

# Multimodal Score-Level Fusion Using Hybrid GA-PSO for Multibiometric System

Cherifi Dalila and Hafnaoui Imane,  
 Institute of Electrical and Electronic Engineering  
 University of Boumerdes, Algeria  
 E-mail: dacherifi@yahoo.fr, hafmane@hotmail.com

Nait-Ali Amine  
 LiSSi, EA 3956, Paris-Est Creteil (UPEC)  
 Creteil, France  
 E-mail: naitali@u-pec.fr

**Keywords:** multibiometric, multimodal, fusion, score level, genetic algorithm, particle swarm optimization, hybrid, GA-PSO

**Received:** May 16, 2014

*Due to the limitations that unimodal systems suffer from, Multibiometric systems have gained much interest in the research community on the grounds that they alleviate most of these limitations and are capable of producing better accuracies and performances. One of the important steps to reach this is the choice of the fusion techniques utilized. In this paper, a modeling step based on a hybrid algorithm, that includes Particle Swarm Optimization and Genetic Algorithm, is proposed to combine two biometric modalities at the score level. This optimization technique is employed to find the optimum weights associated to the modalities being fused. An analysis of the results is carried out on the basis of comparing the EER accuracies and ROC curves of the fusion techniques. Furthermore, the execution speed of the hybrid approach is discussed and compared to that of the single optimization algorithms, GA and PSO.*

*Povzetek: Predstavljena je nova optimirna metoda za iskanje uteži pri kombiniranju dveh virov informacij za biometrično prepoznavo.*

## 1 Introduction

It is becoming increasingly apparent that a unimodal system using a single biometric trait is not sufficient to meet a number of system requirements imposed by several large-scale authentication applications. The limitation of unimodal systems, such as noisy sensor data, intra-classvariations, non-universality, vulnerability to spoof attacks and more, can lower the performance of the system, and make it more susceptible to refusing a legitimate user and jeopardizing personal security. Multibiometric systems seek to alleviate some of these drawbacks by consolidating the evidence presented by multiple biometric sources. These systems are expected to significantly improve the recognition performance of a biometric system besides improving population coverage, deterring spoof attacks, and reducing the failure-to-enroll rate. Multibiometric Fusion can be implemented in different scenarios including the type of fused sources and the level at which the fusion occurs. The sources can be multiple-sensors data, multiple-samples, multiple-algorithms, or multiple-modalities.

As for the levels, Sanderson and Paliwal [1] proposed classifying fusion techniques into two categories: pre-mapping and post-mapping fusion. Pre-mapping fusion techniques, such as sensor-level and

feature-level fusion, perform fusion before matching. Post-mapping fusion techniques, such as rank-level, decision-level, and match score-level fusion, perform fusion after matching. In this paper, our work is focused on the fusion of multimodalities at the score level. This scenario is extensively studied in literature because of the relatively easy access to information at this level, and the fusion of the scores output by the different matchers[2]. This offers the best trade-off between accessibility and fusion convenience.

**Paper contribution:** we propose the use of a hybrid algorithm GA-PSO to optimize the weights assigned to the different biometric modalities used in the fusion at the score level.

The idea of the hybrid GA-PSO is to take advantage of both algorithms so as to gain in time performance and obtain a Multibiometric system with an optimum accuracy.

**Paper structure:** The rest of the paper is structured as follow: We present some of the previous works in literature that tackled this problem in the next section. Section 3 gives a brief overview of GA and PSO as well as some essential definitions. In section 4, we describe how the hybrid GA-PSO works and how it is used to obtain optimum biometric weights. Section 5 covers our experiments including the results obtained and a brief discussion. Our conclusions are highlighted in section 6.

## 2 Literature review

In a comparison study, Damousis and al. [3] used four machine learning techniques to fuse face and voice modalities at the matching level; mainly Gaussian Mixture Models (GMMs), Artificial Neural Networks (ANNs), Fuzzy Expert Systems (FESs), and Support Vector Machines (SVMs). Their research concluded that although all four techniques performed well, SVM gave the best accuracies.

The Sum Rule was proposed by Ross et al. [2] to fuse face, fingerprint, and hand geometry modalities. In order to compare this technique, Wang et al. [4] proposed using the Weighted Sum Rule by assigning weights to iris and face score modalities based on their false accept rate (FAR) and false reject rate (FRR). They concluded that the Weighted Sum Rule performs better at increasing the accuracy of recognition than the Simple Sum Rule. Various techniques were studied in order to assign said weights with varying levels of accuracy and performance. A recent trend has been the inclusion of optimization techniques in the fusion process in the hopes of obtaining the optimum of the biometric performance. Genetic algorithms (Gas) have seen a special interest. In the works of Alford and Hansen [5], a fusion of face and periocular biometrics at the score level based on Genetic and evolutionary computations (GEC) was achieved. Their work showed that better accuracies could be reached using this technique. Giot and al. [6] proposed a faster technique to compute the EERs of fused modalities as a fitness function for a Genetic Algorithm. Particle Swarm Optimization (PSO) was used in the works of Raghavendra and al. [7] in order to fuse near infrared and visible images for improved face verification. Mazouni and al. [8] did a comparison in performance of some Multibiometric fusion techniques on face and voice modalities. In their study, GA and PSO were proven to give the best accuracies, especially with degraded datasets. SVM in these cases gave the worst performances.

The work presented in this paper builds on these previous findings and increases the performance of the implemented systems. Since the recognition systems work with thousands of individuals, reducing the computation times is essential. The proposed approach, GA-PSO, strives to achieve this while keeping the performances at their highest. To our knowledge, no previous work employed a hybrid GA-PSO to fuse biometric modalities at the score level in order to gain good accuracies with better computational times.

## 3 Multimodal score level fusion

During score level fusion, scores are combined to generate a single scalar score which is later used to make the final decision. There are several combination schemes to achieve this. These include statistical rule-based methods such as Simple Sum, Max rule, Min rule, Product rule and Weighted Sum.

### 3.1 Score Normalization

With the methods mentioned above, score normalization is required before fusion of scores. Anil Jain and al. [9] showed in their work that both min-max and z-score methods are sufficient techniques but they are very sensitive to outliers. On the other hand, *tanh* normalization method, introduced by Hampel et al. [10] is both robust and efficient. For this purpose, and in our work, the *tanh-estimators* normalization rule was employed.

Given a matching score  $S_i$ , the normalized score  $\tilde{S}_i$  is computed using the following equation:

$$\tilde{S}_i = \frac{1}{2} \left\{ \tanh \left( 0.01 \left( \frac{S_i - \mu_{GH}}{\sigma_{GH}} \right) \right) + 1 \right\} \quad (1)$$

Where  $\mu_{GH}$  and  $\sigma_{GH}$  are the mean and standard deviation estimates, respectively, of the genuine score distribution as given by Hampel estimators.

### 3.2 Genetic Algorithm and Particle Swarm Optimization

In this work, the focus is on finding the optimum weights  $w_m$  for fusion of  $m$  modalities by weighted sum which is defined by:

$$S_{f_i} = \sum_{m=1}^M w_m \times S_i^m \quad (2)$$

Given that  $w_m \in [0, 1]$  and  $\sum_{m=1}^M w_m = 1$ .

**Genetic algorithm** is a well-known and frequently used evolutionary computation technique. This method was originally developed by John Holland et al. [11]. The GA is inspired by the principles of genetics and evolution, and mimics the reproduction behavior observed in biological populations.

In GA, a candidate solution for a specific problem is called an *individual* or a *chromosome* and consists of a linear list of genes. GA begins its search from a randomly generated population of designs that evolve over successive generations (iterations). To perform its optimization-like process, the GA employs three operators to propagate its population from one generation to another.

- 1) **Selection:** In which the GA considers the principal of “survival of the fittest” to select and generate individuals that are adapted to their environment.
- 2) **Crossover:** It mimics mating in biological populations. The crossover operator propagates features of good surviving designs from the current population into the future population, which will have a better fitness value on average.
- 3) **Mutation:** It promotes diversity in population characteristics. The mutation operator allows for global search of the design space and prevents the algorithm from getting trapped in local minima [11].

**Particle Swarm Optimization** is one of the recent evolutionary optimization methods. This technique was originally developed by Kennedy & Eberhart [12] in order to solve problems with continuous search space. PSO uses social rules to search in the design space by controlling the trajectories of a set of independent particles. The position of each particle,  $x_i$  representing a particular solution of the problem, is used to compute the value of the fitness function to be optimized. In fact, the main PSO operator is the velocity update,  $v_i$ , that takes into account the best position, in terms of fitness value reached by all the particles during their paths during its search  $g_{best}$ , and the best position that the agent itself has reached  $p_{best}$ , resulting in a migration of the entire swarm towards the global optimum.

At each iteration, the particle moves around according to its velocity and position; the cost function to be optimized is evaluated for each particle in order to rank the current location. The velocity of the particle is then stochastically updated according to [13]

$$v_i^{t+1} = k \left( \omega v_i^t + C_1 r_1 (p_{best} - x_i^t) + C_2 r_2 (g_{best} - x_i^t) \right) \quad (3)$$

After, the particle position is updated according to

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (4)$$

Where:

- $\omega$  Inertia weight, a parameter controlling the flying dynamics.
- $r_1, r_2$  random variables in the range  $[0, 1]$ .
- $C_1, C_2$  positive constants controlling the related weighting of corresponding terms.
- $k$  Constriction parameter introduced by Clerc and al. [14].

## 4 The proposed hybrid GA-PSO approach

Although GAs have been successfully applied to a wide spectrum of problems, using GAs for large-scale optimization could be very expensive due to its requirement of a large number of function evaluations for convergence. Compared to GA, PSO has some attractive characteristics. It has constructive cooperation between particles; that is, particles in the swarm share information among themselves. On the other hand, a drawback of PSO is that the swarm may prematurely converge. The underlying principle behind this problem is the fast rate of information flow between particles, resulting in the creation of similar particles with a loss in diversity that increases the possibility of being trapped in local optima.

To deal with all these misgivings, and seeing as both GA and PSO work with an initial population of solutions and combining the searching abilities of both methods

seems to be a reasonable approach, we propose a new algorithm, denoted as GA-PSO, that combines the evolutionary natures and social interactions of both algorithms.

To understand the workings of the algorithm, Figure 1 depicts a schematic representation of the proposed hybrid GA-PSO. As can be seen, GA and PSO both work with the same initial population. The hybrid approach picks  $N$  initial individuals that are randomly generated. The  $N$  individuals are sorted by fitness, and, according to a user defined probability  $P_k$ , the set is divided into two sub-sets  $\{\psi_G, \psi_P\}$ . The top set  $\psi_P$  is used to adjust the particles using the PSO algorithm. The other set  $\psi_G$  is fed into the real-coded GA to create new individuals by selection, crossover and mutation operations. Both resulting populations are combined into one single population of  $N$  individuals, which are then sorted in preparation for repeating the entire run.

In our experiments, and in terms of multimodal fusion, the hybrid algorithm generates an initial population of size  $N$  which consists of the weights  $w_i$  defined in equation (2). In this work, we will fuse two modalities at a time to create fusion scores which makes  $m = 2$ .

The fitness function is defined as the Equal Error Rate (EER). As a reminder, the EER is the point at which the error rates FAR and FRR are equal. The goal is to minimize the value of the EER. For every set of individuals  $(w_1, w_2)$ , the EER of the fused scores  $S_f$  is computed. Knowing that the best fitness is the one with the smallest EER, the individuals are then rearranged and sorted. The whole set is split into two sets which will go through the selection, crossover and mutation processes in case of GA, and velocity and position update in case of PSO. Evaluation of the fitness costs of the “offspring” is once again run and the weights to produce the minimum EER value is picked as optima. If the stopping criteria are not satisfied yet, this procedure is repeated until one of the conditions is met. This is summarized in **Algorithm 1**.

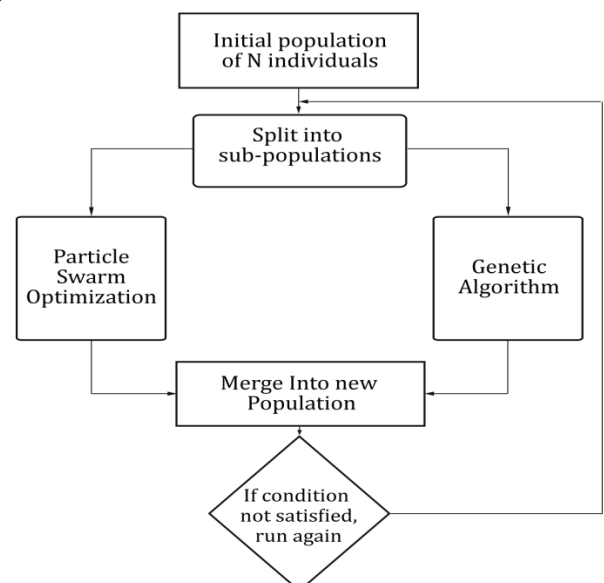


Figure 1: Scheme representation of the Hybrid GA-PSO Algorithm.

**Algorithm 1:** Hybrid GA-PSO to find optimized fusion weights

1. Initialize parameter values
2. Generate random initial population (weights)
3. **While**  $k < iter_{max}$  **do**
4. Evaluate then sort fitness function for every individual based on EER
5. **For**  $i=1:m$
6. Update particle's personal best  $p_{best}$  and global best  $g_{best}$
7. Update particle's velocity and position according to eqs. (3) and (4)
8. **End for**
9. **For**  $i=m+1: end$
10. Select parents to reproduce
11. Generate children through crossover and mutation
12. **End for**
13. Merge the two resulting sub-populations into one population
14. **If** (stopping criteria) **then**
15. Go to 18
16. **End if**
17. **End while**
18. **Return** individuals (fusion weights) that give the best EER

## 5 Results and Discussion

### 5.1 Experiment Setup

Three publicly available Multibiometric databases were used in order to validate the fusion techniques. The NIST BSSR1 Set 1 [15] consists of sets of raw output similarity scores from 517 users of faces and both left and right index live-scan fingerprints coming from the same person. XM2VTS database [16] is built on the XM2VTS face and speech multimodal database, respecting the Lausanne Protocols I and II (LP1 and LP2). LP1 has eight baseline systems and LP2 has five baseline systems. In here, we only deal with the LP1 dataset. The BANCA dataset [17] contains matcher scores of face and speech. There are seven different protocols: Mc, Md, Ma, Ua, Ud, G and P.

In order to validate the aforementioned algorithms and their effectiveness when dealing with Multibiometrics, we split the databases into two separate sets:

- **The training set** which serves to compute the biometric reference of each matcher. In other words, we train the algorithms to attain the optimal weights for each matcher.
- **The testing set** which serves to validate the results of the training by computing the performance of the fusion with the obtained weights.

In our experiments, the hybrid algorithm which combines properties of both GA and PSO runs on the parameters summarized in Table.1.

Parameter	Value
Initial Population size	50
Maximum iterations	50
Splitting probability	$P_k=0.6$
Crossover probability	$P_c=1$
Mutation probability	$P_m=0.05$
Inertia factor	$w=1$
$C_1$ and $C_2$	$C_i=2.05$
Constriction factor	$k=0.73$

Table 1: The parameter values used in the hybrid GA-PSO.

### 5.2 Experimental Results

To compare the performances of the biometric systems, the EER values and Receiver Operating Characteristic (ROC) curves are studied. Table 2 presents the EER values of the single modalities involved in the fusion from each database.

To evaluate its performance, the hybrid is compared to the classical combination rules as well as the single optimization techniques, GA and PSO. Table 3 summarizes the results we obtained from the experiments. Before applying the rules on these scores, they have all been put under the same range  $\{0, 1\}$  using the *tanh-normalization* scheme. The best performance in each of the fused modalities is shown in **bold**.

From the first look, an improvement in accuracy is clearly observed between unimodal and multimodal systems, regardless of the fusion rule applied. In Table 3, even the best matcher in the NIST, Face-C with EER = 4.39%, is outperformed by a simple Max-rule, with an EER = 3.66%.

Figure 2 plots the ROC curves of fused scores using the classical combination rules. We observe that among all these rules, *Simple Sum* gives the better performance even when dealing with degraded data, as is the case of the BANCA Ua subset with (EER = 10.4%).

What interests us is the *Weighted Sum* where the weights associated to the different modalities are optimized through the hybrid GA-PSO. In Figure 3, the ROC curves of fused scores using Simple Sum are plotted against those using GA-PSO. We notice that although *Simple Sum* gave the best results previously, it is outperformed by the hybrid GA-PSO in every dataset. This is not only in terms of EER. From the same figure, we can see that, even when considering the FAR and FRR values, GA-PSO gives better rates.

These results are confirmed in Table 3, where compared to the best EER obtained from Simple Sum in the NIST dataset with the (FaG-FiR) pair (EER = 1.21 %), the improvement in accuracy is quite apparent where the optimizations give a better optimized EER (= 0.43%).

NIST BSSRI				XM2 VTS				BANCA					
Face – G	Finger – R	Face – C	Finger – L	Face 1	Speech 2	Face 5	Speech 3	Face G	Speech G	Face Md	Speech Md	Face Ua	Speech Ua
5.69	4.39	5.52	7.91	1.81	6.61	6.57	4.51	11.32	1.98	10.58	4.33	28.5	15.1

Table 2: EER (%) of unimodal Biometric modalities from NIST, XM2VTS and BANCA

Fusion Technique	NIST BSSRI		XM2 VTS		BANCA		
	FaG – FiR	FaC – FiL	FIS2	F5S3	F – S (G)	F – S (Md)	F – S (Ua)
<i>Max</i>	5.49	3.66	1.45	3.06	2.19	5.45	15.4
<i>Min</i>	5.52	7.91	1.81	6.46	7.32	5.61	28.5
<i>Product</i>	2.70	4.77	1.64	5.66	2.35	3.36	16.9
<i>Simple Sum</i>	1.21	1.00	1.24	3.67	1.82	3.37	10.4
<i>Genetic Algorithm</i>	0.44	<b>0.75</b>	<b>0.87</b>	1.88	1.07	<b>2.24</b>	<b>10.4</b>
<i>Particle Swarm O.</i>	0.62	<b>0.75</b>	<b>0.87</b>	<b>1.85</b>	1.07	<b>2.24</b>	11.1
<i>Hybrid GA-PSO</i>	<b>0.43</b>	<b>0.75</b>	<b>0.87</b>	<b>1.85</b>	<b>0.91</b>	<b>2.24</b>	<b>10.4</b>

Table 3: EERs (%) of fused scores using fusion techniques

Running Time	Genetic Algorithm	Particle Swarm opt.	Hybrid GA-PSO
Time to run 50 iterations	105	220	315
Time to reach a global min.	76	38	12

Table 4: Running time of optimization algorithms in (sec).

Although the performances of GA, PSO and the hybrid GA-PSO are closely similar in most datasets, the employment of the hybrid GA-PSO always reaches optimum weights which in turn gives the best EER values, to the contrary of GA and PSO, which sometimes tend to get stuck on local minima. We notice that even with the degraded data, the execution of this hybrid optimization technique provides good performance rates.

### 5.3 Discussion

When it comes to comparing the optimization techniques to each other, there are not one but many points to consider. It is clear from the results discussed in the previous section and as can be observed in Figure 4, that GA, PSO, and GA-PSO mostly result in the same best accuracies. But they differ in other aspects such as the *time consumption* (see Figure 5). Genetic Algorithm, due to the fact that it covers large search spaces, has a larger computational time. On the other hand, we have PSO that, as a consequence of its fast operations, consumes less computational time but converges quickly to local minima. The hybrid GA-PSO takes advantage of both algorithms where it gains in computational time, by adding the benefit of fast search property of PSO, and still covers the large search space efficiently. This is observed in Figure 5, where the cost function is plotted against the number of iterations run by all three algorithms. It can clearly be noted, with the XM2VTS dataset, that GA-PSO takes much fewer iterations (#iterations = 2) to reach the global optimum than either

PSO (#iterations = 9) or GA (#iterations = 38). Table 4 puts in value the amount of time in CPU-time that each algorithm takes to be executed for 50 iterations and the time to reach a global minimum. It seems, from a first look, that the hybrid algorithm gives the least favorable running time. That is quite logical since GA-PSO computes the cost function three times in one iteration while PSO computes it twice, and GA, once. But when taking into consideration that it takes much fewer iterations to reach a global optimum, it is actually faster than the two other algorithms. After many runs of these programs, it has been noticed that, although GA and PSO mostly give good results, they would occasionally get stuck in local minima, as is the case in Figure.5.b with the NIST dataset. It appears at first glance that GA reached a good place faster than the other algorithms. But in fact, it reached a local minimum and got stuck there. Be that as it may, after giving it more time, it did reach the same global minimum.

On the other hand, the hybrid GA-PSO is observed to always converge to a global point in the shortest time.

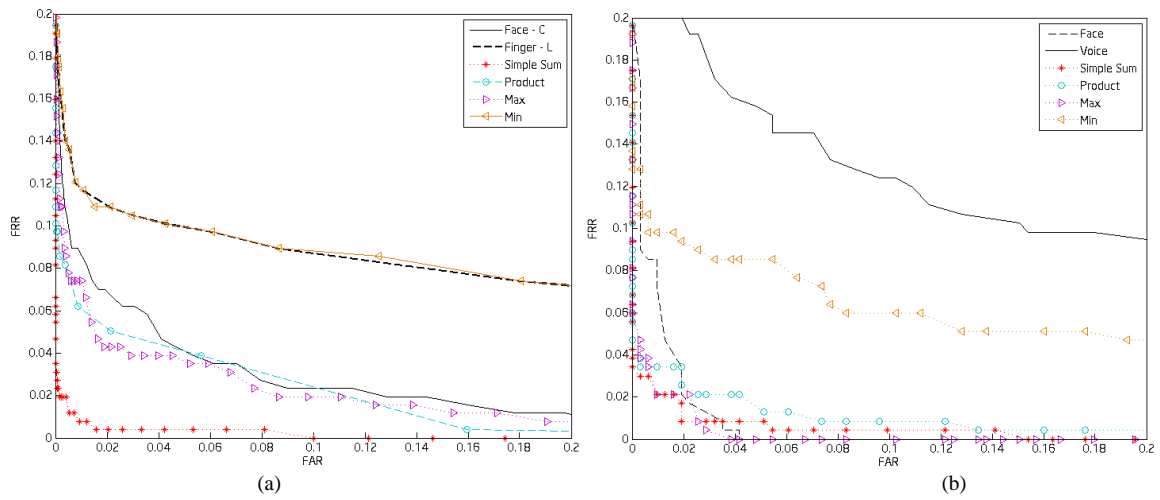


Figure 2: ROC curves of fused scores using classical combination rules from (a) NIST (b) BANCA.

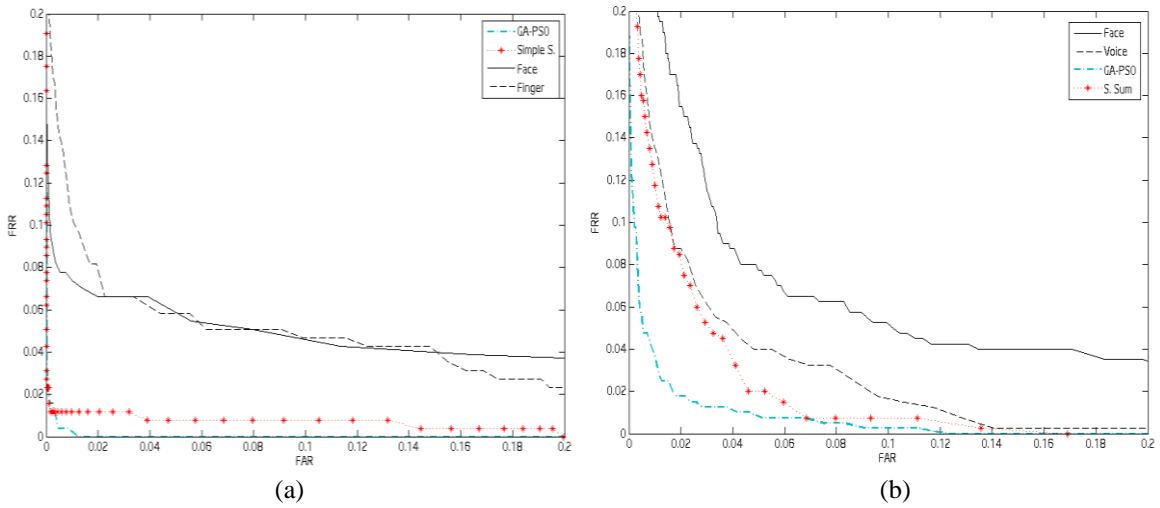


Figure 3: ROC curves of fused scores using hybrid GA-PSO from (a) NIST (b) XM2VTS.

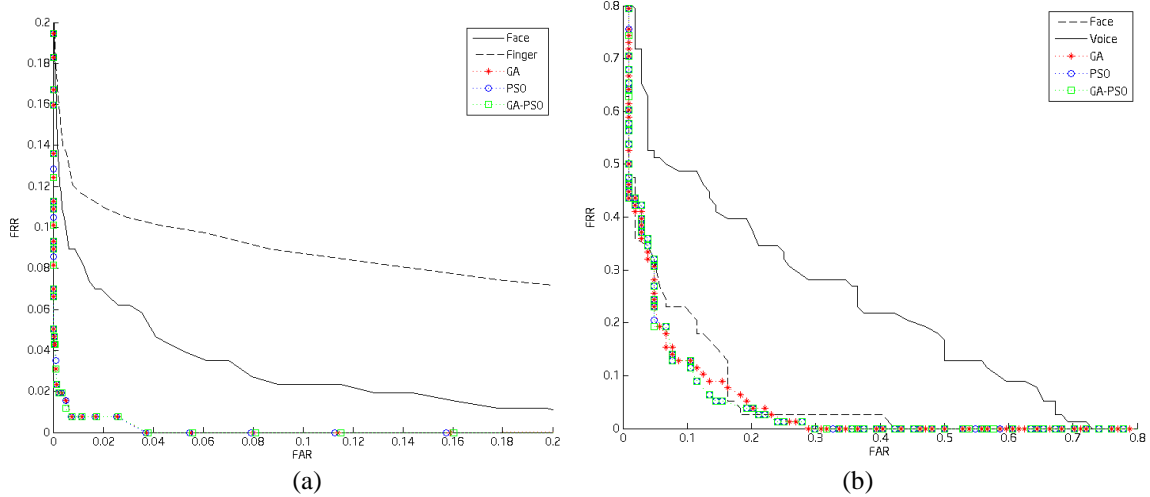


Figure 4: ROC Curves of fused scores from (a) NIST (b) BANCA Databases using Optimization Techniques.

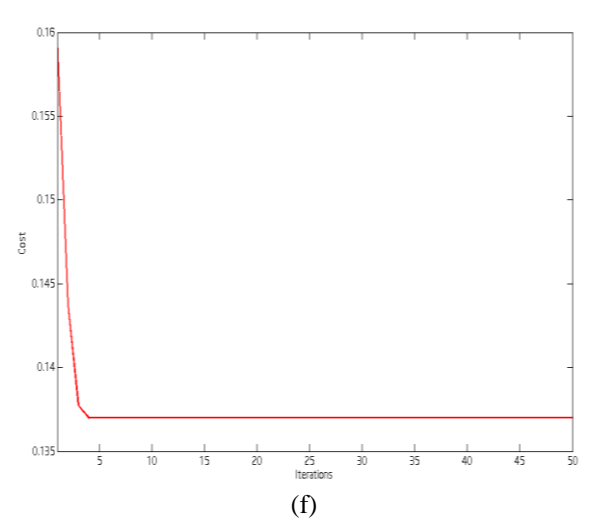
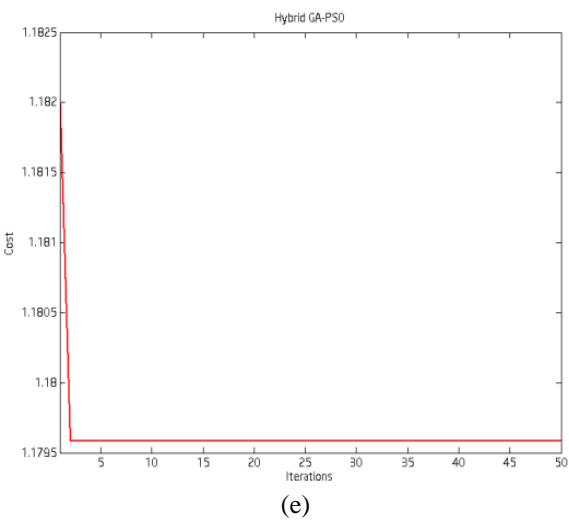
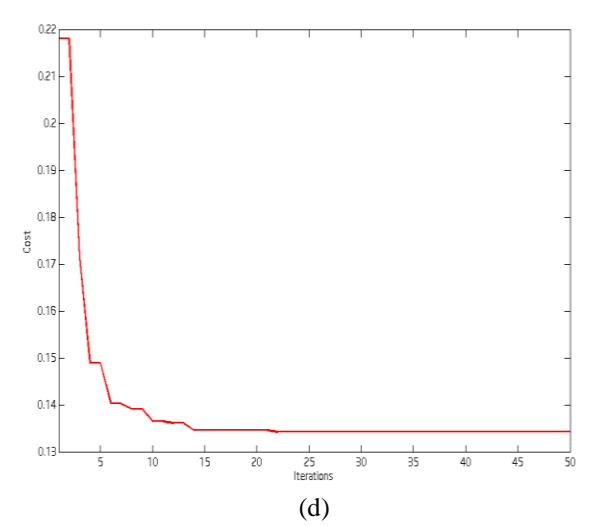
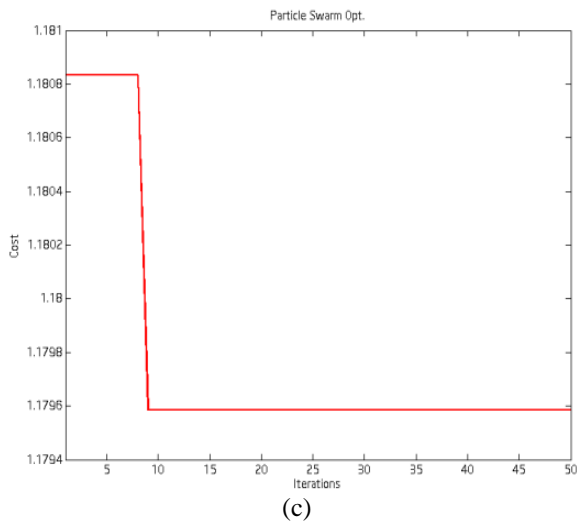
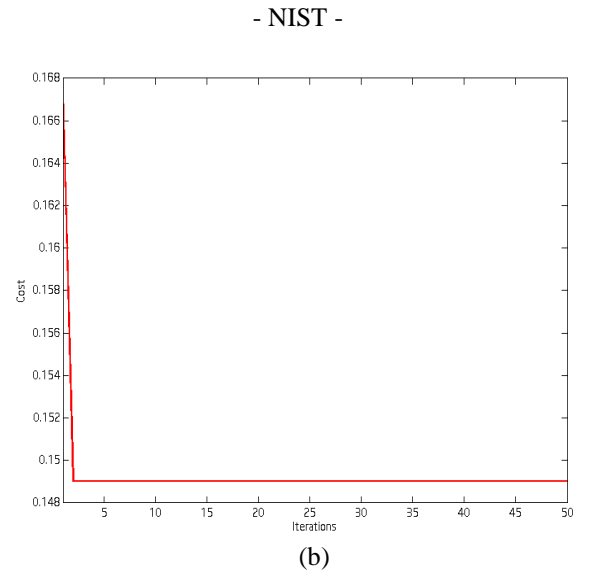
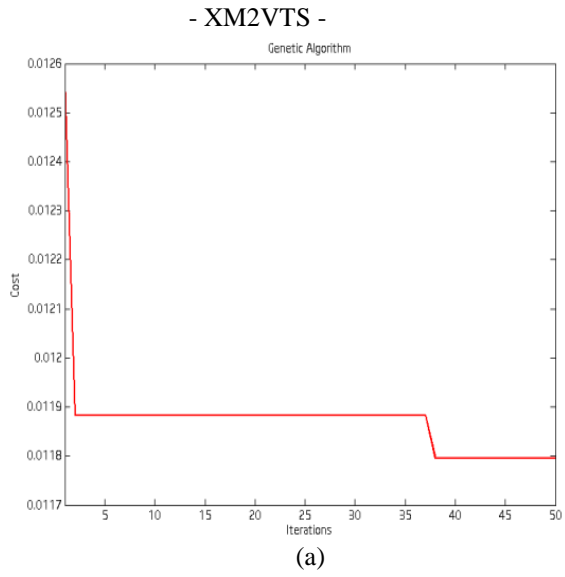


Figure 5: Cost Function vs. Number of Iterations for (a) (b) Genetic Algorithm (c) (d) Particle Swarm Optimization (e) (f) Hybrid GA-PSO.

## 6 Conclusion

This paper proposes a hybrid GA-PSO approach to combining biometric modalities at the score level. With the Weighted Sum rule, the role of the hybrid is to optimize the weights associated with the fused modalities to reach optimum EER values. A normalization based on the tanh-normalization scheme is performed beforehand to put the score modalities on a same unified range. The performance of the hybrid is compared to that of the classical combination rules and the single GA and PSO algorithms. Our results show that the GA-PSO was successful in obtaining much better accuracies on the three different public biometric databases as compared to the classical rules. The time execution of the optimization techniques is also studied. We observe that the GA-PSO outperforms the single GA and PSO in terms of computational time where we find that since the hybrid takes advantage of the properties of both GA and PSO, it assures that the optimum is reached and in the least number of iterations. This makes the hybrid GA-PSO a faster and more robust technique.

## References

- [1] C. Sanderson and K. K. Paliwal, "On the Use of Speech and Face Information for Identity Verification," IDIAP Research Report 04-10, Martigny, Switzerland, 2004.
- [2] A. A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of Multibiometrics*, vol. 6, no. ISBN-13:978-0-387-22296-7. Springer-Verlag, pp. 74–75, 2006.
- [3] I. G. Damousis and S. Argyropoulos, "Four Machine Learning Algorithms for Biometrics Fusion: A Comparative Study," *Appl. Comput. Intell. Soft Comput.*, vol. 2012, pp. 1–7, 2012.
- [4] Y. Wang, T. Tan and A. K. Jain, "Combining Face and Iris Biometrics for Identity Verification", *Proc. 4th Int'l Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 805-813, Guildford, UK, June 9-11, 2003.
- [5] A. Alford, C. Hansen, G. Dozier, K. Bryant, J. Kelly, T. Abegaz, K. Ricanek, and D. L. Woodard, "GEC-based multi-biometric fusion," *IEEE Congr. Evol. Comput.*, pp. 2071–2074, Jun. 2011.
- [6] R. Giot and C. Rosenberger, "Genetic programming for multibiometrics," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 1837–1847, Feb. 2012.
- [7] R. Raghavendra, B. Dorizzi, A. Rao, and G. Hemantha Kumar, "Particle swarm optimization based fusion of near infrared and visible images for improved face verification," *Pattern Recognit.*, vol. 44, no. 2, pp. 401–411, Feb. 2011.
- [8] M. Romaisaa and R. Abdellatif, "On Comparing Verification Performances of Multimodal Biometrics Fusion Techniques" *Int. J. Comput. Appl.*, vol. 33, no. 7, pp. 24–29, 2011.
- [9] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognit.*, vol. 38, no. 12, pp. 2270–2285, Dec. 2005.
- [10] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust statistics: the approach based on influence functions*, vol. 114. John Wiley & Sons, 2011.
- [11] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press, 1975.
- [12] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.
- [13] A. T. Al-Awami, A. Zerguine, L. Cheded, A. Zidouri, and W. Saif, "A new modified particle swarm optimization algorithm for adaptive equalization," *Digit. Signal Process.*, vol. 21, no. 2, pp. 195–207, Mar. 2011.
- [14] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, 2002.
- [15] "NIST biometric score set," *National Institute of Standards and Technology*, 2006. [Online]. Available: <http://www.itl.nist.gov/iad/894.03/biometricscores/>.
- [16] N. Poh and S. Bengio, "Database, protocols and tools for evaluating score-level fusion algorithms in biometric authentication," *Pattern Recognition*, 2006. [Online]. Available: <http://personal.ee.surrey.ac.uk/Personal/Norman.Poh/web/fusion>.
- [17] N. Poh, "BANCA score database." [Online]. Available: [http://info.ee.surrey.ac.uk/Personal/Norman.Poh/web/banca\\_multi](http://info.ee.surrey.ac.uk/Personal/Norman.Poh/web/banca_multi).