5

Biometrical Fusion – Input Statistical Distribution

Luis Puente, María Jesús Poza, Belén Ruíz and Diego Carrero
Universidad Carlos III de Madrid
Spain

1. Introduction

Biometric systems are based on the use of certain distinctive human traits, be they behavioral, physicial, biological, physiological, psychological or any combination of them. As reflected in the literature, some of the most frequently used biometric modalities include fingerprint, face, hand geometry, iris, retina, signature, palm print, voice, ear, hand vein, body odor and DNA. While these traits may be used in an isolated manner by biometric recognition systems, experience has shown that results from biometric systems analyzing a single trait are often insufficiently reliable, precise and stable to meet specific performance demands (Ross et al. 2006). In order to move system performance closer to the level expected by the general public, therefore, novel biometric recognition systems have been designed to take advantaje from taking multiple traits into account.

Biometric fusion represents an attempt to take fuller advantage of the varied and diverse data obtainable from individuals. Just as is the case with human recognition activities in which decisions based on the opinions of multiple observers are superior to those made by only one, automatic recognition may also be expected to improve in both precision and accuracy when final decisions are made according to data obtained from multiple sources.

its discussion of data fusion in biometric systems, the present chapter will analyze distinct types of fusion, as well as particular aspects related to the normalization process directly preceding data fusion.

2. Biometrics

Biometric recognition involves the determination of the identity of an individual according to his/her **personal qualities** in opposition to the classical identification systems which depend on the users' **knowledge** of a particular type of information (e.g., passwords) or **possession** of a particular type of object (e.g., ID cards).

In biometrics, 'recognition' may be used to refer to two distinct tasks. In the first one, that is called verification, an individual claims to be certain user who has been previously registered(enrolled) into the system. It is also possible that the individual does not indicate his/her identity but there exist some additional information that allow to suppose it. In this case system operation is reduced to confirm or reject that a biometric sample belongs to the claimed identity. In the second task, identification, it is not available such prior information about individual's identity, the system must determine which among all of the enrollees the

subject is, if any. In the present chapter, only verification will be discussed insofar as identification may be understood as the verification of multiple entities.

In both cases (verification and identification), a **sample** of a predetermined biometric **trait** (e.g., face, voice or fingerprint) is **captured** (e.g., via photo, recording or impression) from a subject under scrutiny (**donor**), this is done using an adequate **sensor** for the task (e.g., camera, microphone or reader/scanner). A sample is called **genuine** when its donor identity and the identity of the **claimed user** are the same and it is called **impostor** when they are not. Following its capture, the sample is processed (**feature extraction**) in order to obtain the values of certain predefined aspects of the sample. This set of values constitute the **feature vector**. The feature vector is then **matched** against the **biometric model** corresponding to the individual whose identity has being claimed[1]. This model has been created at the time of that user enrols into the system. As a result of the matching, an evaluation of the degree of similarity between the biometric sample and the model is obtained, it is called **score**[2]. With this information, the **decision is taken** either to **accept** the subject as a genuine user or to **reject** the subject as an impostor (see Fig. 1).
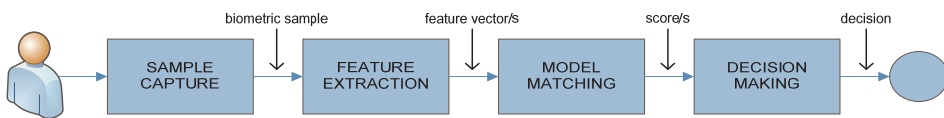

Fig. 1. Monobiometric[3] process.

In the majority of biometric recognition systems currently in use only a single biometric trait is captured in order to confirm or reject the claimed users's identity. Such systems are known as monobiometric. Nevertheless, at they heart is a pattern recognizer, which arrives at a final decision with the results obtained from a single sample processed according to a single algorithm.

Fig. 1 presents a simple representation of the biometric recognition process in monobiometric systems. After a subject presents the biometric trait which the system's sensor is designed to process, in the first stage (i.e., capturing sample), a biometric sample is obtained by the sensor and processed by the system to eliminate noise, emphasize certain features of interest and, in general, prepare the sample for the following stage of the process. In the next step (i.e., feature extraction), characteristic parameters of the sample are quantified and a feature vector that groups them is obtained. Following quantification, the system proceeds to match the feature vector (i.e., model matching) against others captured during the training phase that correspond to the individual whose identity is being claimed.

---

[1]The common structure of all biometric recognition systems is performed in two phases: (1) an initial training phase in which one or various biometric models are generated for each subject, and a later one called recognition phase, in which biometric samples are captured and matched against the models.

[2]The present chapter interprets scores as representing similarity. While, in practice, scores may also indicate difference, no generality is lost here by interpreting scores in this way since a linear transformation of the type ($s' = K\text{-}s$) can always be established.

[3]Term derived from the Greek monos (one) + bios (life) + metron (measure) and preferred by the authors of the present chapter over the term "unibiometric", also found in the literature but involving a mix of Greek and Latin morphological units. The same comment should be made about polybiometric and multibiometric terms.

These latter vectors are often represented in biometric systems with models that summarizes their variability. As a result of the matching process, a score is obtained quantifying the similarity between the sample and the model. In the final stage (i.e., decision making) and as a result of the score generated, the biometric system makes a decision to accept the sample genuine or to reject it as impostor.

3. Biometric fusion

In polybiometric systems, various data sources are used and combined in order to arrive at the final decision about the donor's identity. These systems are composed of a set of monobiometric parallel subprocesses that operate the data obtained from the distinct sources in order to finally combine them (i.e., fusing data). This fused data is then processed by system through a single subprocess until th final decision can be made regarding the truth of the claimed identity

In the construction of polybiometric recognitions systems, certain parameters must be set in response to the following questions:

- What are the distinct sources of biometric data being analyzed?
- At what point in the biometric process will the data be fused or, said another way, what intermediate data will be used for this fusion?
- What algorithm is most adequate for bringing about a particular type of fusion?

The following sections of the present chapter will analyze these different aspects of system design.

4. Data sources

In order to respond to the first question of the previous paragraph regarding multiple data sources, the following specific questions must also be considered (Ross 2007).

- How many sensors are to be utilized? In the construction of multi-sensor systems, different sensors (each with distinct performances) are used in order to capture multiple samples of a single biometric trait. In one example of this sort of polybiometric system, simultaneous photographs are captured of a subject's face using both infrared and visible light cameras.
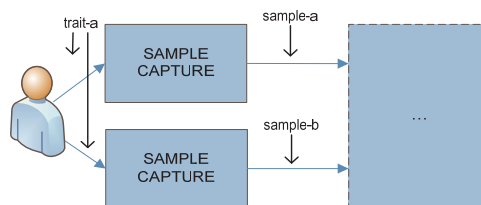

Fig. 2. Multi-sensor system.

- How many instances of the same biometric trait are to be captured? Human beings can present multiple versions of particular biometric traits (e.g., fingerprints for different fingers, hand geometry and veins for each hand and irises for each eye). As a result and with a schema similar to that of multi-sensor systems, multi-instance systems are designed to capture various instances of the same biometric trait.

- How many times is an instance of a particular trait to be captured? Using a single sensor and a single instance of a particular trait, it is nevertheless possible to obtain distinct samples of that instance under different conditions (e.g., video images taken of a trait instance from different angles or voice recordings taken at different moments and with different speech content). These multi-sample systems may also be represented by a schema similar to that of multi-sensor systems.

- How many different biometric traits are to be captured? Biometric recognition systems may be designed to analyze a single biometric trait (i.e., unimodal systems) or various traits (i.e., multimodal systems). The particularities of the latter type of system are represented by the schema below.
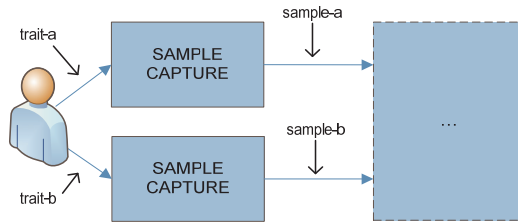

Fig. 3. Multimodal systems.

- How many distinct feature extraction algorithms are to be utilized in the processing of the biometric samples? Multi-algorithm systems are designed to use various algorithms for the feature extraction from biometric samples. In this case, the use of different extraction algorithms may allow the system to emphasize different biometric features of interest (e.g., spectral or prosodic features of a voice sample) and produce different feature vectors for each.
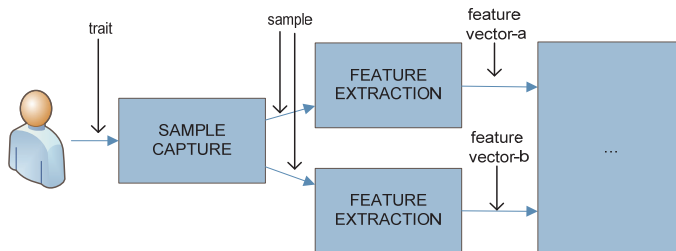

Fig. 4. Multi-algorithm systems.

- Against how many types of patterns and using how many methods are the feature vectors to be matched? Multi-matching systems are biometric recognition systems that allow match the feature vectors against various types of models or/and. using multiple techniques.

- Finally, it is also possible to construct hybrid systems systems of an even greater complexity that incorporate more than one type of the multiple data source discussed above.
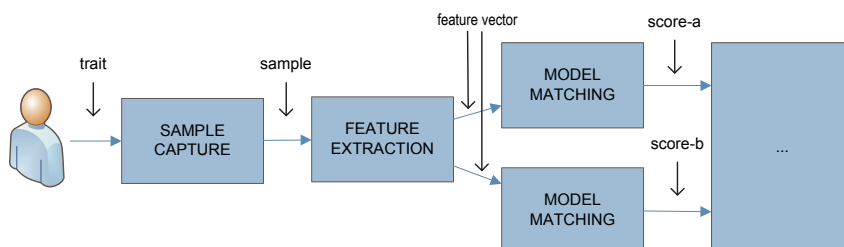
Fig. 5. Multi-matching schema.

5. Fusion level

As discussed earlier, biometric fusion is composed by a set of monobiometric subprocesses that work in parallel to operate the data obtained from distinct sources. Once this different data has been operated and fused, it is then handled by the system through a single subprocess until the point where the donor's identity final decision can be made. This process is represented in Fig. 6 below.
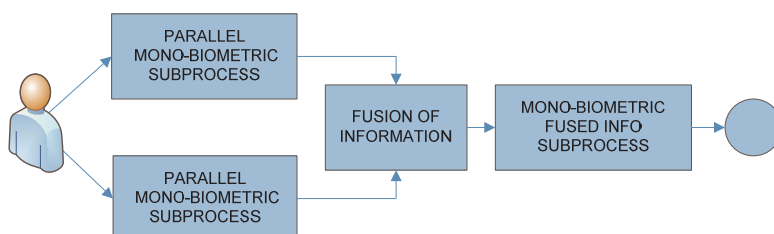

Fig. 6. Biometric fusion process.

Having considered the biometric fusion schema, it is time to return to the questions articulated earlier in the chapter and analyze now at what level of the process the fusion should be carried out or, in other words, what type of data the system should fuse. The possible responses in the literature to these points allow to establish diverse characterizations of data fusion systems defined as fusion levels(Ross 2007) (Joshi et al. 2009) (Kumar et al. 2010).

The first point at which data fusion may be carried out is at the sample level, that means immediately following sample capture by system sensors. This type of fusion is possible in multi-sensor, multi-instance and multi-sample systems and may be obtained by following a particular sample fusion method. The form that this method takes in each case depends on the type of biometric trait being utilized. While fusion may range from a simple concatenation of the digitalized sample data sequence to more complex operations between multiple sequences, but it is almost always carried out for the same reason: to eliminate as many negative effects as possible associated with the noise encrusted in the data samples during capture. Once the fused sample has been generated, it may be used by the system for feature extraction.

The second point at which data may be combined is immediately following the feature extraction. At the feature level, vectors derived from the different sources are combined, yielding a single, fused vector.
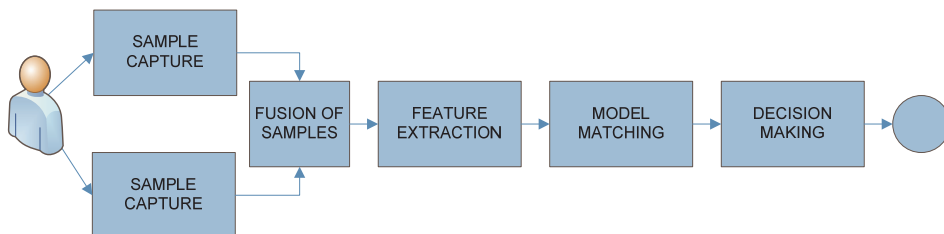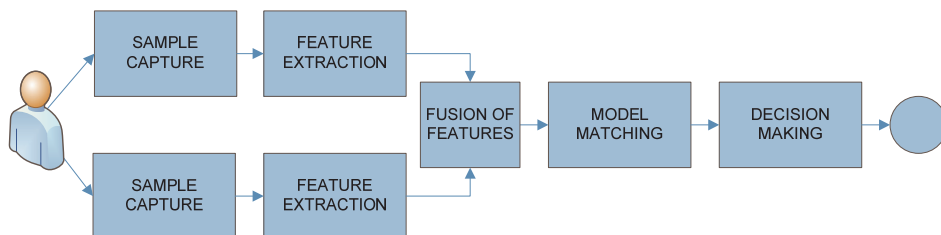
Fig. 7. Sample level fusion.

Fig. 8. Feature level fusion.

Another alternative is the fusion of scores obtained following the matching of different sample data against corresponding models. The new score resulting from this fusion is then used by the system to reach the final decision. This sort of fusion is normally carried out according to mathematic classification algorithms ranging in type from decision trees to techniques from the field of artificial intelligence, the latter of which offering learning capabilities and requiring prior training. The present chapter focuses particularly on this latter type of fusion which will be developed in much greater detail in sections below.
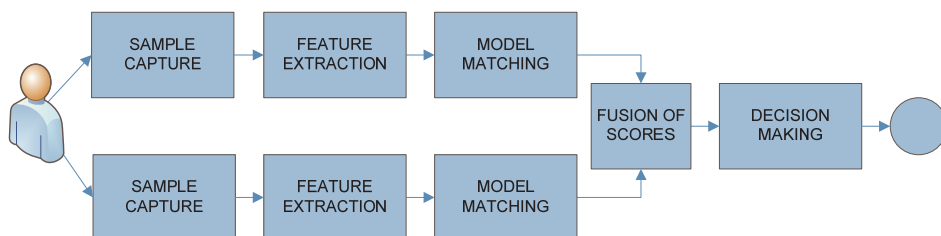

Fig. 9. Score level fusion.

Fusion may also be carried out on the final decisions obtained for each monobiometric process through the use of some kind of Boolean function. The most frequent algorithms used in this type of fusion are AND, OR and VOTING. With the first type, the final, combined decision is *GENUINE* if and only if each monobiometric process decision is also *GENUINE*. For the second type, the final, combined decision is *IMPOSTOR* if and only if each monobiometric process decision is also *IMPOSTOR*. Finally, for the third type combined decision is that of the majority of monobiometric process decisions which may or may not have been previously weighted.
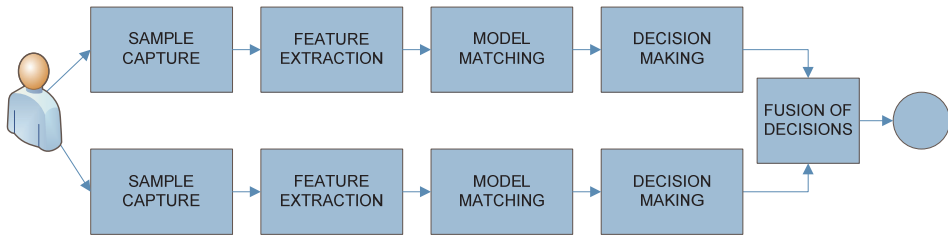
Fig. 10. Decision level fusion.

Finally, dynamic classifier selection schema uses scores generated at the data matching level in order to determine what classifier offers the highest degree of confidence. The system then arrives at a final decision through the application of solely the selected classifier. This is represented in Fig. 11 below.
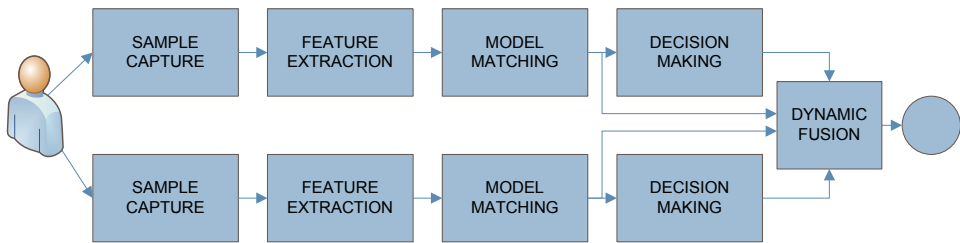
Fig. 11. Dynamic classifier selection.

6. Biometric performances

 For the recognition of a individual by a classical recognition system, the data collected (e.g., passwords or ID cards information) from the subject must be identical to the previously recorded data into the system. In biometric recognition systems, however, almost never the data captured from a subject (nor the feature vectors obtained from them) are identical to the previous ones (Ross et al. 2006). The reasons for these variations are manifold and include the following:

- Imperfections in the capture process that create alterations (e.g., noise) in the data;
- Physical changes in the capture environment (e.g., changes in lighting and degradation of the sensors used); and
- Inevitable changes over time in individual's biometric traits.
- As a result of the unrepeatibility of biometric samples, the process of biometric recognition can not be deterministic and it must be based on the stochastic behaviour of samples. In this way, rather than flatly asserting correspondence between a biometric sample and the corresponding model, biometric systems only permit the assertion that this correspondence has a certain probability of being true.

The differences observed among the distinct biometric samples taken of a single trait from a single donor are known as intra-class variations. On the other hand, inter-class variation refers to the difference existing between the samples captured by the system from one subject and those of others. The level of system confidence in the correctness of its final decision is determined according to these two types of variation. The lesser the intra-class

variation and the greater the inter-class variation are, the greater the probability that the final decision is correct.

In the matching model step, the system assigns a score to the sample feature vector reflecting the system's level of confidence in the correspondence between sample and claimed identity. If this score (*s*) lies above a certain threshold (*th*) (i.e., if: s ≥ th), the system will decide that the sample is genuine. However, if the score lies below the threshold the system will decide that the sample is an impostor one.

Insofar as score, as understood here, is a random variable, the probability that any particular score corresponds to a genuine sample can be defined by its probability density function (pdf) $f_g(s)$. Similarly, the probability that the score corresponds to an impostor sample can be defined by a pdf $f_i(s)$. As a result, the terms 'false match rate' (FMR) or 'false acceptance rate' (FAR) may be defined as the probability that an impostor sample be taken by the biometric system as genuine. Similarly, the terms 'false not match rate' (FNMR) or 'false rejection rate' (FRR) may be defined as the probability that a genuine sample be taken for as an impostor one.

When the decision score threshold is established in a system (see Fig. 12), the level of system performance is therefore established, because FAR and FRR directly depend on its value. Wether threshold values increases, FAR will also increase while FRR will decrease [Stan et al. 2009]. The optimal value of *th* can be obtained by minimizing the cost function established for the concrete system use. This cost function defines the balance between the damage that can be done by a false acceptance (e.g., a subject is granted access by the system to a protected space in which he or she was not authorized to enter) and that done by a false rejection (e.g., a subject with authorization to enter a space is nevertheless denied entry by the system).
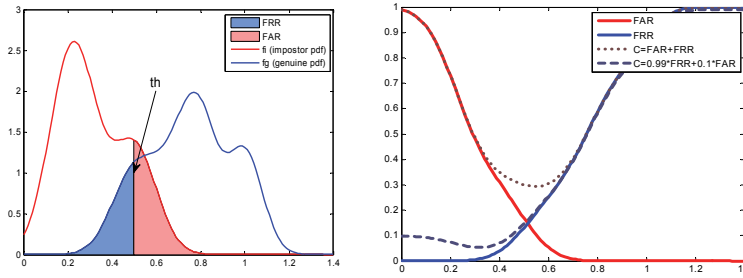
Fig. 12. Error rates and pdfs (left). Error rates and cost functions (right).

$$FRR(s) = \int_{-\infty}^{th} f_g(s)\,ds, \quad FAR(s) = \int_{th}^{\infty} f_i(s)\,ds \tag{1}$$

The National Institute of Standards and Technology (NIST) proposes as a cost function the one shown in formula 2, which is a weighted sum of both error rates. C_{FR} and C_{FA} correspond to the estimated costs of a false rejection and false acceptance, respectively, and P_g and P_i indicate the probabilities that a sample is genuine or impostor. Is obviously true that *Pi+Pg=1* (Przybocki et al. 2006):

$$C = C_{FR} \cdot FRR \cdot P_g + C_{FA} \cdot FAR \cdot P_i \tag{2}$$

In NIST recognition system evaluations, the costs of a false acceptance and a false rejection are quantified, respectively, at 10 and 1, whereas the probabilities that a sample is genuine or impostor are considered to be 99% and 1%, respectively. With these parameters and normalizing the resulting expression, the following equation is obtained (formula 3):

$$C = 0.99 \cdot FRR + 0.1 \cdot FAR \tag{3}$$

For reasons of expediency, however, the present chapter utilizes other criteria that nevertheless enjoy wide use in the field. According to these criteria, $C_{FA} = C_{FR}$ and $P_g = P_i$, such that the resulting cost function may be defined as the following (formula 5):

$$C = FRR + FAR \tag{4}$$

Another value used in the characterization of biometric systems is the equal error rate (EER) which, as shown below, indicates the point at which the error rates are equal:

$$EER(s) = FRR(s) \quad | \quad FRR(s) = FAR(s) \tag{5}$$

As a final concept to consider here, the receiver operating characteristic curve (ROC curve) is a two-dimensional measure of classification performance and is regularly used in the comparison of two biometric systems. The ROC curve represents the evolution of the true acceptance rate (TAR) with respect to that of the FAR (Martin 1997):

$$TAR = f(FAR) \quad where \ TAR = 1 - FRR \tag{6}$$
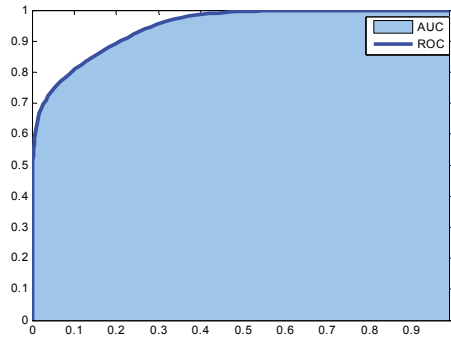
Fig. 13. ROC curve and area under the curve (AUC).

Through the analysis of the ROC curve, the evaluation of a recognition system may be carried out by considering the costs associated with errors even where the latter have not been previously established. In particular, using the area under the convex ROC curve (AUC), system performance may be expressed as a single numeric value and evaluated: the system considered the best being that with the greatest AUC (Villegas et al. 2009)(Marzban 2004).

$$AUC = \int_0^1 ROC(FAR) \tag{7}$$

7. Single scores distribution

Let the simplest case of match score distribution be supposed where, for a single source, scores are distributed according to the following criteria:

$$s \in \Re, f_g = pdf\left(s \mid gen.\right) = N(1,1), \ f_g = pdf\left(s \mid imp.\right) = N(-1,1) \tag{8}$$

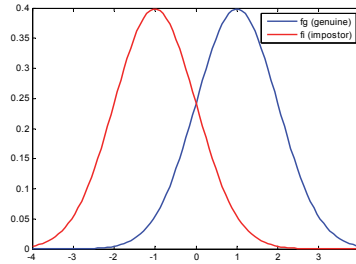Fig. 14. Gaussian distribution of scores.

Given the symmetry of the functions, it can be held that the threshold value minimizing the cost function can be located at *th=0*, point at which FAR and FRR are equal, defining also the EER as shown in formula 9:

$$EER = \int_{-\infty}^{0} f_g\left(s\right) ds = \int_{0}^{\infty} f_i\left(s\right) ds \tag{9}$$

From an estimation, the value *EER=15.85%* is obtained. It is clear, then, that the farther apart the centroids or the smaller the deviations of the distribution functions are, the smaller the error rates.

8. Multiple score fusion

Let it be supposed that match score fusion is to be applied to the results of two processes having generated independent scores (s_1 and s_2) and with distribution functions identical to those described in the previous section of the present chapter. Thus, a match score vector is formed with Gaussian distribution functions for both genuine and impostor subject samples. This vector will have two components, each of which integrating the results from each of the monobiometric classifiers.
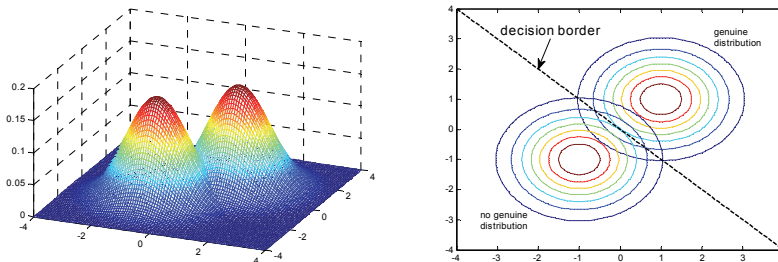

Fig. 15. Representation of two-dimensional Gaussian distribution scores.

$$v = (s_1, s_2) \,|\, s_1, s_2 \in \Re$$
$$f_{g1} = pdf\left(s_1 \,|\, genuine\right) = N(1,1), \quad f_{g2} = pdf\left(s_2 \,|\, genuine\right) = N(1,1) \tag{10}$$
$$f_{f1} = pdf\left(s_1 \,|\, impostor\right) = N(-1,1), f_{f2} = pdf\left(s_2 \,|\, impostor\right) = N(-1,1)$$

In Fig.15, the distribution functions are presented together for both genuine and impostor subject score vectors. Right image represents the contour lines of the distribution functions. Observing it, it seems intuitive that, just as was done in the previous section of the present chapter and applying the criteria for symmetry discussed therein, the best decision strategy is that which takes as a genuine subject score vector any vector found above and to the right of the dotted line which, in this particular case, corresponds to *s1+s2 ≥ 0*. This converts the threshold, for the one-dimension scores, to a boundary line decision in this two-dimension space (an hiperplane if n dimensions space).

$$f_g(v) = f_{g1}(s_1) \times f_{g2}(s_2), \, f_g(v) = f_{f1}(s_1) \times f_{f1}(s_2) \tag{11}$$

Following this, the resulting estimation of the EER is shown in formula 12. In the specific case proposed here, the resulting EER is found to be 7.56% indicating an important improvement owing to the fact that the centroids of the distribution functions have been separated here by a factor of $\sqrt{2}$.

$$EER = \iint_G f_i(v) \, ds_1 \, ds_2, \quad G : s_1 + s_2 \geq 0$$
$$EER = \iint_I f_g(v) \, ds_1 \, ds_2, \quad I : s_1 + s_2 < 0 \tag{12}$$

9. Using gaussian mixture model classifiers

 Gaussian mixture model (GMM) classifiers are used in order to create a model of statistical behaviour represented by the weighted sum of the gaussian distributions estimated for the class of genuine training score vectors and another similar model to represent the class of impostor vectors. Using the two models, the vectors are classified using the quotient of the probabilities of belonging to each of the two classes. If this quotient is greater than a given threshold (established during the system training phase), the vector is classified as genuine. If the quotient is below the given threshold, the vector is classified as an impostor. Such a procedure is quite similar to that discussed in the previous section of the chapter.

In a situation such as that described in the paragraph above, the following points indicate the expectations for a training process and test using GMMs:

- These models (f_g', f_i') of sums of Gaussian functions should maintain a certain similarity to the generative sample distribution ;
- The established threshold may be equivalent to the theoretical decision boundary between genuine and impostor score vectors; and
- Test results clearly approach the theoretic FAR and FRR.

In order to test the fitness of these premises, 1000 two-dimensional random vectors (Vg) following the distribution function of the genuine vectors and another 1000 vectors (Vi) following the distribution function of the impostor vectors have been taken as training data. With these vectors, GMMs were created to approximate the distribution functions[4].

---

[4]For the training and tests of the GMMs performed here a version of EM algorithm has been used.
 http://www.mathworks.com/matlabcentral/fileexchange/8636-emgm

$$f_g \approx f_g^{'} = \sum_{k=1}^{n_g} w_{gk} \cdot N\left(\mu_{gk}, \sigma_{gk}\right), f_i \approx f_i^{'} = \sum_{k=1}^{n_i} w_{ik} \cdot N\left(\mu_{ik}, \sigma_{ik}\right) \tag{13}$$

The models obtained in the training phase for 10 Gaussian models (10G) derived from the simulated data training are presented below in Table 1:

Genuine s_1

K	W	μ	σ	CV[5].
1	0.1374	1.6513	0.1366	-0.0159
2	0.0930	1.5475	0.3065	-0.0689
3	0.0954	-0.2555	0.3610	0.0162
4	0.1504	0.7806	0.1926	0.0199
5	0.0885	2.1518	0.3166	0.0184
6	0.0767	1.1427	0.3610	-0.0333
7	0.0776	-0.1481	0.4024	0.0222
8	0.1362	0.7321	0.1797	0.0017
9	0.0783	0.0929	0.4629	-0.0726
10	0.0666	2.6161	0.3393	0.0316

Genuine s_2

K	W	μ	σ	CV.
1	0.1374	0.9382	0.1875	-0.0159
2	0.0930	-0.2543	0.3520	-0.0689
3	0.0954	0.8522	0.2825	0.0162
4	0.1504	1.5416	0.1641	0.0199
5	0.0885	1.8540	0.3072	0.0184
6	0.0767	2.5772	0.4198	-0.0333
7	0.0776	2.0269	0.3084	0.0222
8	0.1362	0.4505	0.1629	0.0017
9	0.0783	-0.4208	0.3757	-0.0726
10	0.0666	0.5487	0.4818	0.0316

Impostor s_1

	W	μ	σ	CV.
1	0.0806	-1.8355	0.2769	-0.0462
2	0.0811	-0.6579	0.3288	0.0030
3	0.1356	-0.3908	0.1417	0.0027
4	0.0966	-2.3122	0.3943	-0.0498
5	0.1183	-1.4191	0.1553	-0.0279
6	0.0994	0.2029	0.3683	0.0397
7	0.0823	-0.8118	0.3312	0.0032
8	0.0811	-2.1787	0.3847	-0.0317
9	0.0668	0.5181	0.3286	0.0824
10	0.1583	-1.0723	0.2032	0.0163

Impostor s_2

	W	μ	σ	CV.
1	0.0806	0.2294	0.3898	-0.0462
2	0.0811	0.4612	0.3891	0.0030
3	0.1356	-1.3111	0.1581	0.0027
4	0.0966	-0.7795	0.2842	-0.0498
5	0.1183	-1.4853	0.1977	-0.0279
6	0.0994	-0.4434	0.2838	0.0397
7	0.0823	-2.4402	0.3102	0.0032
8	0.0811	-2.0999	0.3894	-0.0317
9	0.0668	-1.7791	0.4830	0.0824
10	0.1583	-0.4784	0.1519	0.0163

Table 1. Gaussian Mixture Model (GMM)

In Fig.16 (left), genuine and impostor models are presented for the score s_1 of the score vector. With red lines indicating the impostor model and black lines indicating the genuine sample model, each of the 10 individual Gaussian distributions with which the GMM classifier approximated the distribution of the training data are represented by the thin lines on the graph. The weighted sums of these Gaussian functions (see Formula 13) are represented by the thick lines on the graph. The result has an appearance similar to two Gaussian distributions around +1 and -1. Fig. 18 (right) shows the contour lines of the two-dimensional models.

For a value of *th = 0.9045* (calculated to minimize) it was found that FAR = 8.22% and FRR = 7.46%.
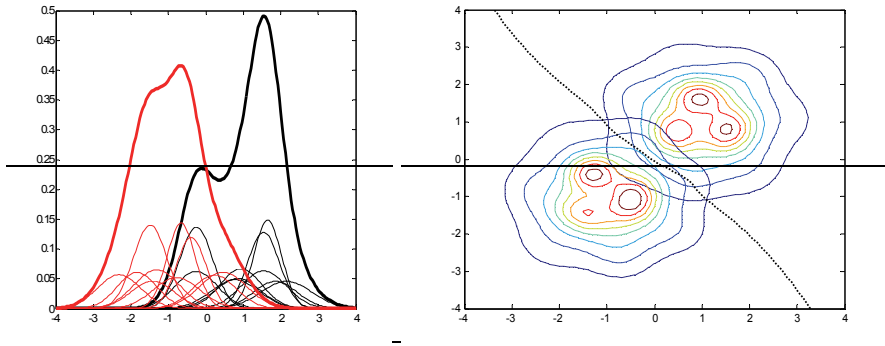
---

[5]Covariance S1-S2 o S2-S1

Fig. 16. GMM with 10G. Single axe (left). Contours for two-dimension model (right).

$$decision(v) = genuine \Leftrightarrow \frac{f_g^{'}(v)}{f_i^{'}(v)} \geq th \qquad (14)$$

In Fig.16. the decision boundary line, at which the quotient of pdfs is equal to the threshold and which separates genuine and impostor decisions, presented as a dotted line. This line is quite near to the proposed boundary. Then the formula 14 represents a transformation from a two-dimension criterion to a one-dimension threshold, which, of course, is easier to manage.

If the same exercise is repeated for a model with 3 Gaussians (3G) and for another with only 1 Gaussian (1G), the following results are obtained:

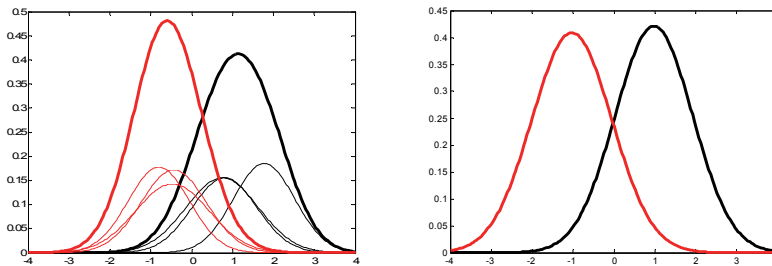| N Gaussian	FAR	FRR	MER[6]	th	AUC
10	8.22%	7.46%	7.84%	0.9045	97.12%
3	7.97%	7.52 %	7.75%	0.9627	97.05%
1	7.99%	7.56%	7.77%	0.9907	97.10%

Table 2. Gaussian Mixture Model (GMM).

Fig. 17. Single axe GMM with 3G (left) and 1G (right).
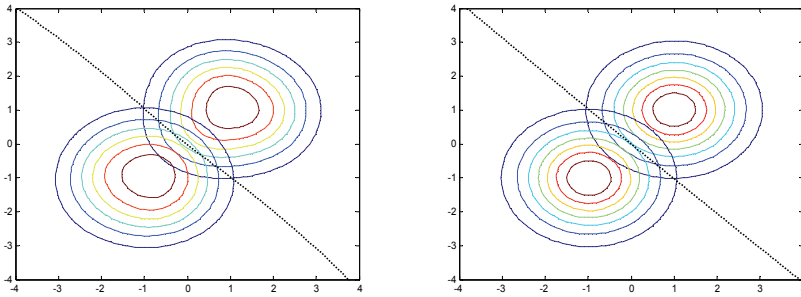
---

[6] MER = (FAR*FRR)/2

Fig. 18. Contour lines for two-dimension GMM with 3G (left) and 1G (right).

Changing the threshold value (see Fig.19), distinct decision boundaries and their corresponding error rates may be obtained. With these values, a ROC curve may be drawn and its AUC estimated.
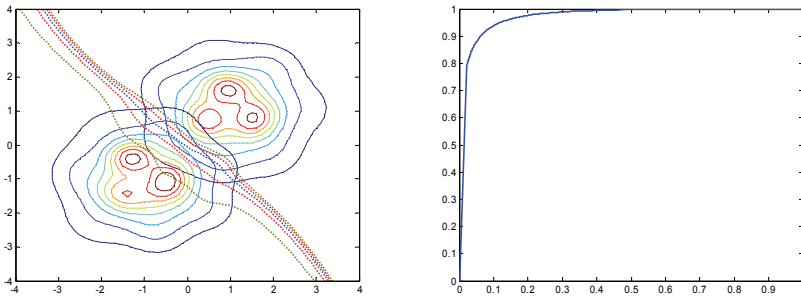

Fig. 19. Various decision boundaries (left) and ROC Curve (right) for GMM 10G.

10. Using support vector machine classifiers

A support vector machine (SVM) is a classifier that estimates a decision boundary between two vector sets (genuine and impostor ones) such that maximizes the classification margin. In the training phase of a SVM, a model is constructed that defines this boundary in terms of a subset of data known as support vectors (SV), a set of weights (w) and an offset (b).

$$decision(v) = genuine \Leftrightarrow -b + \sum_k w_k \left(SV_j \cdot v' \right) \geq 0 \quad [7] \tag{15}$$

The equation above defines the distance of a vector (v) to the boundary, where positive distances indicate genuine samples and negative distances indicate impostor samples[8]. For other kind of boundary lines is possible to select between different kernel functions. Then the general decision function is shown in formula 16, where $K(sv,v)$ represents the adequate kernel function. The kernel implied in formula 15 is called "linear kernel".

---

[7] v' indicates the transpose v vector.
[8] For the examples presented in this chapter, SVM-Light software has been used.

$$decision(v) = genuine \Leftrightarrow -b + \sum_k w_k \cdot K\left(SV_j, v'\right) \geq 0 \qquad (16)$$

Given the data distribution and the fact that the expected separation boundary is a straight line, it may be assumed that the linear kernel is the most adequate kernel function here. Fig.20 shows the distribution of genuine samples (in blue) and impostor samples (in red). Points indicated with circles correspond to the support vectors generated in the training phase. The central black line crossing the figure diagonally represents the set of points along the boundary line, which is also quite close to the theoretical boundary.
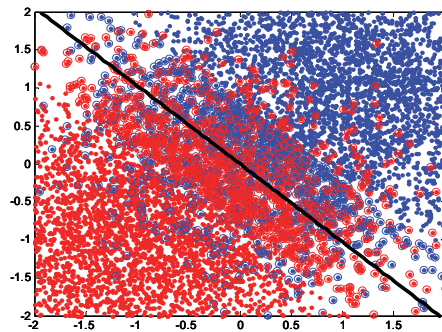

Fig. 20. SVM with linear kernel.

The results of the test data classification demonstrate the performance indicated below for

Kernel	FAR	FRR	MER	nSV[9]	AUC
Linear	8.01%	7.56%	7.78%	1956	95.06%

Table 3. Results of SVM test

The classifier establishes a transformation of the vector space into a real value whose module is the distance from the boundary, calculated such that the system be optimized to establish the decision threshold at the distance of 0. Just as in the case of GMMs, system behavior can be analyzed using the ROC curve and, more specifically, the AUC through the adjustment of this threshold value (see Table 3).

11. Using neural network classifiers

An artificial neural network (ANN) simulates an interconnected group of artificial neurons using a computational model. In this context, a neuron is a computational element that operates n-inputs in order to obtain just one output following a transfer function like the one shown at formula 16. Where s_k is the k-esime neuron input, w_k is the weigth of k-esime input w_0 represent the offset and finaly represents a function (typically sigmoid or tanh) that performs the transference between neurons.

---

[9]nSV: Number of support vectors

$$z = \text{func}(w_0 + \sum_{k=1}^{N} (w_k \cdot s_k)) \qquad (16)$$

A typical ANN groups its neurons in a few layers, so that, the certain layer neuron outputs are only connected to the next layer neuron inputs.

The neural network training step gets as a result the weight for every neuron input that minimizes the error rates.

Then the simplest network is one which has only one neuron with two input and one output (2-1-1). This way, the transfer function has no effect on the system and at the end decision function becomes a linear combination of the inputs and therefore the training estimates a linear separator similar to the one seem before for SVM with linear kernel.

Applying neural networks to above described data, is possible to obtain the following results:[10]

Struct	FAR	FRR	MER	AUC
2-1-1	7.94%	7.62 %	7.78%	95.06%

Table 4. Results of 3 ANN test

12. Beta distributions

One common way in which monobiometric systems present their scores is through likelihood estimates (the probability that the sample is genuine). In such cases, the score rangeis limited to 0-1 (0-100%). Ideally, instances of genuine subject scores would be grouped together around 1 or a point close to 1, while impostor subject scores would be grouped together around zero or near it. Both would demonstrate beta distributions. An example of this ideal situation is plotted in Fig.23 with the pdf for genuine samples follows *Beta(5,1)* and the pdf for impostor samples follows *Beta(1,5)*. Because the symmetrical properties of these functions, the equilibrium point can be clearly located at *s = 0.5* with an solving the integral in formula 1 the *EER = 3.12%*.

As it was done for the Gaussians , identical distribution functions are established for both dimensions of the two-dimension score space, then   a theoretical value of EER= 0.396% would be obtained. Also is possible the same routine and evaluate system performances for GMM, SVM and NN classifiers[11].

Fig. 21 shows the pdf's used in this example and de model obtained for them, while table 5 display test results.

Classifier.	FAR	FRR	MER
GMM 10G	0.55%	0.31%	0.43%
GMM 3G	0.56%	0.28%	0.42%
GMM 1G	0.54%	0.28%	0.41%
SVM Linear	0.48%	0.34%	0.38%
NN (2-1-1)	0.43%	0.35%	0.39%

Table 5. Test results.

---

[10]For the examples with ANN, Neural Network Toolbox™ have been used.
http://www.mathworks.com/products/neuralnet/
[11]For the examples, the same number of genuine and imposter vectors were randomly generated as the previous sections
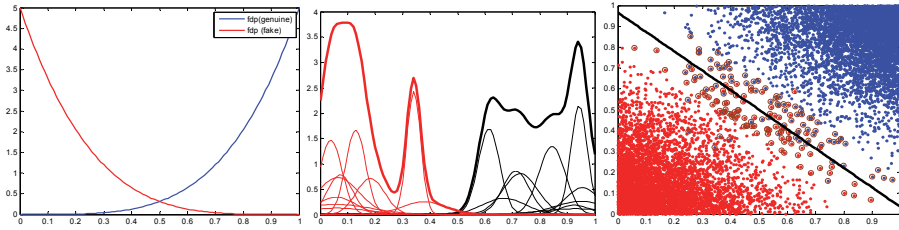
Fig. 21. Single axe pdfs (left), model with 10G (centre), vector plots and SVM linear model.

13. More realistic distributions

Unfortunately, the distribution functions for real scores are not as clear-cut as those presented in Fig. 21. Scores for impostor subject samples, for example, are not grouped around 0, but rather approach 1. Similarly, genuine subject sample scores often tend to diverge from 1. Distributions similar to those in Fig. 22 are relatively common. To illustrate this, *pdf.genuine = Beta (9,2)* and *pdf.impostor = Beta (6,5)* have been chosen for the first score (s_1).
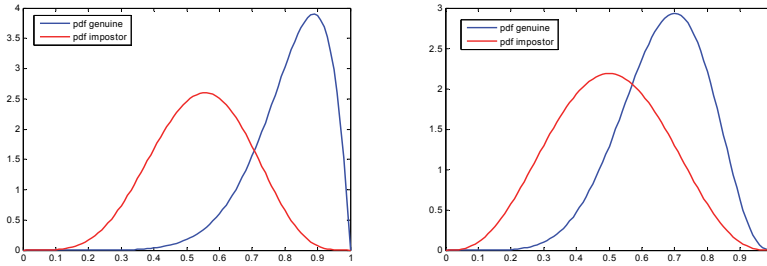

Fig. 22. Single axe score1 pdfs (left) and score2 pdfs (right)..

These particular distributions don't display any symmetrical property then the equilibrium point estimated loking for *FAR = FRR* and as a result the threshold value of *th = 0.7* with an *EER of 15.0%* has been obtained. Eve more, the optimal threshold value does not coincide here with the ERR. Then in order to minimize the cost function threshold must adopt a value of *0.707* yielding the error rates of *FAR = 16.01%, FRR = 13.89%* and *MER = 14.95%*.

In order to further simulate real conditions, score 2 has been supposed here to display a different behavior, to wit, *pdf.genuine = Beta (8,4)* and *pdf.impostor = Beta (4,4)*, as it is displayed in Fig 22

As can be seen, the equilibrium point is found here at a value of *EER = 29.31%* and *th = 0.5896* and the minimum of the cost function at *FAR =34.84%, FRR = 23.13%, MER = 28.99%* and *th = 0.5706%*.

If these two distributions are combined and a two-dimensional score space is established, the resulting pdfs can be represented as the one in Fig.23. It plots these two-dimensional density distributions where de genuine one is found near the point (1,1) while the impostor one is located farther from it.
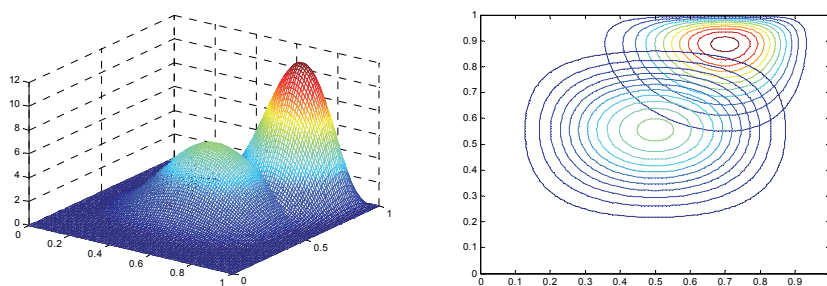
Fig. 23. Combined density distribution, 3D view (left), contour lines (right).

Applying the GMM trainer with 10 Gaussian functions to these distributions, the images in Fig. 24 are obtained representing the set of the 10 Gaussians making up the genuine model; the set of 10 Gaussians making up the impostor model and representing the impostor and genuine models as the weighted sum of each of their Gaussian functions.
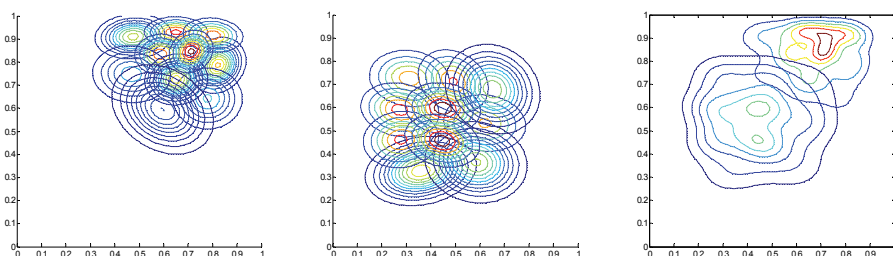
Fig. 24. Contour lines for GMM 10G models. Individual genuine Gaussians (left), individual impostor Gaussians (centre) , both models right (right).

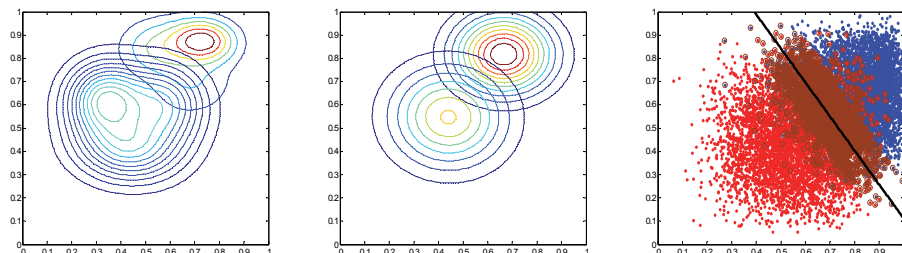Equivalent representations can be obtained using a GMM with 3 Gaussians

Fig. 25. Contour lines for GMM 3G models (left),and GMM 1G models (centre), vectors and SVM boundary (right).

As in previous sections tests conducted with GMM, SVM and ANN classifiers yield the following results:

Classifier	FAR	FRR	MER
GMM 10G	10.19%	9.64%	9.91%
GMM 3G	10.96%	8.94%	9.96%
GMM 1G	10.92%	9.19 %	10.06%
SVM Linear	10.44 %	9.29 %	9.86 %
ANN (2-1-1)	10.49%	9.285 %	9.87 %

Table 6. Results from classifiers test

14. Match score normalization

As described in earlier sections of the present chapter, the data sources in a system of match score fusion are the result of different monobiometric recognition subprocesses working in parallel. For this reason, the scores yielded are often not homogeneous.

In the most trivial case, the source of this lack is different meaning of the scores, they may represent the degree of similarity between the sample and the model or the degree of disimilarity or directly represent the degree of subsystem confidence in the decision made.

Other sources of non-homogeneous scores are include the different numeric scales or the different value ranges according to which results are delivered, as well as the various ways in which the non-linearity of biometric features is presented. Finally, the different statistical behavior of scores must also be taken into account when performing the fusion. For these reasons, the score normalization, transferring them to a common domain, is essential prior to their fusion. In this way, score normalization must be seen as a vital phase in the design of a combination schema for score level fusion.

Score normalization may be understood as the change in scale, location and linearity of scores obtained by distinct monobiometric recognition subprocesses. In a good normalization schema, estimates of transformation parameters must not be overly sensitive to the presence of outliers (robustness) and must also obtain close to optimal results (efficiency) (Nandakumar et al. 2005)(Jain et al 2005)(Huber 1981)

There are multiple techniques that can be used for score normalization. Techniques such as min-max, z-score, median and MAD, double sigmoid and double linear transformations have been evaluated in diverse publications (Snelick et al. 2003) (Puente et al. 2010)**.**

15. Min-max normalization

Perhaps the simplest of currently existing score normalization techniques is min-max normalization. In min-max normalization, the goal is to reduce dynamic score ranges to a known one (tipically: 0-1) while, at the same time, retaining the form of the original distributions.

For the use this technique, it is necessary that maximum and minimum values (*max*, *min*) were provided by the matcher prior to normalize. Alternatively, these may be evaluated as the maximum and minimum data of the values used during system training. In this way, a linear transformation is carried out where 0 is assigned to the minimum value 1 to the maximum value[12]. This transformation function is shown in the following formula 16:

---

[12]Where match scores indicate the difference between a sample and reference, 1 should be assigned to the minimum value and 0 to the maximum.

$$s_k^{'} = \frac{s_k - \min}{\max - \min} \qquad\qquad (16)$$

Applying this transformation to the observations generated in a previous section of the current chapter (see '13. More realistic distributions'), the following results are obtained:

Classifier	FAR	FRR	MER
GMM 10G	17.62%	17.67 %	17.64%
GMM 3G	17.27%	18.47 %	17.87%
GMM 1G	19.49%	16.36%	17.93%
SVM Linear	18.22%	17.31 %	17.77 %
ANN (2-1-1)	16.70%	18.80%	17.75%

Table 7. Result after min-max normalization

16. Z-score normalization

Due to its conceptual simplicity, one of the most frequently used transformations is z-score normalization. In z-score normalization, the statistical behavior of the match scores is homogenized through their transformation into other scores with a mean of 0 and a standard deviation of 1.

$$s_k^{'} = \frac{s_k - \mu}{\sigma}, where\ \mu = \frac{1}{N}\sum_{i=1}^{N} s_i \ and\ \sigma^2 = \frac{1}{N-1}\sum_{1=1}^{N}\left(s_i - \mu\right) \qquad (17)$$

Clearly, it is necessary that the mean and standard deviation of the original match scores be known prior to normalization or, as in min-max normalization, they should be estimated from training data.

Z-score distributions do not retain the forms of the input distributions, save in cases of scores with a Gaussian distribution, and this technique does not guarantee a common numerical range for the normalized scores.

Test results from z-score normalization are shown below:

Classifier	FAR	FRR	MER
GMM 10G	10.13%	9.76 %	9.94%
GMM 3G	10.96%	8.96%	9.96%
GMM 1G	10.92%	9.19%	10.06%
SVM Linear	10.38%	9.33%	9.86%
ANN (2-1-1)	9.43%	10.36%	9.90%

Table 8. Result after z-score normalization

17. Median and MAD

The median and MAD (median absolute deviation) normalization technique uses the statistical robustness resulting from the median of a random distribution to make it less sensitive to the presence of outliers. Nevertheless, median and MAD is generally less

effective than z-score normalization, does not preserve the original distribution and does not guarantee a common range of normalized match scores.

$$s_k' = \frac{s_k - mediam}{MAD}, \text{ where } MAD = mediam\left(\left|mediam - s_i\right|\right) \tag{18}$$

Classifier	FAR	FRR	MER
GMM 10G	10.15%	9.64 %	9.89%
GMM 3G	10.40%	9.51%	9.96%
GMM 1G	10.49%	9.44%	9.97%
SVM Linear	9.94%	9.77%	9.85%
ANN (2-1-1)	9.53%	10.28%	9.90%

Table 9. Test results wiht mediam-MAD notmalization

18. Double sigmoid normalization

In one particular study from the literature, a double sigmoid transformation is proposed as a normalization scheme (Cappelli et al. 2000):

$$s_k' = \begin{cases} \dfrac{1}{1 + e^{-2\left(\frac{s_k - d}{r_1}\right)}} & s_k \le d \\[3mm] \dfrac{1}{1 + e^{-2\left(\frac{s_k - d}{r_2}\right)}} & s_k > d \end{cases} \tag{19}$$

According to the double sigmoid normalization technique, match scores are converted to the interval [0,1]. While the conversion is not linear, scores located on the overlap are nevertheless mapped onto a linear distribution (Fahmy et al. 2008).

Classifier	FAR	FRR	MER
GMM 10G	11.40%	10.04%	10.72%
GMM 3G	10.32%	9.94%	10.13%
GMM 1G	11.19%	10.20%	10.70%
SVM Linear	11.10%	9.66%	10.38%
ANN (2-1-1)	10.70%	9.65%	10.06%

Table 10. Test results for double sigmoid normalization.

19. Double linear normalization

Scores yielded by monobiometric classifiers are interpreted as pair of a decision and confidence. The decision, thus, is made according to the location side of the score is located respect to the threshold while confidence is de distance between them. Thus, the greater the distance to the threshold, the greater will be the weight assigned to the score for the final decision.

Generally, this distance does not enjoy a homogeneous distribution for scores of genuine and impostor observations. As a result, in distributions such as that presented in Fig.23, scores of impostor samples tend to have a greater likelihood than those of genuine scores.

In order to compensate that kind of heterogeneity, a transformation has been proposed in (Puente et al. 2010) to make distributions more uniform around the decision threshold:

$$s_k^{'} = \begin{cases} \dfrac{0.5}{th - \min}(s_k - \min) & s_k \leq d \\ \dfrac{0.5}{\max - th}(s_k - th) + 0.5 & s_k > d \end{cases} \tag{19}$$

GMM	FAR	FRR	MER
GMM 10G	11.32%	9.61%	9.92%
GMM 3G	11.03%	8.90%	9.96%
GMM 1G	10.92%	9.19%	10.06%
SVM Linear	10.44%	9.29%	9.86%
ANN (2-1-1)	9.98%	9.78%	9.88%

Table 11. Test results for double linear normalization.

20. Conclusions

The principal conclusion that can be drawn from the present chapter is undoubtedly the great advantage provided by score fusion relative to monobiometric systems. In combining data from diverse sources, error rates (EER, FAR and FRR) can be greatly reduced and system stability greatly increased through a higher AUC.

This improvement has been observed with each of the classifiers discussed in the present chapter. Nevertheless and in consideration of comparative studies of normalization techniques and fusion algorithms, it can be noted that the specific improvement produced depends on the algorithms used and the specific case at hand. It is not possible, therefore, to state *a priori* which techniques will be optimal in any given case. Rather, it is necessary to first test different techniques in order to pinpoint the normalization and fusion methods to be used.

One final conclusion that stands out is that improvements in error rates are directly linked to the number of biometric features being combined. From this, it may be deduced that the greater the number of features being fused, the larger the improvement will be in the error rates.

21. References

Huber, P. J. (1981). *Robust Statistics* (John Wiley & Sons, 1981).

Martin, A.; Doddington, G. ; Kamm, T. ; Ordowski, M. & Przybocki M. (1997). *The DET Curve in Assessment of Detection Task Performance*, Eurospeech, 1997, pp. 1895–1898.

Cappelli, R. ; Maio, D. & Maltoni, D. (2000). *Combining fingerprint classifiers*, in: Proceedings of First International Workshop on Multiple Classifier Systems, 2000, pp. 351–361.

Ghosh, J. ; *Multiclassifier Systems: Back to the Future*, Proceedings of the Third International Workshop on Multiple Classifier Systems, p.1-15, June 24-26, 2002.

Kumar, H. G. & Imran M. (2010). *Research Avenues in Multimodal Biometrics*. IJCA, Special Issue on RTIPPR(1):1–8, 2010. Published By Foundation of Computer Science.

Ross, S. ; Nandakumar, K. & Jain, A. K. (2006). *Handbook of Multibiometrics*, Springer Publishers, 1st edition, 2006. ISBN: 0-3872-2296-0.

Ross, A. (2007). *An Introduction to Multibiometrics,* Proc. of the 15th European Signal Processing Conference (EUSIPCO), (Poznan, Poland), September 2007.

Ross, A. & Poh, N. (2009). *Multibiometric Systems: Overview, Case Studies and Open Issues*, in Handbook of Remote Biometrics for Surveillance and Security, M. Tistarelli, S. Z. Li and R. Chellappa (Eds.), Springer, 2009. ISBN: 978-1-84882-384-6.

Tejas, J. ; Somnath, D. & Debasis, S. (2009). *Multimodal biometrics: state of the art in fusion techniques*, International Journal of Biometrics, v.1 n.4, p.393-417, July 2009

Li ; S.Z. (2009). *Encyclopedia of Biometrics*. Springer Science + Business Media, LLC. 2009. ISBN 978-0-387-732002-8.

Villegas, M. & Paredes R. (2009). *Score Fusion by Maximizing the Area under the ROC Curve*. IbPRIA '09 Proceedings of the 4th Iberian Conference on Pattern Recognition and Image Analysis. Springer-Verlag Berlin, Heidelberg 2009. ISBN: 978-3-642-02171-8. DOI 10.1007/978-3-642-02172-5_61.

Marzban, C. (2004). *A comment on the roc curve and the area under it as performance measures*. Technical report, The Applied Physics Laboratory and the Department of Statistics, University of Washington (2004)

Doddington, G. ; Liggett, W. ; Martin, A. ; Przybocki, M. & Reynolds, D. (1998). *Sheeps, Goats, Lambs and Wolves: A Statistical Analysis of Speaker Performance* in the NIST 1998 Speaker Recognition Evaluation, Proc. ICSLD 98, Nov. 1998.

Nandakumar, K., Jain, A.K. & Ross, A.A. (2005). *Score normalization in multimodal biometric systems*. Pattern Recognition 38 (1212), 2270–2285 (2005)

Jain, K. ; Nandakumar K. & Ross, A. (2005). *Score Normalization in Multimodal Biometric Systems*, Pattern Recognition, Vol. 38, No. 12, pp. 2270-2285, December 2005. Winner of the Pattern Recognition Society Best Paper Award (2005).

Ross, A. ; Rattani, A. & Tistarelli, M. (2009). *Exploiting the Doddington Zoo Effect in Biometric Fusion*, Proc. of 3rd IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), (Washington DC, USA), September 2009.

Snelick, R. ; Indovina, M. ; Yen, J. & Mink, A. (2003). *Multimodal Biometrics: Issues in Design and Testing*, in Proceedings of Fifth International Conference on Multimodal Interfaces, (Vancouver, 2003), pp. 68–72.

Puente, L. ; Poza, M.J. ; Ruíz, B., García, A. (2010). *Score normalization for Multimodal Recognition Systems*. Journal of Information Assurance and Security, volume 5, 2010, pp 409-417.

Fahmy, M.S.; Atyia, A.F. & Elfouly, R.S. (2008). *Biometric Fusion Using Enhanced SVM Classification*, Intelligent Information Hiding and Multimedia Signal Processing,

2008. IIHMSP '08 International Conference on, vol., no., pp.1043-1048, 15-17 Aug. 2008

Przybocki , M. A. ; Martin, A. F. & Le, A. N. (2006). *NIST speaker recognition evaluation chronicles – Part 2"*, Proc. Odyssey 2006: Speaker Lang. Recognition Workshop, pp. 1 2006.