IDIAP RESEARCH REPORT

A Study of the Effects of Score Normalisation Prior to Fusion in Biometric Authentication Tasks

Norman Poh [a] Samy Bengio [a]

IDIAP–RR 04-69

December 2004

submitted for publication

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

IDIAP Research Institute **www.idiap.ch**
Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 04-69

A Study of the Effects of Score Normalisation Prior to Fusion in Biometric Authentication Tasks

Norman Poh Samy Bengio

December 2004

submitted for publication

Abstract. Although the subject of fusion is well studied, the effects of normalisation prior to fusion are somewhat less well investigated. In this study, four normalisation techniques and six commonly used fusion classifiers were examined. Based on 24 (fusion classifiers) as a result of pairing the normalisation techniques and classifiers applied on 32 fusion data sets, $4 \times 6 \times 32 = 768$ fusion experiments were carried out on the XM2VTS score-level fusion benchmark database, it can be concluded that trainable fusion classifiers are potentially useful. It is found that some classifiers are very *sensitive* (in terms of Half Total Error Rate) to normalisation techniques such as Weighted sum with weights optimised using Fisher-ratio and Decision Template. The mean fusion operator and user-specific linear weight combination are relative less sensitive. It is also found that Support Vector Machines and Gaussian Mixture Model are the *least sensitive* to different normalisation techniques, while achieving the best generalisation performance. For these two techniques, score normalisation is unnecessary prior to fusion.

1 Introduction

Fusion of multimodal biometrics has been shown to be a promising approach to improve biometric system accuracy [10]. Often, scores are normalised before the actual fusion of scores take place. Some studies claim that score normalisation is *necessary* because scores from different systems are incomparable [16], In [1], it was even mentioned that score incomparability affects the performance of fusion if linear combination is used. Although there exists a number of comparative studies in the literature, e.g. [11] in a general classification context and [7, 18, 19] in biometric authentication, we are interested here to study the effects of score normalisation prior to fusion. The key question is: "Is score normalisation *necessary*? When should it be applied?". In [12], the issue of score normalisation prior to fusion was examined in the context of handwritten digit classification. The term score normalisation is called "confidence transformation" in [12]. Various normalisation techniques together with fixed and trainable fusion classifiers were tested. Furthermore, the parameters in the normalisation techniques and the fusion classifiers were also *joinlty* optimised. The experimental results show that the joint-optimisation strategy did not benefit from the combination. According to the same study, among fixed rule classifiers tested (e.g., mean, minimum, maximum, etc), the mean rule works best. Among trainable classifiers using linear combination, the Support Vector Machine (SVM) algorithm performs best. However, when observed closely the experimental results, the fusion due to un-normalised or normalised scores are after all, not much different (e.g. in one experiment setting, 99.79% for un-normalised score versus 99.85% for normalised case). There was unfortunately no statistical significance test supporting the experiments. In the context of biometric authentication, according to a recent study [8], normalisation techniques were addressed according to two criteria: robustness and efficiency. *Robustness* refers to insensitivity to *outliers* while *efficiency* refers to proximity of the obtained estimate to the optimal estimate when the distribution of the data is known. Several normalisation techniques were considered, and can be grouped into linear and non-linear transformations (to be detailed later).

Although robustness and efficiency are certainly important criteria, we are more interested in the *generalisation performance*, i.e., how well the combination of a chosen normalisation technique and a chosen fusion classifier performs on the unseen data. In our opinion, in the definition of robustness, there are at least two types of *outliers*. In statistical terms, outliers lie in the extremities of a distribution. Consider a single-dimension score and a binary classification problem such as biometric authentication. According to the usual statistical definitions, the outliers of each class are at the two extreme ends of the (class-dependent) score distribution. However, we are only interested in the outliers that are near the decision boundary (the "bad" outliers) and not interested in the outliers that are correctly classified but far from the decision boundary (the "good" outliers). Hence, we argue that for the purpose of classication, only the bad outliers should be given important considerations. In our opinion, the efficiency aspect requires the distribution to be known or specified in advance. In [8], normalisation techniques that enforce a parameterised non-linear mapping functions are given high efficiency, whereas linear normalisation techniques which *do not* change the original (and often *unknown*) distribution cannot be analysed in this way. In case of known Gaussian score distribution, however, the associated normalisation technique is considered to have high efficiency. Unfortunately, in [8], there was no objective study of how efficiency affects the generalisation performance. We conjecture that for good generalisation performance, there is no need to assume a distribution *prior* to fusion. This is because the actual joint distribution of scores can be modeled using parametric (e.g. Gaussian Mixture Models) or non-parametric approaches (e.g. Parzen windows) [5, Chap 2]. Furthermore, using the wrong distribution assumption may lead to degraded performance. For instance, it has been shown [4] that fusion using linear combination of expert scores may lead to poor performance when the weights are found using the Fisher discriminant analysis procedure. This is because this method assumes that the class-dependent scores are sampled from a multivariate Gaussian distribution. In biometric authentication fusion, however, class-dependent scores do not neccessarily follow a normal distribution. This observation is also supported by our experimental results here.

The empirical results in [8, Table 2] show that using the orignal scores fused using average (the mean fusion operator), the generalisation performance is *as good as* applying complicated normalisation techniques such as z-score and tanh normalisation techniques (their difference is not statistically significant), despite large number of simulated scores. Thus, this observation further raises the question whether score normalisation is necessary or not. One possible disadvantage of using complicated normalisation techniques is that the generalisation performance could even degrade if the normalisation parameters required are wrongly estimated.

In this paper, we conducted a systematic study of normalisation effects on different fusion classifiers. Prior to experimentation, we posited that score normalisation should be designed with the fusion classifier to be used. In other words, using the wrong pairing of score normalisation technique and fusion classifier may result in bad fusion performance. To verify our hypothesis, we identified four different methods to perform score normalisation that are relevant to biometric authentication, namely, no normalisation, zero-mean unit variance (the z-score), margin normalisation and probabilistic inverse normalisation (*inv*). The first two methods are linear transformations whereas the last two are non-linear transformations. Probabilistic inverse normalisation simply "reverses" the process due to sigmoid (or hyperbolic tangent), by using the inverse of sigmoid (or hyperbolic tangent) function. We emphasised here the normalisation techniques that require few parameters to tune. Normalisation techniques studied in [8] such as double-sigmoid, tanh and bi-weight are thus not considered here. Instead, we propose a new normalisation technique called the *margin score normalisation*, which does not have any free parameters to be estiamted.

We also selected a number of commonly used fusion classifiers. Here, we are mainly interested in fusion at score-level, and not at decision level. This is to ensure that valuable correlation (or higher order statistics), if exists, is not lost due to the crisp output. The fusion classifiers used are the mean operator, weighted sum (wsum) with weights found using Fisher discriminant analysis [5, Sec. 3.6] and user-specific weights [9], Gaussian Mixture Model (GMM) [5, Sec. 2.6], Support Vector Machines [17] and Decision Template (DT) [11]. The pairing of 4 normalisation techniques and 6 classifiers result in 24 fusion approaches.

This paper is organised as follows: Section 2 discusses the score-level fusion benchmark database used and the evaluation procedure. Sections 3 and 4 discuss the scores normalisation techniques and classifiers considered, respectively. Section 5 presents the experimental results. This is followed by conclusions in Section 6.

2 Database and Evaluation

The XM2VTS benchmark fusion database was used [14]. It contains a total 32 multimodal and intramodal fusion data sets. Each of these data sets contains the scores of two different experts. They can be from the same modalities but different features (intramodal) or from different modalities (muultimodal). The client-independent setting is used throughout all experiments. The scores of these fusion experiments are made publicly available[1].

The most commonly used performance visualising tool in the literature is the Decision Error Trade-off (DET) curve [13]. It has been pointed out [2] that two DET curves resulting from two systems are not comparable because such comparison does not take into account how the thresholds are selected. It was argued [2] that such threshold should be chosen *a priori* as well, based on a given criterion. This is because when a biometric system is operational, the threshold parameter has to be fixed *a priori*. As a result, the Expected Performance Curve (EPC) [2] was proposed. The criterion used (on a development set) is called the Weighted Error Rate (WER), which is a weighted sum between false acceptance and false rejection errors by a parameter known as α. The generalisation performance is measured (on an evaluation set) by Half Total Error Rate (HTER) which is a special case of WER with $\alpha = 0.5$. The EPC curve can be interpreted similarly to the DET curve, i.e., the lower the curve, the better the generalisation performance. Section 4.1 discusses how the EPC curve is calculated. In this study, the *pooled* version of EPC is used to visualise the performance. The idea is to plot a single

[1] http://www.idiap.ch/∼norman/fusion

EPC curve instead of 32 EPC curves for each of the 32 fusion experiments. This is done by calculating the false acceptance and false rejection errors for *each* of the α values. The pooled EPC curve and its implementation can be found in [14].

3 Score Normalisation Techniques

Suppose that $y_i \in \{y_1, \dots y_N\}$ is a score to be fused and there are N base-experts (or classifiers). The score normalisation techniques discussed here are applied to *each* y_i, *independent* of the output of other classifier scores. The zero-mean unit-variance (or z-score) is calculated as:

$$y_i^{zmun} = \frac{y_i - \mu_i}{\sigma_i} \tag{1}$$

where μ_i and σ_i are mean and standard deviation of y_i calculated from a *development set*. Various linear normalisation techniques have been considered in [8], such as min-max, decimal scaling and median. They are defined as:

$$y_i^{mm} = \frac{y_i - \min(y_i)}{\max(y_i) - \min(y_i)}, \tag{2}$$

$$y_i^{dec} = \frac{y_i}{10^{\log_{10} \max y_i}}, \tag{3}$$

$$y_i^{med} = \frac{y_i - \mathrm{median}(y_i)}{\mathrm{median}\left(|y_i - \mathrm{median}(y_i)|\right)}. \tag{4}$$

The probabilistic inversion is just the inverse of sigmoid or hyperbolic tangent function. These functions are only applied to base-classifiers whose output is a sigmoid or a hyperbolic tangent function (typically a Multi-Layer Perceptron, MLP). The motivation for such inversion is that correlation, which measures the *linear dependency of relationship* is often *lost* (see Fig. 1 for an illustration). The usual definition of sigmoid and tangent hyperbolic are:

$$\mathrm{sigmoid}(z) = \tfrac{1}{1+\exp(-z)} \qquad \text{and} \qquad \tanh(z) = \tfrac{\sinh(z)}{\cosh(z)} \tag{5}$$

respectively. If y_i is an output of a sigmoid or a hyperbolic tangent function, its inverse is:

$$\mathrm{sigmoid}^{-1}(y_i) = -\log\left(\tfrac{1}{y_i} - 1\right) \qquad \text{and} \qquad \tanh^{-1}(y_i) = \tfrac{1}{2}\log\left(\tfrac{1+y_i}{1-y_i}\right), \tag{6}$$

respectively. A better way of achieving the same effect is to use the score just before applying sigmoid or hyperbolic tangent function. We are considering the less fortunate case whereby we are given the output scores and are unable to obtain the original scores (this is the case for the XM2VTS benchmark score data sets).

Suppose that $y_i^k \sim Y_i^k$ is the output of expert (classifier) i given that the true class label is $k = \{C, I\}$, that is either a client or an impostor, obtained from the distribution $P(Y_i^k)$. Then, it can be shown that False Acceptance Rate (FAR) and False Rejection Rate (FRR) are:

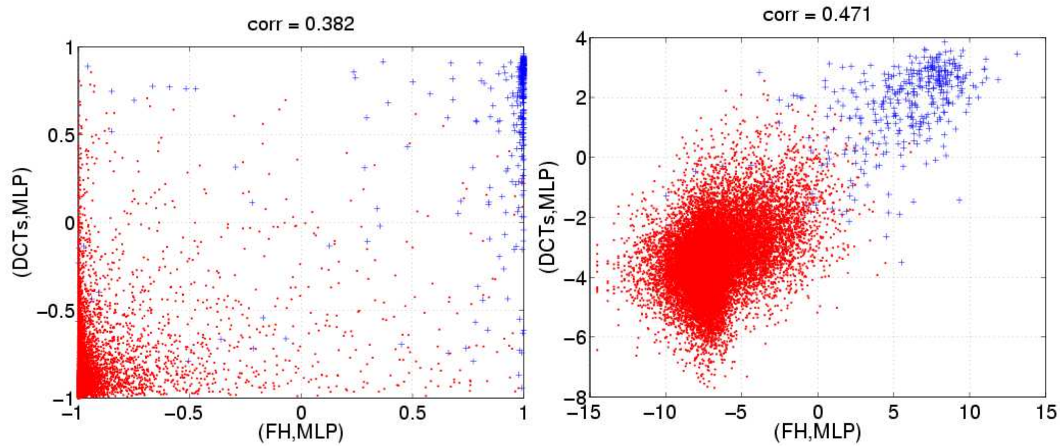
$$\mathrm{FAR}_i(\Delta) = 1 - \int_{-\infty}^{\Delta} P(Y_i^{k=I})dy \qquad \text{and} \qquad \mathrm{FRR}_i(\Delta) = \int_{\Delta}^{\infty} P(Y_i^{k=C})dym \tag{7}$$

respectively as functions of the decision threshold Δ. Replacing Δ by y_i, the margin transformed score can be defined as:

$$y_i^M = \mathrm{FRR}_i(y_i) - \mathrm{FAR}_i(y_i). \tag{8}$$

Note that the value of y_i^M is always in the range $[-1, 1]$, making it attractive to be interpreted as a confidence. In a separate study [15], the magnitude of y_i^M was used as a confidence value.

The margin score transformation is somewhat similar to the double sigmoid function, or the tanh-estimator described in [8]. However, these non-linear functions have free parameters (3 in the double



(a) original

(b) prob. inverse

Figure 1: Scatter plots of one of the 32 fusion data sets using (a) the original score prior to fusion and (b) probabilistic inversed scores. The two base-classifiers use the same modality but different feature set. The X-axis is a face expert based on historgram features and an MLP classifier, labeled as (FH,MLP). The Y-axis is also a face expert based on DCTMod2 features and an MLP classifier, labeled as (DCTs,MLP). Hence, they are expected to be somewhat correlated. Their corresponding correlations are measured to be 0.382 for (a) and 0.471 for (b). In all 32 datasets involving MLPs, the correlation among classifiers are systematically under-estimated in the original score space than in the probabilistic inverse space.

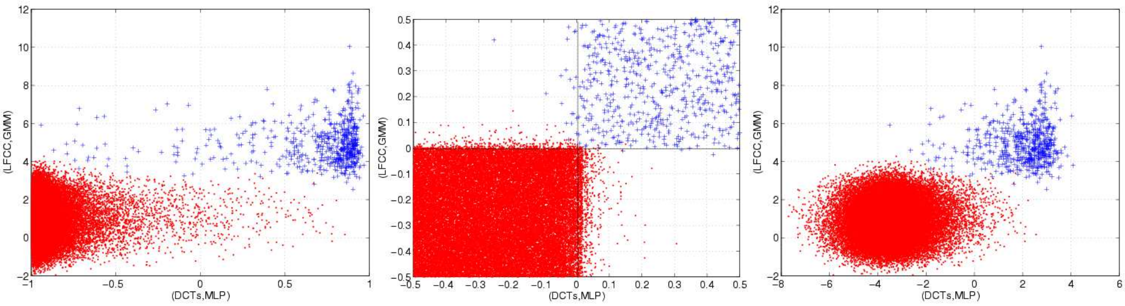
sigmoid function and 2 in the tanh-estimator). The double sigmoid function relies on heuristic to tune these free parameters whereas the tanh estimator relies on the (robust) Hampel estimates. Since the margin score transformation has the same form as these two non-linear transformations, with the additional advantage of *no free parameters*, we will use the margin score only in our experiments.

Fig. 2(a) shows the original scores before any normalisation. Note that the X-axis is the output of an MLP whereas the the Y-axis is the output of a GMM. Figure Fig. 2(b) shows the margin-transformed scores and figure Fig. 2(c) show the probabilistic inversed scores, applied only to the MLP output (the GMM output remains unchanged). The z-scores are not shown here because it is similar to Fig. 2(a), except the change of scale in the X- and Y-axes.

The margin score has a very convenient interpretation. The zero value on either axis implies the optimal *a posteriori* decision threshold (obtained on the training set). Hence, if the AND operator is used, the upper right corner of the partition ($X \geq 0, Y \geq 0$) will be classified as clients whereas the rest of the partitions as impostors. On the other hand, if the OR operator is used, the lower left partition ($X < 0, Y < 0$) will be classified as impostors whereas the rest of the partitions as clients.

4 Classifiers Investigated

Classifiers which commonly employ score normalisation can be categorised into their nature of capability. Here we consider linear and non-linear classifiers. Linear classifiers used are the mean operator, weighted sum (*wsum*) with weights found using Fisher ratio [5, Sec. 3.6], user-specific method [9] and Support Vector Machine (SVM) with a linear kernel [17]. Non-linear classifiers considered are Gaussian Mixture Model (GMM) and Decision Template (DT). Linear classifiers are considered here because in biometric authentication, the non-linearity part of the problem has mostly been solved by



(a) original (b) margin (c) prob. inverse

Figure 2: Scatter plots of one of the 32 fusion data sets using (a) the original score prior to fusion, (b) margin-transformed scores, and (c) probabilistic inverse scores. The X-axis is the output of an MLP classifier and the Y-axis is the output of a GMM classifier. A plus sign denotes a client access (upper right cluster) whereas a dot denotes an impostor access (lower left cluster).

the base-experts. Hence, it is probable that at the score-level fusion, one does not need complicated classifiers. The additional advantage is that one will have less risk in overfitting the data. Nevertheless, both linear and non-linear classifiers are investigated here. It should be mentioned that other fixed rule classifiers such as minimum, maximum and product have been carried out (not shown here) and they did not perform as well as the average/mean rule. This confirms the findings in [10] that the average rule is relatively robust as compared to other fixed rule classifiers.

4.1 A Separate Threshold Estimation Procedure

Note that the fusion classifiers to be discussed have a threshold (or bias) parameter that is tuned *after* the classifiers are built. This is done by forward-propagating the classifiers with the development and evaluation *input* scores (to be fused). In this way, one obtains the *fused* scores on both the development and evaluation sets. The threshold parameter is then tuned by minimising by a criterion called Weighted Error Rate (WER). It is defined as:

$$\mathrm{WER}_\alpha(\Delta) = \alpha \mathrm{FAR}(\Delta) + (1-\alpha)\mathrm{FRR}(\Delta), \tag{9}$$

where $\alpha \in [0, 1]$ balances between FAR and FRR. The optimal threshold is calculated by minimising WER with respect to the threshold Δ, i.e.:

$$\Delta^* = \arg\min_{\Delta} \mathrm{WER}_\alpha(\Delta). \tag{10}$$

Having chosen an optimal threshold using the WER threshold criterion discussed previously, the final performance is evaluated on the *evaluation* set. In this study, we use the Half Total Error Rate (HTER) as an evaluation criterion. It is a special case of WER with $\alpha = 0.5$. This implies that the class priors for client and impostor distributions are equal. It further assumes that the cost of false acceptance and false rejection are also equal). The curve that plots HTER versus α is called the Expected Performance Curve (EPC) [2]. It is constructed as follows: for various values of α in Eqn. (9) between 0 and 1, select the optimal threshold Δ on a development (training) set using Eqn. (10), and compute the HTER on the evaluation (test) set with this threshold.

The reason for a separate tuning procedure of the threshold (or bias) parameter is that the fusion classifiers often are not trained using WER. It is also unclear whether training a fusion classifier with a specific WER (of a given α) is beneficial or not. However, it is certainly easier to adjust the threshold after the fusion classifier is built. The added advantage is that the fusion classifier does not have to be retrained; instead, the threshold is re-adjusted on the development set using Eqn. (10) and the new HTER value at a given α can then be re-evaluated. In this way, the threshold (bias) parameter need not be considered when building the fusion classifier.

4.2 Linear Classifiers

The linear classifiers have the following form:

$$y_{COM} = \sum_{i=1}^{N} \alpha_i y_i - \Delta = \boldsymbol{\alpha}^T \boldsymbol{y} - \Delta \tag{11}$$

where, for convenience, we introduced the vector representation $\boldsymbol{\alpha} = [\alpha_i, \ldots, \alpha_N]^T$ and $\boldsymbol{y} = [y_i, \ldots, y_N]^T$. We write $\boldsymbol{y}^k$ when $\boldsymbol{y}$ is known to belong to class $k = \{C, I\}$. Let $\boldsymbol{\mu}^k$ and $\boldsymbol{\Sigma}^k$ be the mean and covariance matrix of $\boldsymbol{y}^k$, respectively. The within-class covariance matrix is defined as:

$$S_w = \sum_{k=\{C,I\}} \boldsymbol{\Sigma}^k$$

The Fisher linear discriminant solution of $\boldsymbol{\alpha}$ for a two-class problem (see [5, pg. 110]) is:

$$\boldsymbol{\alpha} = S_w^{-1} \left(\boldsymbol{\mu}^C - \boldsymbol{\mu}^I \right) \tag{12}$$

Note that α_i can take on any values and their sum is not necessary equal to 1.

The SVM with linear kernel has a *dual form* of weight. Suppose that $\boldsymbol{y}^{(j)}$ and $t^{(j)}$ are the input and target output of example j and $\omega^{(j)}$ is its associated *embedding strength* obtained after SVM training. Large $\omega^{(j)}$ implies that the associated example is difficult to classify, and vice-versa for small $\omega^{(j)}$. Examples with $\omega^{(j)} > 0$ are known as support vectors. The linear discriminative function of SVM can be written as (the usual sign is omitted to obtain the margin value; and the bias is omitted for the reason explained in Section 4.1):

$$f(\boldsymbol{y}) = \sum_j \omega^{(j)} t^{(j)} \left\langle \boldsymbol{y}^{(j)}, \boldsymbol{y} \right\rangle = \left(\sum_j \omega^{(j)} t^{(j)} \boldsymbol{y}^{(j)T} \right) \boldsymbol{y} = \boldsymbol{\alpha}^T \boldsymbol{y}, \tag{13}$$

where $\langle \cdot, \cdot \rangle$ is the linear kernel. As shown here, $\sum_j \omega^{(j)} t^{(j)} \boldsymbol{y}^{(j)}$ is the dual form of weight in SVM (see [6, Chap 2]).

In addition, we used client-specific weights as described in [8]. The idea is to search the weight space exhaustively on a per client basis so that the Equal Error Rate (EER) (WER with $\alpha = 0.5$) is minimised. This is done with the constraint that the weights sum to one. Whenever there are more than one location with EER= 0, we choose the weights closest to the equal weights, as described in [8].

4.3 Effects of Linear Score Normalisation on Linear Classifiers

Eqns. (1–4) all are linear transformations and have the following forms:

$$y_i^{lin} = A_i (y_i - B_i), \tag{14}$$

where A_i is a scaling factor and B_i is a bias. Suppose that a linear classifier is used. Then, the fused score can be written as:

$$\begin{aligned} y_{COM} &= \sum_i \alpha_i y_i^{lin} - \Delta = \sum_i \alpha_i A_i (y_i - B_i) - \Delta \\ &= \sum_i \underbrace{\alpha_i A i}\, y_i - \underbrace{\sum_i \alpha_i A_i B_i - \Delta} \end{aligned} \tag{15}$$

Comparing Eqn. (15) with the linear combination without normalisation, as in Eqn. (11), we see that the first underbraced term is the new weight whereas the second underbraced term is the new decision threshold. This shows that if $y_i|\forall_i$ are unevently scaled, their scaling factor A_i may not be necessary as it will be automatically absorbed by the weight. This implies that if scores are not evently scaled, the weights in the linear combination should be allowed to take on any values, without the constraint $\sum_i \alpha_i = 1$. This further implies that linear score normalisation may not be *necessary*.

4.4 Non-Linear Classifiers

We used a Bayesian classifier with density estimated using a GMM. The prior probability of each class (client or impostor) is set to be equal. This reduces the Bayesian classifier to be two competing GMMs, one for each class. The Gaussian parameters as well as the mixture parameters are calculated using the Expectation-Maximisation (EM) algorithm. We used GMM with a diagonal covariance matrix. This is a standard algorithm as discussed in [5, Chap 2].

The DT approach has been well studied in [11]. Using the same notation as before, let $\boldsymbol{\mu}^k$ be the mean vector prototype for class k. During testing, the base-classifier outputs, $\boldsymbol{y}$ is compared to each of the prototype $\boldsymbol{y}^k$. The output is then calculated as:

$$y_{COM} = -\left(\|\boldsymbol{y} - \boldsymbol{\mu}^{k=C}\| - \|\boldsymbol{y} - \boldsymbol{\mu}^{k=I}\|\right) \tag{16}$$

where $\|\cdot\|$ denotes the Euclidean distance. A negative sign is introduced here so that the measure is interpreted as similarity (the larger it is, the closer $\boldsymbol{y}$ is to the client prototype). Due to possibly different (dis)similarity measures other than Euclidean distance, DT is generally considered a non-linear classifier. In the Euclidean distance case, it is a linear classifier. In our opinion, DT is in fact a specific case of GMM with a single Gaussian component and an identity covariance matrix. The only difference is that the equivalent version of GMM uses likelihood whereas DT uses a similarity measure. This difference is somewhat negligible because likelihood is proportional to similarity measure.

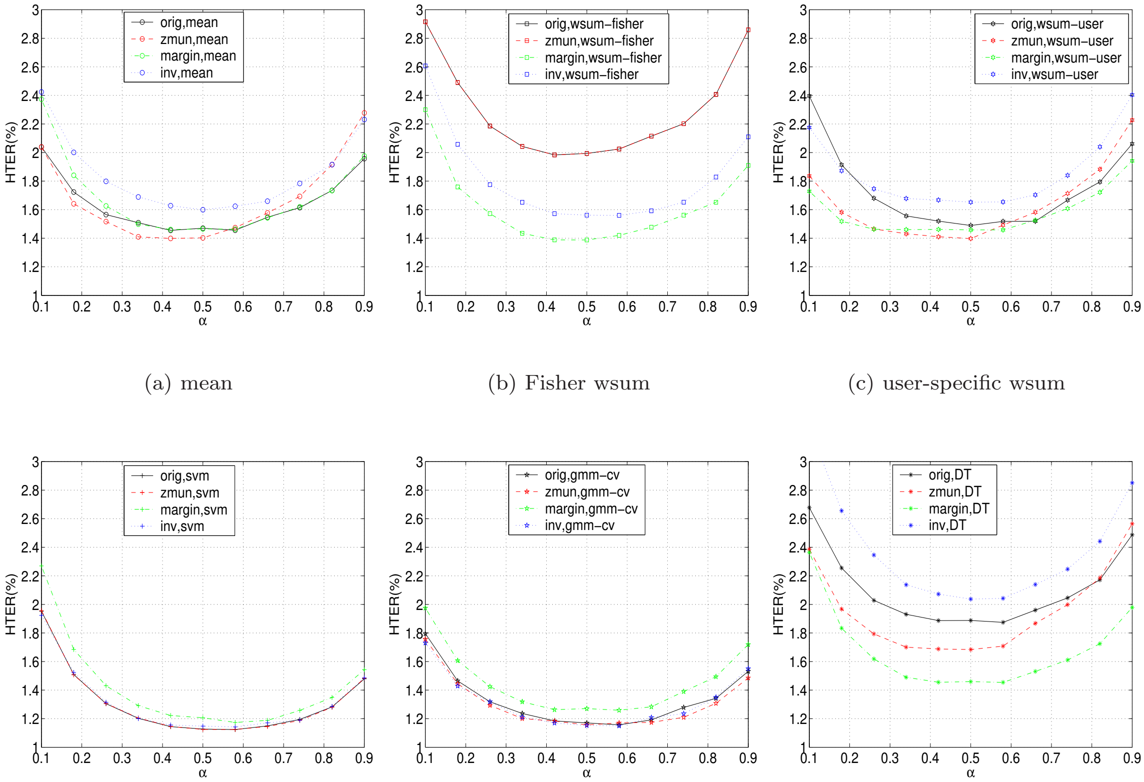
5 Experiments

Figs. 3(a)–(f) show the effects of four normalisation techniques on each of the six classifiers under study. Each curve in Figs. 3 is a pooled EPC curve over 32 fusion experiments. The lower the curve, the better the performance. Rather than subjectively observing the pooled EPC curves for "how sensitive a given classifier performs across normalisation techniques and different α values", we propose an objective measure called the *HTER sensitivity measure*. Suppose the (pooled) EPC curve of experiment p is defined by $\text{HTER}_{\alpha,p}$, where $\alpha \in [0, 1]$. The EPC curve of experiment p is plotted as $\text{HTER}_{\alpha,p}$ versus α. One possible definition of sensitivity is:

$$\text{sensitivity}_\alpha = \log|\min_p(\text{HTER}_{\alpha,p}) - \max_p(\text{HTER}_{\alpha,p})| \tag{17}$$

The minimum and maximum operators define the envelops of the EPC curves across different p normalisation techniques, for a given α value. $|\cdot|$ takes the absolute value of its variable. The log function is used here so that the absolute difference of the upper EPC and lower EPC curves are projected onto the logarithmic scale. We plot the distribution of HTER sensitivity (across different α values) using a boxplot, as shown in Fig. 4. As can be observed, weighted sum with Fisher discriminant analysis and DT are highly sensitive to different normalisation techniques while SVM and GMM are less sensitive to different normalisation techniques. The mean and user-specific weights are moderately insensitive to normalisation techniques.

In terms of generalisation performance (see Fig. 3), SVM and GMM have the best overall performance, across all 32 fusion datasets. SVM is slightly better than GMM although their difference is insignificant according to the HTER significance test [3] at 90% of confidence. Although the user-specific weight fusion approach uses the prior knowledge of client-specific information (as opposed to



(a) mean (b) Fisher wsum (c) user-specific wsum

(d) SVM (e) GMM (f) DT

Figure 3: Pooled EPC curves, each derived from 32 fusion data sets, as a result of applying the four normalisation techniques under study, prior to fusion using (a) the mean operator, (b) wsum with weights found using Fisher discriminant analysis, (c) weighted sum with weights found using user-specific method (d) SVM, (e) GMM and (f) DT.

the rest of the fusion classifiers under study), it did not perform well in this score fusion database because there are only 3 (resp. 2) genuine scores and 200 (resp. 200) impostor scores on a *per client basis* for Lausanne Protocol I (resp. II) to train the fusion classifier. In [8], this approach works better than the mean fusion classifier because as many as 6 genuine scores were available for training. Thus, when lacking the amount of client-specific information, its generalisation performance is only as good as the mean operator. This is partly due to the constraint that when there are several weights where its EER is evaluated to be zero (in the development set), one chooses the weight vector that is neareast to equal weights. Taking the viewpoint that DT is a specific case of GMM with one class-dependent component and identity covariance matrix, DT can only perform *as good as* a GMM and *not* better. Note that in these fusion experiments, SVM with a linear kernel and GMM achieve *best* generalisation performance while at the same time are *the most insensitive* to different normalisation techniques.

6 Conclusions

Four normalisation techniques and six commonly used fusion classifiers were examined in this study. In addition, the well studied Decision Template (DT), The normalisation techniques considered are no normalisation, zero-mean unit-variance normalisation (z-score), probabilistic inverse (inversion of

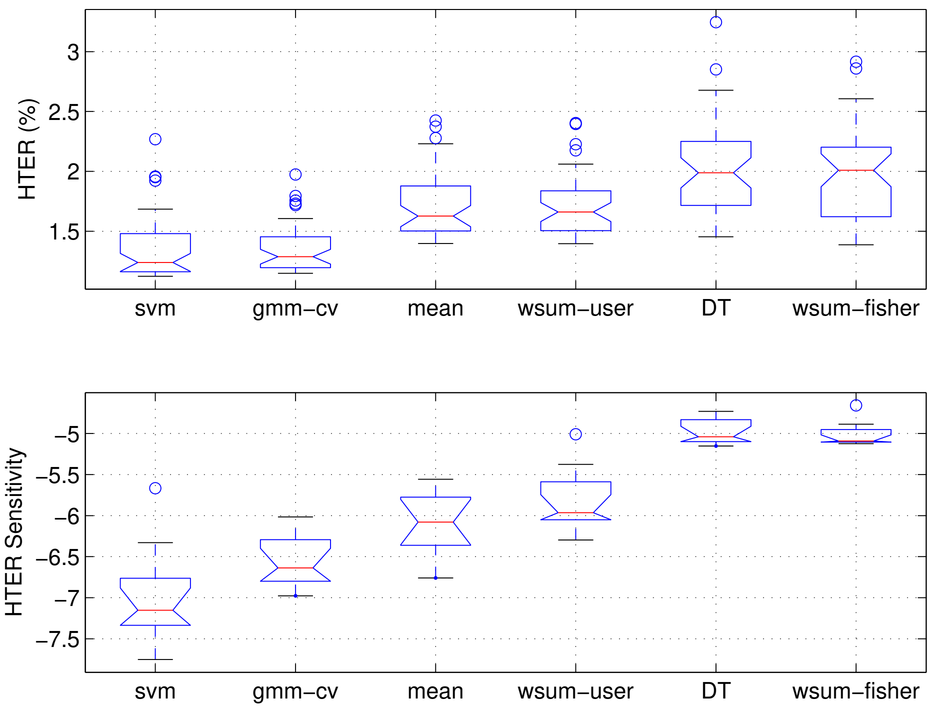


Figure 4: Top: HTER distribution (the Y-axis) across all four normalisation techniques of each fusion classifier (the X-axis). Classifiers are sorted in decreasing generalisation performance (increasing HTER). Bottom: HTER sensitivity (the X-axis) of different classifiers (in the Y-axis) across all four normalisation techniques under study. A bar within each box denotes the median value. The upper and lower edges of a box denotes the upper and lower quantiles of the distribution. Horizontal lines denotes the extents of the data, with ◦'s representing outliers.

sigmoid or hyperbolic tangent function according to the output, usually from a Multi-Layer Perceptron, MLP), and margin normalisation (a one-to-one mapping from a score to a confidence scale in the range $[-1, 1]$ calculated from a development set). The six classifiers investigated are the mean rule, weighted sum optimised using Fisher discriminant, weighted sum optimised using user-specific score data, Support Vector Machines (SVMs), Gaussian Mixture Models (GMMs) and Decision Templates (DTs).

Finally, the overall best generalisation performance is achieved by SVM, which is competitively (but insignificantly) followed by GMM. Hence, based on $24 \times 32 = 768$ fusion experiments carried out on the XM2VTS fusion benchmark database, trainable fusion is potentially useful. We verified that the type of classifiers to be used must be designed carefully together with the normalisation procedure prior to fusion for best generalisation performance. For SVM and GMM, due to the robustness against different normalisation techniques, experimental evidence suggests that scores need not be normalised.

In this study, we proposed the margin score normalisation. It is similar to the double sigmoid function or the tanh-estimator proposed in [8], but has the additional advantage of no free parameters to tune. We further proposed a HTER *sensitivity measure*, in order to evaluate objectively how sensitive a fusion classifier is due to different normalisation techniques.

Based on our experimental evidences, score normalisation prior to fusion is *only* necessary for classifiers that are *sensitive* to normalisation techniques. In particular, weighted sum fusion with weights optimised using the Fisher-ratio, and Decision Template are particularly sensitive to the underlying joint-distribution of scores. Margin score transformation which transforms scores into a bounded confidence space improves the generalisation performance. Similarly, the generalisation performance of user-specific weighted sum also improves using margin scores. This somewhat confirms the findings in [8]. Finally, SVM and GMM are found to be robust across different normalisation techniques while achieving the best generalisation performance. This suggests that for these two fusion classifiers, score normalisation prior to fusion *is unnecessary*.

Acknowledgement

This work was supported in part by the IST Program of the European Community, under the PASCAL Network of Excellence, IST-2002-506778, funded in part by the Swiss Federal Office for Education and Science (OFES) and the Swiss NSF through the NCCR on IM2. This publication only reflects the authors' view.

References

[1] H. Altinay and M. Demirekler. Why Does Output Normalization Create Problems in Multiple Classifier Systems? In *Proc. 16-th Int. Conf. on Pattern Recognition (ICPR)*, pages 20775–20778, Quebec, 2002.

[2] S. Bengio and J. Mariéthoz. The Expected Performance Curve: a New Assessment Measure for Person Authentication. In *The Speaker and Language Recognition Workshop (Odyssey)*, pages 279–284, Toledo, 2004.

[3] S. Bengio and J. Mariéthoz. A Statistical Significance Test for Person Authentication. In *The Speaker and Language Recognition Workshop (Odyssey)*, pages 237–244, Toledo, 2004.

[4] Samy Bengio, Johnny Mariethoz, and Sebastien Marcel. Evaluation of Biometric Technology on XM2VTS. IDIAP-RR 21, IDIAP, 2001.

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

[6] N. Cristianini and J. Shawe-Taylor. *Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, 2000.

[7] J. Fierrez-Aguilar, J. Ortega-Garcia, D. Garcia-Romero, and J. Gonzalez-Rodriguez. A Comparative Evaluation of Fusion Strategies for Multimodal Biometric Verification. In *Springer LNCS-2688, 4th Int'l. Conf. Audio- and Video-Based Biometric Person Authentication (AVBPA 2003)*, pages 830–837, Guildford, 2003.

[8] A. Jain, K. Nandakumar, and A. Ross. Score Normalisation in Multimodal Biometric Systems. *Pattern Recognition (to appear)*, 2005.

[9] A. Jain and A. Ross. Learning User-Specific Parameters in Multibiometric System. In *Proc. Int'l Conf. of Image Processing (ICIP 2002)*, pages 57–70, New York, 2002.

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

[11] L. Kuncheva., J.C. Bezdek, and R.P.W. Duin. Decision Template for Multiple Classifer Fusion: An Experimental Comparison. *Pattern Recognition Letters*, 34:228–237, 2001.

[12] C-L. Liu. Classifier Combination Based on Confidence Information. *Pattern Recognition*, (38):11–28, 2004.

[13] A. Martin, G. Doddington, T. Kamm, M. Ordowsk, and M. Przybocki. The DET Curve in Assessment of Detection Task Performance. In *Proc. Eurospeech'97*, pages 1895–1898, Rhodes, 1997.

[14] N. Poh and S. Bengio. Database, Protocol and Tools for Evaluating Score-Level Fusion Algorithms in Biometric Authentication. Research Report 04-44, IDIAP, Martigny, Switzerland, 2004.

[15] N. Poh and S. Bengio. Improving Fusion with Margin-Derived Confidence in Biometric Authentication Tasks. Research Report 04-63, IDIAP, Martigny, Switzerland, 2004.

[16] S.N. Srihari T.K. Ho, J.J. Hull. Decision Combination in Multiple Classifier Systems. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 16(1):66–75, January 1994.

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

[18] Patrick Verlinde, Gerard Chollet, and Marc Acheroy. Multimodal Identity Verification Using Expert Fusion. *Information Fusion*, 1(1):17–33, 2000.

[19] Y. Wang, Yh. Wang, and Tn. Tan. Combining Fingerprint and Voiceprint Biometric for Identity Verification: an Experimental Comparison. In *Springer LNCS-3072, Int'l Conf. on Biometric Authentication (ICBA)*, pages 663–670, Hong Kong, 2004.