IMPROVING FACE AUTHENTICATION USING VIRTUAL SAMPLES

Norman Poh Hoon Thian, Sébastien Marcel and Samy Bengio

IDIAP, CP 592, 1920 Martigny, Switzerland

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

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.

The authors want 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)".

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:

$$E[y] = E[f(\mathbf{x})] + E[\eta] \quad (3)$$
$$= f(\mathbf{x}). \quad (4)$$

Moreover, the variance of y can be written as:

$$\text{Var}[y] = \frac{1}{N}\text{Var}[\eta] \quad (5)$$

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 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.

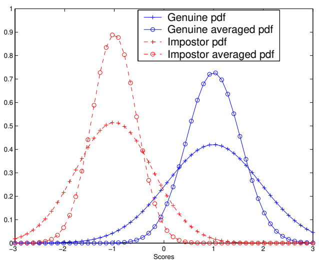


Fig. 1. Averaging scores distribution in a two-class problem

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:

$$\text{err}_c = \frac{1}{N^2}\sum_{i=1}^{N}\text{err}_i \quad (6)$$
$$= \frac{1}{N}\text{mean}(\text{err}_i). \quad (7)$$

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:

$$\text{err}_c = \frac{1}{\alpha}\text{mean}(\text{err}) \quad (8)$$
$$1 \le \alpha \le N.$$

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 ($\forall_{i,j}(\text{err}_i = \text{err}_j)$), then $\text{mean}(\text{err}) = \text{err}_i = \text{err}_c$, which implies that $\alpha = 1$.

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,

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

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*. 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

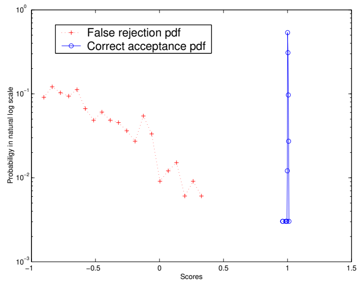
Data sets	Models	FA[%]	FR[%]	HTER[%]
LP1 Test	Original	1.230	2.750	1.990
LP1 Test	Averaged	1.474	1.750	1.612
LP2 Test	Original	1.469	2.250	1.860
LP2 Test	Averaged	1.285	1.750	1.518

3.5. Analysis of the results

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 2. When given

(a) False rejection

(b) Correct acceptance



(c) Corresponding histograms

Fig. 2. Examples of “bad” and “good” photos and their corresponding distribution of virtual scores for client 006

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.

4. CONCLUSION

By applying N various 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 between 1 and N, by taking into account the assumption that these N image samples are dependent on the original image. As a consequence, the classification error, with respect to the original method is reduced by a factor between 1 and N as well.

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

$$y = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} f(h(g(\mathbf{x}, t))) \quad (9)$$

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 9 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.

5. 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.