P(w_1)$ and $P(w_2)$ are often unknown, or assumed to be unknown (such is the case during testing: $P(w_1)$ are $P(w_2)$ are known under laboratory condition but are deliberately assumed to be unknown). In either situation, $P(w_1) = P(w_2) = \frac{1}{2}$. Therefore, Equation 25 can be simplified as:

$$HTER(\theta) = \frac{1}{2}\left(\frac{FR(\theta)}{\|w_1\|} + \frac{FA(\theta)}{\|w_2\|}\right). \tag{27}$$

This error is commonly called Half Total Error Rate (HTER).

To compare HTER and probability of classification error (ERR), ERR can be rewritten as:

$$\begin{aligned} ERR(\theta) &= \frac{FR(\theta)}{||w_1||} \times \frac{||w_1||}{||w_1|| + ||w_2||} + \frac{FA(\theta)}{||w_2||} \times \frac{||w_2||}{||w_1|| + ||w_2||} \\ &= \frac{FR(\theta) + FA(\theta)}{||w_1|| + ||w_2||}, \end{aligned} \tag{28}$$

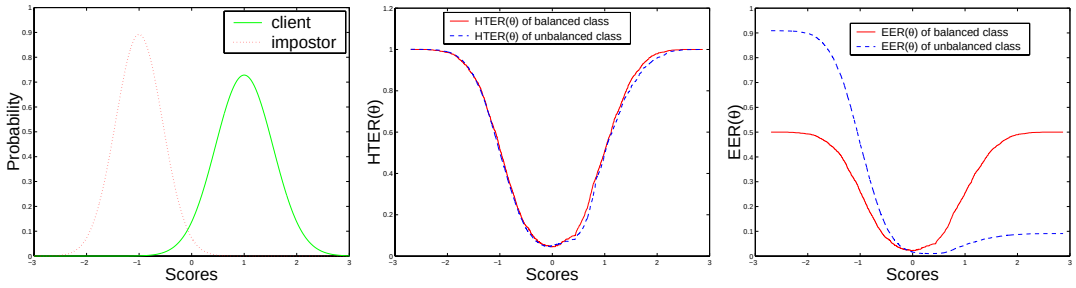
using the knowledge that $P(w_1) = \frac{||w_1||}{||w_1||+||w_2||}$ and $P(w_2) = \frac{||w_2||}{||w_1||+||w_2||}$.

To give an idea how these two errors behave, we have generated client and impostor sets of scores artificially. The client has a density distribution of $\mathcal{N}(1, 0.3)$ (mean 1; variance 0.3) while the impostor has a density of $\mathcal{N}(-1, 0.2)$. These two distribution functions are shown in Figure 5(a).

For the first case (called balanced class configuration), the client and impostor sets have 1000 access scores respectively. For the second case (called unbalanced class configuration), the client set has 1000 scores while the impostor set has 10000 accesses, i.e., unbalanced by a factor of 10. The HTER and ERR curves (as a function of threshold θ) of these two cases are shown in Figure 5(b) and (c). Note that, due to unblanaced class prior, the ERR is affected while HTER is not.

Note that, in reality, errors committed in FA and FR have different costs. Let C_{FA} and C_{FR} be the cost of FA and FR, respectively. Then, Equations 27 and 28 can be written in terms of cost as:

$$C_{HTER}(\theta) = \frac{1}{2}\left(\frac{FR(\theta)}{||w_1||} \times C_{FR} + \frac{FA(\theta)}{||w_2||} \times C_{FA}\right) \tag{29}$$



(a) *pdf* (b) HTER(θ) (c) ERR(θ)

Figure 5: Artificially generated scores and their HTER(θ) and EER(θ) curves

and

$$C_{EER}(\theta) = \frac{FR(\theta) \times C_{FR} + FA(\theta) \times C_{FA}}{||w_1|| + ||w_2||} \tag{30}$$

respectively.

These two types of error (or cost) have an important impact on our result, to be described in the next section.

7 Analysis of results using the mean operator

By using the HTER criteria from Equation 27, we explicitly calculated the HTER as a funciton of θ for the evalluation set of LP1 on experiment using original samples and virtual samples combined with the mean operator. The HTER curves is shown in Figure 6(a). This graph has a form similar to Figure 5(b), as expected. However, it gives little information on how much one method wins over the other method. To visualise this information, we introduce the diffrence of HTER as:

$$\triangle(\theta) = C^{ORL}(\theta) - C^{VIR}(\theta), \tag{31}$$

where C^{ORL} is the cost of using original samples and C^{VIR} is the cost of using virtual samples. For both cases, the cost could be evaluated using HTER criterion ($C_{HTER}(\theta)$) or ERR criterion ($C_{ERR}(\theta)$). Distinctions are made here because they give different results.

7.1 Equal cost assumption

In the following section, the costs of FA and FR are assumed equal, i.e., 1. Different values of these costs will be used later.

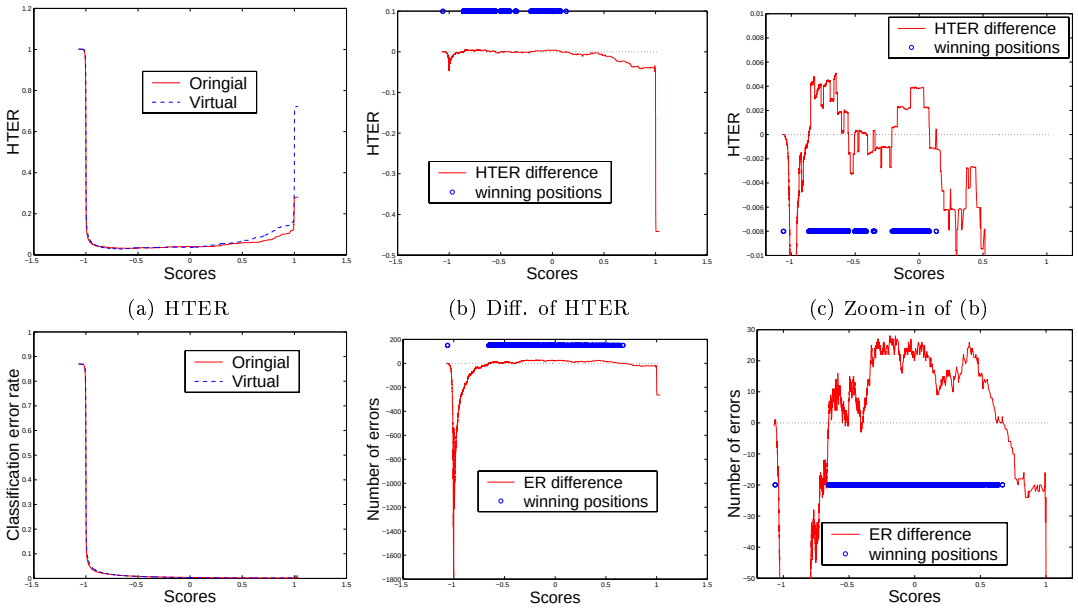
The difference according to HTER criteria, $\triangle_{HTER}(\theta)$, is shown in Figure 6(b). Figure 6(c) shows a zoom-in version of (b). The blue circles show the position where $\triangle_{HTER}(\theta)$ is positive, i.e., where $C^{VIR}_{HTER}(\theta)$ is smaller than $C^{ORL}_{HTER}(\theta)$, which is desirable.

When using the ERR as criteria, according to Equation 28, the ERR curves for both the original and virtual methods are shown in Figure 6(d). As expected, it is similar to Figure 5(c), with their tails not easily perceived due the highly unbalanced class prior. In this particular data set (LP1 Evaluation set), the client set has 600 scores and the impostor set has 40000 scores.

The cost difference of both the original and virtual methods, as a function of threshold θ are shown in Figure 6(e). The zoom-in version of it is shown in Figure 6(d).

Comparing HTER (Figure 6(a-c)) and EER (Figure 6(d-f)) criteria, one can observe that the virtual methods wins over the original method using the ERR criteria because the winning positions

are almost always continuous within two bounds $[a, b]$, where $a > -1$ and $b < 1$. This is unfortunately not the case for HTER criteria.



(a) HTER (b) Diff. of HTER (c) Zoom-in of (b)

(d) ERR (e) Diff. of ERR (f) Zoom-in of (e)

Figure 6: Comparaison of the original and virtual methods using mean operator using HTER and classification error (ERR) criteria based on the evaluation data set of LP1. There are 40000 impostor accesses comparing to 600 client accesses.

7.2 Explanation for winning bound in ERR and HTER

We believe that these winning positions bounded in $[a, b]$ for EER criterion are not a conincidence. The same is true for the discontinued winning positions for the HTER criterion.

To analyse this behaviour, it is desirable to know "how much contribution a FA and a FR is to the overall cost". This quantity is just the derivative of the cost. For both the HTER and ERR criteria, their derivatives can be calculated from Equation 29 and 30 as:

$$\frac{\delta C_{HTER}(\theta)}{\delta\theta} = \frac{C_{FR}}{2||w_1||}\frac{\delta FR(\theta)}{\delta\theta} + \frac{C_{FA}}{2||w_2||}\frac{\delta FA(\theta)}{\delta\theta} \tag{32}$$

and

$$\frac{\delta C_{ERR}(\theta)}{\delta\theta} = \frac{C_{FR}}{||w_1|| + ||w_2||}\frac{\delta FR(\theta)}{\delta\theta} + \frac{C_{FA}}{||w_1|| + ||w_2||}\frac{\delta FA(\theta)}{\delta\theta} \tag{33}$$

respectively.

In biometric application, $||w_2|| \gg ||w_1||$. If everything else cosidered equal, i.e., $C_{FA} = C_{FR} = 1$ and , $|\frac{\delta FA(\theta)}{\delta\theta}| \approx |\frac{\delta FR(\theta)}{\delta\theta}|$, for the case of HTER criterion, increase of one FR contributes $\frac{1}{2||w_1||}$ to the cost while increase of one FA contributes $\frac{1}{2||w_2||}$. Obviously, contribution of FA is downplayed by the factor $\frac{1}{2||w_2||}$ because $\frac{1}{2||w_2||} \ll \frac{1}{2||w_2||}$.

This minimum cost can be found by setting the derived cost function to zero. We will study only the HTER criterion because it is more relevant in this application. Setting Equation 32 to zero, together with Equation 23 and 24 gives:

$$0 = \frac{C_{FR}}{2||w_1||}\frac{\delta FR(\theta)}{\delta\theta} + \frac{C_{FA}}{2||w_2||}\frac{\delta FA(\theta)}{\delta\theta}$$

$$0 = \frac{C_{FR}}{2||w_1||}\frac{\delta}{\delta\theta}\left(||w_1||\int_{-\infty}^{\theta} p(y|w_1)dy\right) + \frac{C_{FA}}{2||w_2||}\frac{\delta}{\delta\theta}\left(||w_2||\int_{-\infty}^{\theta} p(y|w_2)dy\right)$$

$$0 = \frac{C_{FR}}{2}p(\theta^*|w_1) + \frac{C_{FA}}{2}p(\theta^*|w_2) \quad (34)$$

Indeed, there exists one single threshold θ^* that optimises the cost criteria. The wider the score space where the virtual method wins over the original method (the winning positions), the higher the probability that the virtual method will win. This is because θ^* will have higher probability of falling into one of these winning positions.

7.3 Unequal cost assumption

What if the cost of FA and FR are different? From Equation 32, intuitively, one could predict that high C_{FA} will increase favourably the contribution of $\frac{\delta FA(\theta)}{\delta\theta}$ which is downplayed by very large $||w_2||$. Indeed, this is often the case because false acceptance is often very serious in high security application. According to the NIST standard, $C_{FA} = 10$ while $C_{FR} = 1$.

Using these convention, we calculated the cost according to HTER for the case ($C_{FA} = 10$, $C_{FR} = 1$) and ($C_{FA} = 10$, $C_{FR} = 1$). HTER criterion is used because it is a more realistic and relevant criterion in biometric authentication then the ERR criterion. The cost curves are shown in Figure 7.

In real application, the threshold θ takes on a value that optimises best the cost function. We calculated the optimal threshold θ^* together with its corresponding minimum cost of HTER. The results are shown in Table 7.

Table 7: Different combination methods of virtual scores on the evaluation set of LP1.

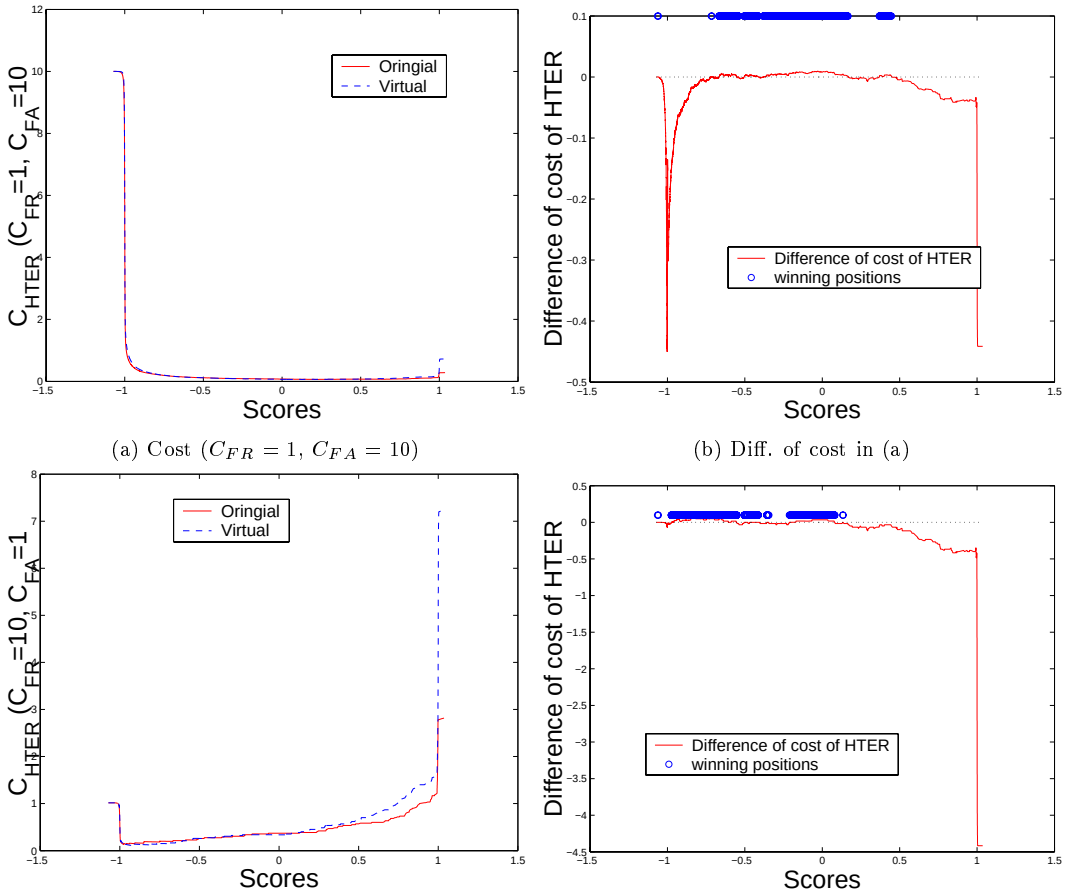
C_{FR}	C_{FA}	Original Method		Virtual Method	
		Min. cost	θ^*	Min. cost	θ^*
1	1	0.0313	-0.5910	0.0280	-0.6503
1	10	0.0641	0.2322	0.0635	0.0759
10	1	0.1343	-0.9754	0.1095	-0.9221

On all three different assumption of costs of FA and FR, the virtual method seems to be robust. Although the virtual method does not garantee to win over the original method at all score space, it is at least better when using the optimal threshold.

It is indeed exciting to observe in Figure 7(a) and (b) how the virtual method wins over the original method. High cost of FA gives favourable result to the virtual method. Inversely, high cost of FR gives favourable result to the orignal method. However, even in this disadvantage situation, Figure 7(d) shows that the virtual method still wins over the original method with a very narrow bound of $[a, b]$ values.

It should be emphaised here that the threshold θ does not take on any value. It takes a specific value that minimises the cost. As long as this optimal threshold, θ^*, falls within $[a, b]$, then the virutal method will always be beneficial.

One possible explanation to why there exist a bound $[a, b]$, often continuous, but not necessarily so, where the virtual method will win over the original method is due to the reduction of variance in using multiple virtual samples comparing to the original method. When variance reduces for both the client and impostor *pdf*s, the peak of distribution will become higher than those of the original *pdf*s. The tails, on the other hand, will be longer and thiner comparing to the original *pdf*s. As a result, the overlaping regions, where errors are made, reduces. When computing the difference between the cost function of the original and the virtual method, the virtual method wins over the original method at the area (the winning positions) where both the client and impostor overlaps. This intutively shows



(a) Cost ($C_{FR} = 1$, $C_{FA} = 10$)

(b) Diff. of cost in (a)

(c) Cost ($C_{FR} = 10$, $C_{FA} = 1$)

(d) Diff. of cost in (c)

Figure 7: Comparaison of the original and virtual methods merged using mean operator according to the cost of HTER, based on the evaluation data set of LP1

why the virtual method is useful. In practice, we found that such winning positions are not often continuous due to the different class prior and cost of FA and FR, according to the HTER criteria.

It is well known in the problem of regression that reduction of variance gives more accurate output function. In the problem of two-class classification as thoroughly studied here, reduction of variance by means of averaging virtual samples *does* leads to improved classification performance in both EER and HTER criteria. Furthermore, the gain in EER criteria is more consistent (better) than the gain in HTER. In addition to this obversavtion, high cost of FA in biometric authentication favours this virtual method. As a conclusion, the virtual method is an effective way of improving a general biometric authentication system when only one sample is available.

Acknowledgement

The authors wish to thank the Swiss National Science Foundation for supporting this work through the National Centre of Competence in Research (NCCR) on "Interactive Multimodal Information Management (IM2)". Special thanks go to Christine Marcel who provided the trained MLP models.