Information Sciences 460–461 (2018) 594–610

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Multi-objective optimization for modular granular neural networks applied to pattern recognition

Patricia Melin*, Daniela Sánchez

Tijuana Institute of Technology, Tijuana Mexico

ARTICLE INFO

Article history:
Received 31 March 2017
Revised 8 September 2017
Accepted 10 September 2017
Available online 11 September 2017

Keywords:
Modular neural networks
Granular computing
Hierarchical genetic algorithm
Human recognition
Multi-objective optimization
Pattern recognition
Face recognition
Ear recognition

ABSTRACT

A new method for Modular Neural Network optimization based on a Multi-objective Hierarchical Genetic Algorithm is proposed in this paper. The modular neural network using a granular approach and its optimization using a multi-objective hierarchical genetic algorithm provides better results than when the modular neural network is applied without a granular approach and optimization of parameters. The optimization of different parameters of the modular granular neural network architecture, such as the number of modules (sub-granules), size of the dataset for the training phase, goal error, learning algorithm, number of hidden layers and their respective number of neurons are performed in the proposed method. The fitness functions aim at minimizing the size of the dataset for the training phase and the error using a multi-objective approach. This method can be used in different areas of application, such as human recognition, classification problems or time series prediction. In this case the proposed method is tested with human recognition based on the face and ear biometric measures, where the proposed method aims at finding non-dominated solutions based on the number of data points for training and the recognition error. Benchmark face and ear databases are used to illustrate the advantages of the proposed approach.

1. Introduction

The human body has physiological characteristics that can be used by biometric systems to perform person identification [23,38]. These person identification systems have some advantages over the traditional methods of identification (cards or passwords), and one advantage can be for example, biometric measures cannot be stolen by other persons [21,22]. The most commonly used biometric measures are the face, ear, iris, voice, fingerprint, and hand geometry [18,21,36] and in different works have proven to be a good option to perform person identification [38]. In this paper, the proposed approach is tested in the case of human recognition using the face and ear [33,45] as biometric measures. Artificial neural networks (ANNs) have demonstrated to be an effective technique to learn patterns and they are a good choice in image recognition problems; because the main characteristic of neural networks is that they have the ability to learn complex data. The back propagation algorithm is the most commonly used learning technique used by artificial neural networks (ANNs) [35,43]. The design of ANN architectures can be a complex task and their parameters must be carefully established, because these parameters can affect the final result, but this problem can be easily solved using an optimization method to design the ANN architectures. If for an artificial neural network, the design of its architecture can be considered complex, then when modular neural

* Corresponding author.
E-mail addresses: pmelin@tectijuana.mx (P. Melin), danielasanchez.itt@hotmail.com (D. Sánchez).

http://dx.doi.org/10.1016/j.ins.2017.09.031

networks are used this complexity increases because a modular neural network consists of several modules (i.e. several individual or monolithic ANNs) and each module learns different information.

Modular neural networks (MNNs) and hierarchical genetic algorithms (HGAs) have been used in this paper because in previous works [31,40,45] they have outperformed their conventional counterparts, such as monolithic ANNs and traditional genetic algorithms (GAs), when complex problems are considered.

In [40], modular neural networks and granular computing approaches are fused to create a new concept called modular granular neural network (MGNN). In this case, the optimization of the MGNN architectures is performed seeking the minimization of the recognition error, but the main limitation of this method is that as the optimization of the number of data points for the training phase is performed, if more information (data for the training phase) is given to the MGNN, better is its performance and the testing phase is not a challenge because the MGNN already learned almost all the information. For this reason, in this paper a new optimization method for MGNNs is proposed based on a multi-objective approach.

There are three general approaches to multi-objective optimization (aggregating functions, based on the population and based on the Pareto front concept) [1]. In different works the optimization of artificial, ensemble or modular neural networks architectures has been performed. In some of these works, the optimization is performed considering only one fitness function, and usually is the minimization of the error or by using aggregating functions [31,40]. However, in other works artificial neural networks are optimized using a multi-objective approach, as in [4,30]. The main difference between an ensemble and modular neural networks is that in the ensemble neural network each module learns the same information; meanwhile in a modular neural network each module learns different information. In both kinds of neural networks, the final decision is based on using the highest activation of each module after using the testing data and the highest activation of these activations is taken with its respective class. In this paper, the multi-objective optimization of modular neural networks is performed, using the minimization of two fitness functions. Ensemble or modular neural networks have 2 design phases; the training phase and the testing phase, and usually the minimization of the error is used, but there is always a question of how much data is needed for the training phase to achieve a good result, for this reason a second fitness function is used, to be able to find this parameter, which is one of the advantages that the proposed method provides. For this reason the two fitness functions considered in this approach are the minimization of the recognition error in the testing phase and the number of data points for the training phase with a multi-objective approach based on the Pareto front concept.

Nowadays, many optimization techniques have been proposed, such as particle swarm optimization (PSO) [24], ant colony optimization (ACO) [13], bat algorithm (BA) [50], differential evolution (DE) algorithm [42], firefly algorithm (FA) [52,54] among others. These techniques have been also modified to perform multi-objective optimization, and some variations and different applications of this multi-objective approaches have been proposed respectively, such as Micro-MOPSO [16], MOACO [39], MOBA [51], MODEA [3] and MOFA [53].

Another optimization technique is the genetic algorithm (GA), and some of its multi-objective variants are NSGA [41], NPGA [19], MOGA [15] and MOSASS [26]. There are other approaches based on GA, where other elitism approaches are proposed, such as in PAES [26], PESA [12], SMGA [20], among others.

However, in this paper the proposed multi-objective hierarchical genetic algorithm (MOHGA) is based on a micro genetic-algorithm. This kind of algorithm has a very small population and was proposed by David E. Goldberg [17] and the first implementation was performed by K. Krishnakumar [25] in 1989. There are many algorithms [37,49] based on that original approach, but the proposed MOHGA is based on the work by C. Coello and G. Toscano in [9–11]. Here a reinitialization process and an external memory are used, and the population of the micro-genetic algorithm (μGA) is divided into two parts: replaceable and non-replaceable parts. These and other characteristics are integrated into the proposed Multi-objective Hierarchical Genetic Algorithm MOHGA to perform the optimization of MGNN architectures applied to human recognition based on the face and ear as biometric measures. In general the main motivation of the proposed method is to show that, independently of the database, the optimization of MGNNs architectures can be performed and thus, avoiding the use of other methods such as trial and error, where obtaining optimal architectures is not guaranteed. This method allows having many advantages, such as to obtain better results (lower recognition error), using a smaller data set for the training phase, optimal architectures and as in all the multi-objective approaches to obtain multiple final solutions, where depending on an external restriction only one solution can be considered as the best solution of the Pareto front. In this work the best solution is one that has a recognition rate higher than of 95%, and with the lowest number of data for training. This solution is taken from the optimized solutions with both fitness functions (lower recognition error and smallest data set for the training phase).

This paper is organized as follows. In Section 2, the description of the proposed method is presented. The results obtained using benchmark databases for testing the proposed method are explained in Section 3. In Section 4, statistical comparisons of the results are presented. Finally, conclusions are given in Section 5.

2. Proposed method

This section describes the general architecture of the proposed method. This method proposes a multi-objective hierarchical genetic algorithm (MOHGA) for the optimization of the modular granular neural network architectures.

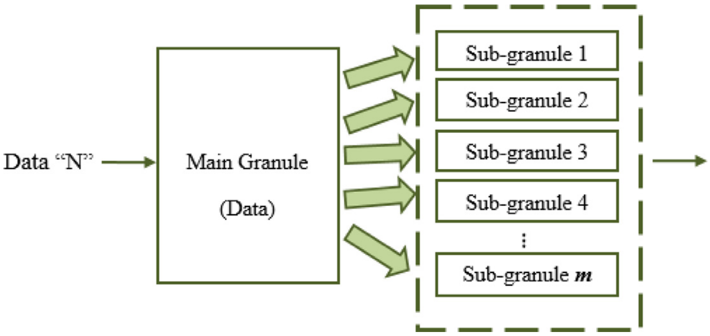

Fig. 1. The granulation method proposed in [40].

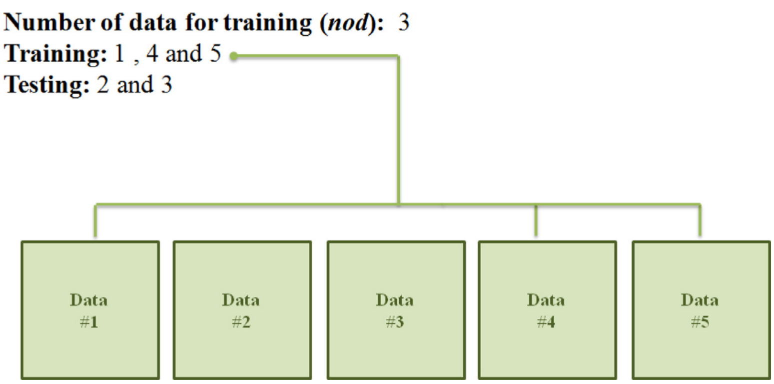

Fig. 2. Example of training and test data selection.

2.1. General architecture of the proposed method

One of the main advantages provided by evolutionary multi-objective optimization is obtaining an entire population of solutions. In this paper, many modular granular neural network architectures, using different data sets for the training phase, will be obtained using this approach. The optimization of modular granular neural networks using a hierarchical approach has been already proposed in [40] using the ear as a biometric measure. In this work, the advantage of using modular granular neural networks (MGNN) over the conventional artificial neural networks (ANNs) is verified, because the division of a problem into smaller sub problems (modules or sub granules) allows achieving better results than with conventional ANNs. The main idea in [40] was to perform the minimization of the recognition error and a new form of granulation was proposed, where the main granule is the whole database, and the database is divided into different number of sub granules, and each sub granule can have different size. This method is illustrated in Fig. 1.

In [40], some parameters of the MGNNs are optimized, such as the number of sub granules (modules), percentage of data for training, error goal, learning algorithm, number of hidden layers and their respective number of neurons. One disadvantage in that work is that we need to know in advance the search space for the optimization of the percentage of data for the training phase, because the evolutions finished easily when the search space allows the use of up to 80 percent of data for the training phase. For this reason two tests were performed; the first using up to 80 and the second using up to 50 percent of data for the training phase. Also a new method to select which data to use for each phase (training and testing) was proposed based on the percentage of data for training. In this paper, the proposed method uses the same approach to select the data, but a very important modification is made and the selection is based on the number of data for the training phase. Depending on the number of data for training (***nod***), the method randomly selects which images will be used for each phase. An example is shown in Fig. 2, where the total number of data (***tnod***) is equal to 5. These images will form the set of images that each granule will use to train each module.

2.1.1. Description of the proposed multi-objective hierarchical genetic algorithm

The idea is the optimization of MGNN parameters such as; the number of sub granules (modules), number of data for training, goal error, learning algorithm, number of hidden layers and their respective number of neurons, but also, the optimal Pareto set is searched.

An optimal Pareto set is a set of solutions that are non-dominated with respect to each other. In this case, non-dominated means that a solution is no worst in any of the objectives and it is better in at least one objective than the others [1]. The optimal Pareto sets can have different sizes and this usually depends on the number of objectives.

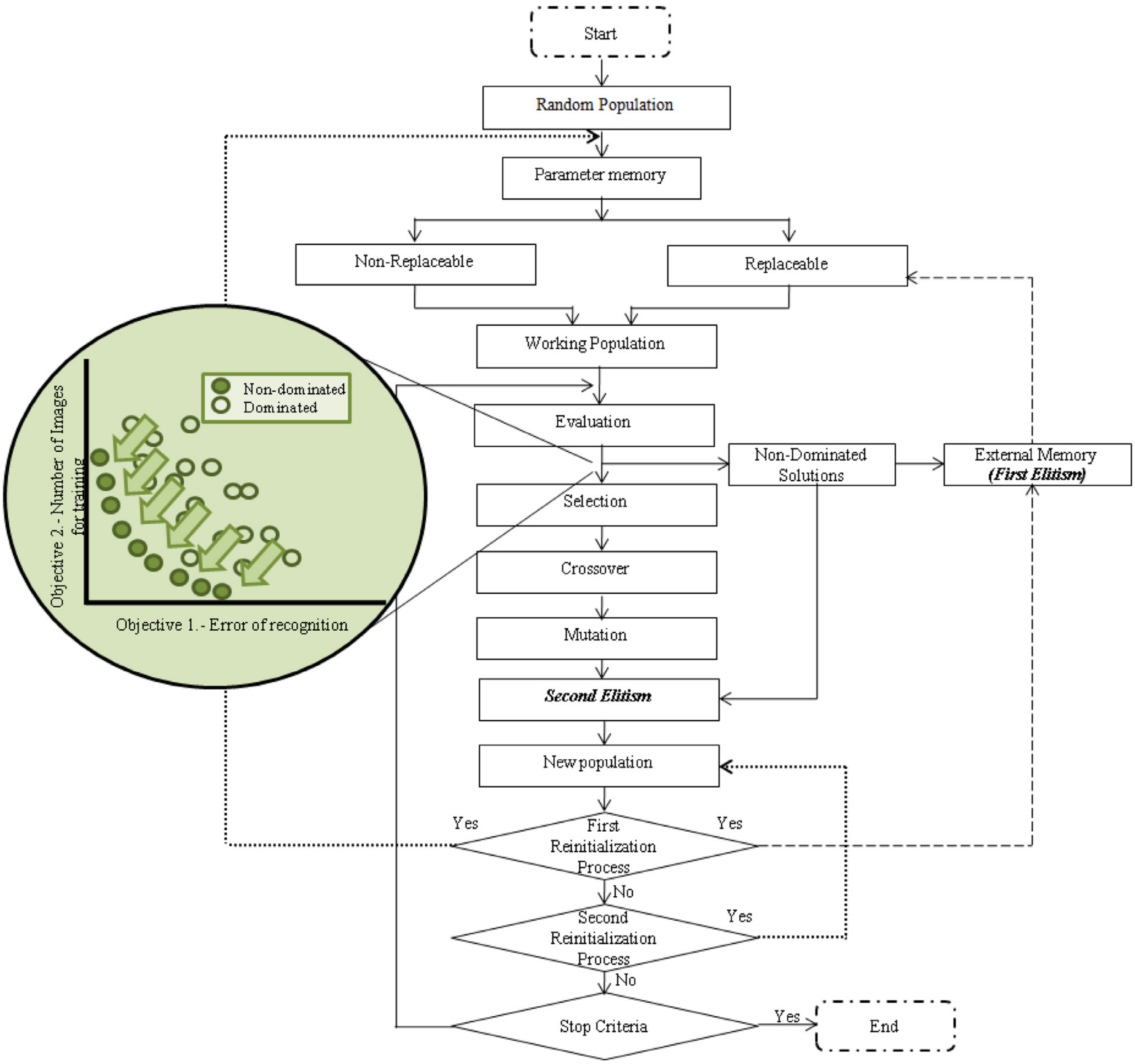

Fig. 3. Diagram of the proposed MOHGA.

As previously mentioned, the proposed MOHGA is based on the first version of the micro-genetic algorithm (μGA) proposed by C. Coello and G. Toscano in [9–11]. They proposed in their algorithm the division of the population in two parts (replaceable and non-replaceable parts), a re-initialization process is activated each 5 generations where the population is reinitialized, an external memory (where the non-dominated solutions are stored), and an adaptive grid to maintain diversity among other characteristics. The proposed MOHGA is based on the reinitialization process, the external memory and the division of the population. The main differences between the μGA and the proposed method are: the second form of elitism, the first re-initialization process (created specifically to be used with neural networks) and the memory of parameters used in the proposed method. These and other processes will be described in detail later. The diagram of the proposed MOHGA is shown in Fig. 3. First, a general population is randomly created, and divided into replaceable (this part will change during the evolution) and a non-replaceable part (this part will never change) and a smaller population (called working population, and the size is established depending on the application, the micro genetic algorithm usually uses 3 to 5 individuals) is taken with certain probability from both parts. The working population is the population used in the evolution. In this case, the well-known conventional genetic operators are also used [28].

In Fig. 4, the chromosome of this multi-objective hierarchical genetic algorithm is illustrated. The proposed MOHGA optimizes: the number of modules, the number of hidden layers of each module, the number of neurons in particular layers, learning algorithm and error goal, using as fitness functions: the minimization of the recognition errors and number

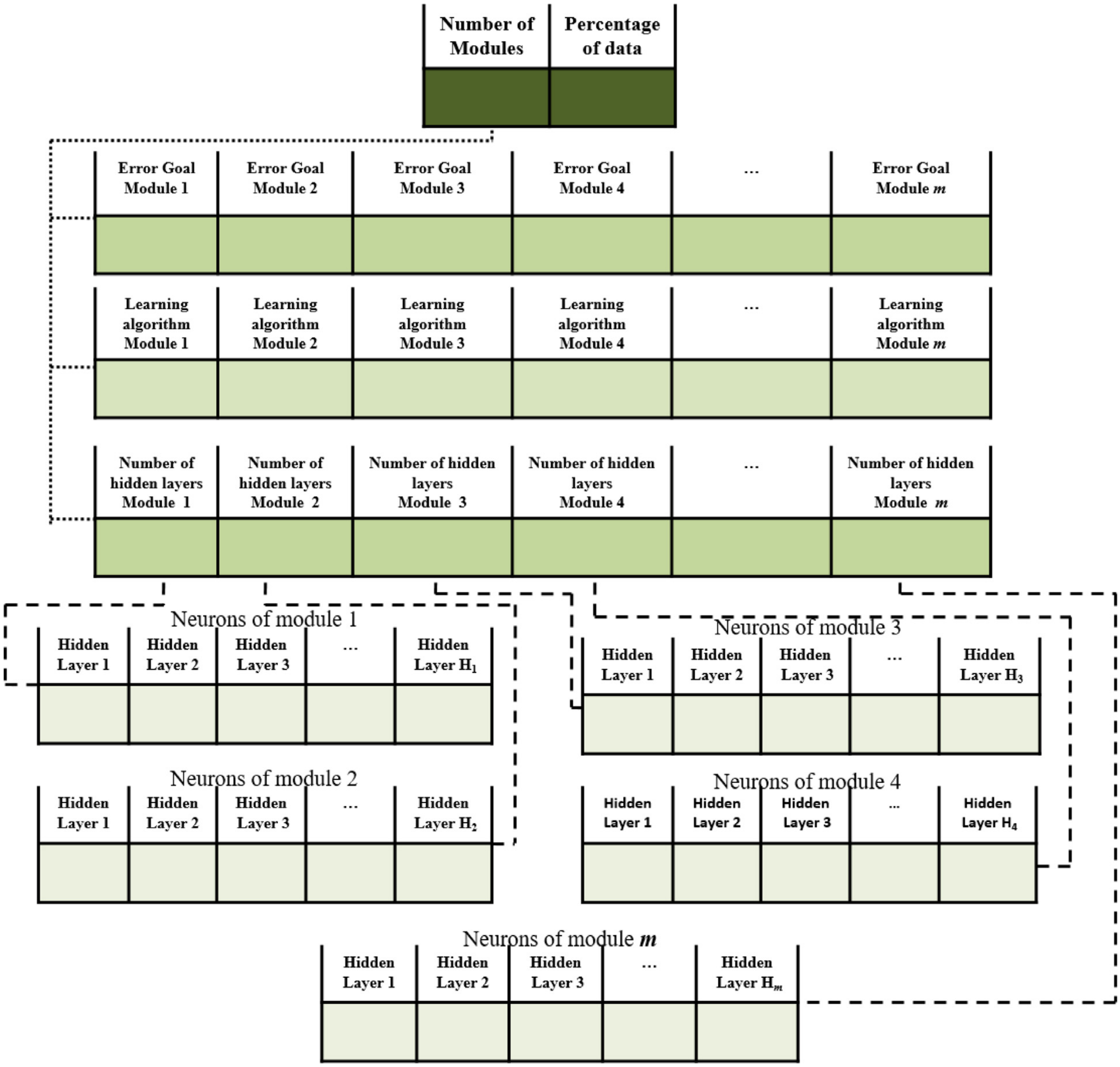

Fig. 4. Chromosome of the proposed MOHGA.

of data for the training phase. The fitness functions are given by the equations:

$$minimize\ f1 = \sum_{i=1}^{m} \left(\left(\sum_{j=1}^{n_m} X_j \right) / n_m \right) \tag{1}$$

$$minimize\ f2 = nod \tag{2}$$

Where $\boldsymbol{m}$ is the total number of modules, $\boldsymbol{X_j}$ is 0 if the module provides the correct result and 1 if not, and $\boldsymbol{n_m}$ is the total number of data points used for testing in the corresponding module. The number of data for the training phase is represented by ***nod***.

The size of the chromosome can be calculated as:

$$\boldsymbol{Size} = 2 + (\boldsymbol{m} \times \mathbf{3}) + (\boldsymbol{m} \times \boldsymbol{h}) \tag{3}$$

Where $\boldsymbol{m}$ is the total number of sub granules (modules) and $\boldsymbol{h}$ is the total number of hidden layers per sub granule (modules) that can be used by the MOHGA. These variables, $\boldsymbol{m}$ and $\boldsymbol{h}$ can be established depending on the application or the particular database.

	Number of data for training				
	1	2	3	4	Average of Evaluations
Numbers of Evaluations	7	3	8	2	5

Fig. 5. Example of the second reinitialization process.

2.1.1.1. Elitism. The proposed multi-objective hierarchical algorithm has two forms of elitism and these are described below.

(1) **First elitism (external memory):** This form of elitism is the same as the one proposed in [9–11], all the found non-dominated solutions are stored in an external memory, in each generation the new found non-dominated solutions are compared with the already stored and the dominated ones are eliminated. In [9–11] the external memory has a limited size, but in this work the size is unlimited.
(2) **Second elitism:** The second form of elitism is very similar to the conventional form of elitism [28], the difference is that randomly in each generation one non-dominated solution is chosen and saved to avoid its modification with the genetic operators and later is reinserted in the new population.

2.1.1.2. Reinitialization processes. The main idea of the reinitialization processes used in the proposed method is to maintain diversity in the population and avoid a recurring problem in genetic algorithms, which is the one of falling into local optima. The reinitialization processes used in the proposed multi-objective hierarchical genetic algorithm are described below.

(1) **First reinitialization process:** The log-sigmoid transfer function is used in the modular granular neural networks and for this reason the MGNN outputs are activations with values between 0 and 1. In MGNNs as in conventional ANNs, when they are used for pattern recognition depending on how many persons or classes are trained in each module, when particular data is tested as output, a vector of neuron activations with values between 0 and 1 is obtained (the vector length is equal to the number of persons or classes learned by the module). To define the recognized class, the position in this vector with the highest value is used. It is assumed that the first position of the vector is the first person or class of the corresponding module and so on up to the last person or class. When each data selected for the testing phase is used, if the data is correctly recognized (i.e. corresponds to the appropriate person or class) the highest activation is saved in another vector. When the testing phase is finished, an average of the elements of this vector is performed and when this average is less than the established average, the re-initialization process is activated. In this process, as in the micro-genetic algorithm (μGA) [9–11] (they activate this process each five generations), the working population is again taken from the non-replaceable and replaceable portions with a certain probability, but first two non-dominated solutions of the external memory are randomly taken and compared against two other elements randomly chosen of the replaceable portion. If the two elements of the replaceable part are dominated by the two elements of the external memory, these are eliminated and replaced by the non-dominated (if the external memory only has one solution the process is performed with only one element). With this process, the replaceable part tends to have more non-dominated solutions.
(2) **Second reinitialization process:** The second reinitialization process is activated each three generations. As a method to maintain the diversity in the data for training (i.e. to achieve that each data is evaluated approximately the same number of times), the average of evaluations of each data set for training is calculated, if one or more are under the average, all the working population in the second gene (the number of data points for the training genes) will be again randomly generated only with the validations that are under the average of evaluations. This process is performed in order to be sure that the population will explore all possible data, so the best solutions will be distributed on the Pareto front. In Fig. 5, an example of the number of evaluations is illustrated, with the established criterion the number of data will be only generated with 2 and 4 data elements for the training phase. It is important to say that for this example the total number of data (***tnod***) is 5, but at least one data must not be learned by the modular granular neural networks (i.e. that data will be used for the testing phase).

2.1.1.3. Memory of parameters. In [28] some parameters for the genetic operators are recommended, and have demonstrated to achieve good results [31,40], but here a memory of parameters is proposed, and this memory consists in saving some parameters and how many non-dominated solutions were found, because the main objective of this memory is to find those parameters that allow having more non-dominated solutions. The memory of parameters has a fixed size, and this memory is activated with the second reinitialization process (each three generations). Each three generations a random decision is performed of either using a new set of parameters for the genetic operators or if they will be taken from the ones already established in the memory of parameters, and if they were randomly established are stored in the parameter memory. As previously mentioned, the parameter memory has a fixed size, and when this memory is already full, a set of parameters can be replaced with another set if it achieves more non-dominated solutions.

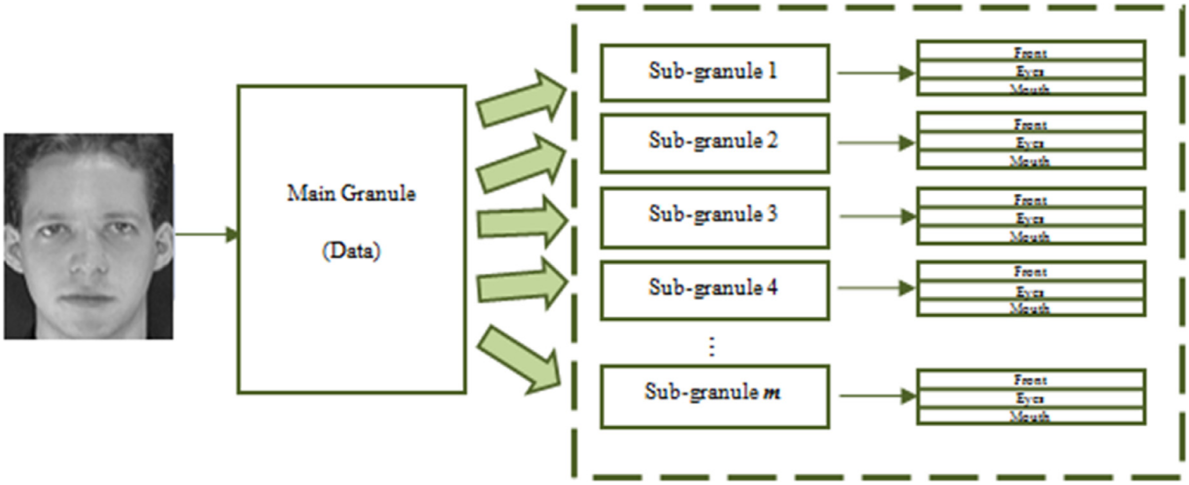

Fig. 6. The granulation applied for the face.

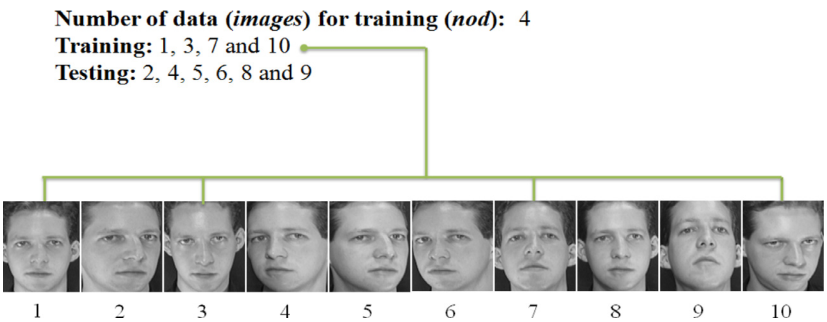

Fig. 7. Example of training and testing data selection.

Table 1
Parameter values of the MNNs.

Parameters of MNNs	Minimum	Maximum
Modules (m)	1	10
Number of images for training	1	***tnod*** - 1
Error Goal	0.000001	0.001
Learning Algorithm	1	3
Hidden Layers (h)	1	5
Neurons for each hidden layers	20	300
Epoch	–	2000

2.2. General architecture of the proposed method applied to human recognition

The human recognition based on the face and ear was used to test the proposed method. In Fig. 6 the architecture of the modular granular neural network based on [40] is used, where the final integration is performed using the winner takes all method. This method consists in taking the highest activation of each module and finally comparing them and the class with the highest activation is the final answer. The granulation can be from 1 to m sub granules (modules), each of the sub granules can have different size, which means; each sub granule will have different number of persons. In this case each sub granule is divided into 3 sub modules because this biometric measure has three regions of interest (front, eyes and mouth).

Some parameters of the MGNNs will be optimized, such as; number of sub granules (modules), size of data set for the training phase, goal error, learning algorithm, number of hidden layers and their respective number of neurons, but these parameters can also be established randomly. In Fig. 7, an example of the method to select the images is shown.

In this paper, non-optimized trainings and the results obtained by the MOGHA are presented to compare results. For the non-optimized trainings and the evolutions, the minimum and maximum values used for the search space applied to human recognition based on face and ear are shown in Table 1. The size of the data set (***tnod***) depends on (when the method is applied to human recognition) the number of images for each person.

Table 2
Parameters of the MOHGA.

Genetic operator	Value
General population	30
Non-replaceable part	25
Replaceable part	25
Working population	5
Maximum number of generations	30

Table 3
Values for the genetic operators.

Genetic operator	Value
Type of selection	Roulette wheel or Stochastic universal sampling
Selection rate	0.5–0.9
Type of crossover	Single point Or Multi point
Crossover rate	0.4–0.9
Type of mutation	Real
Mutation rate	0.001–0.05

As learning methods, the backpropagation algorithms have demonstrated to have a good and faster performance than others [31,40,45] and are considered for this reason to perform the simulations:

1. Gradient descent with scaled conjugate gradient (SCG).
2. Gradient descent with adaptive learning and momentum (GDX).
3. Gradient descent with adaptive learning (GDA).

For this application, the average calculated using the activations vector for the activation of the first reinitialization process is 0.850. The stopping criterion in this MOHGA is when the maximum number of generations is achieved. As previously mentioned, a general population is randomly created, and divided into a replaceable and a non-replaceable part, in this case both parts were equitably divided and a smaller population (the working population) is taken randomly from both parts. In Table 2, some parameters of the MOHGA are presented.

The parameter memory of each evolution performed in this paper can store 5 sets of parameters. The ranges for the genetic operators are presented in Table 3, i.e. when the parameters are randomly selected they are established in these ranges.

2.2.1. Databases

In order to test the effectiveness of the proposed method, three databases are used to perform the human recognition using the face and ear as a biometric measures. In [40], the ear was used as biometric measure; in this paper that database is also used to compare the proposed optimization. The databases used are described below. The pre-processing is used as support, but it can be optional or it can be changed, for example grayscale, binary or color images can be used without any problem. As an example, pre-processing is not applied for the first database, but with the second database different preprocessing techniques are tested, and for the third database the pre-processing described in [40] is used.

2.2.1.1. ORL database. The ORL database [5] contains 40 persons and 10 images correspond to each person. The image dimensions are 92×112, in PGM format. A resizing of 100×100 pixels is performed to each image. An example of this database is presented in Fig. 8.

2.2.1.2. FERET database. The FERET database [34] contains 11,338 images from 994 persons. The image dimensions are 512×768, in PGM format. A sample of this database is shown in Fig. 9.

2.2.1.3. Ear database. The database from the Ear Recognition Laboratory of the University of Science & Technology Beijing (USTB) was used to test the proposed method and compare with the results obtained in [40]. The database contains 77 persons (each person has 4 images of one ear). The image dimensions are 300×400, BMP format [44]. Fig. 10 shows a sample of the images.

2.2.2. Pre-processing for the FERET database

The Viola-Jones algorithm [46,47] was used to detect the face in each image, because the images are very large and a resizing of 100×100 pixels is also performed to each image. Finally each image is converted to grayscale and divided into forehead, eyes and mouth. In Fig. 11, the pre-processing process is illustrated.

Fig. 8. Sample faces of the ORL database.

Fig. 9. Sample of the FERET database.

Fig. 10. Sample of the ear recognition laboratory from the USTB.

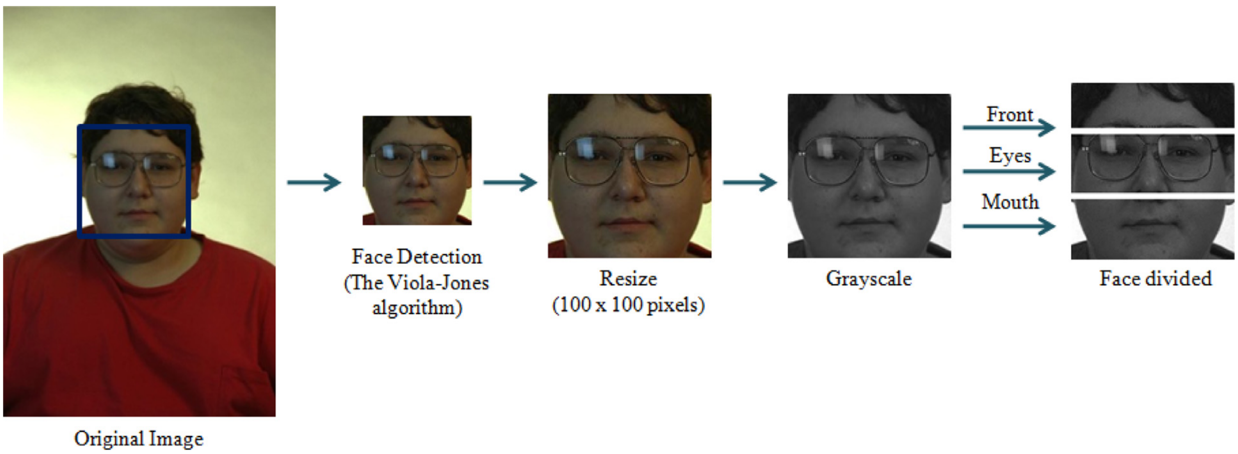

Fig. 11. Sample pre-processing performed to the images of FERET Database.

Table 4
Summary of non-optimized results.

Database	Best (Error/Rec. Rate)	Average (Error/Rec. Rate)
ORL	0.08750	0.55961
	91.25%	44.04%
USTB	0	0.00303
(3 nod) [40]	100%	99.69%
USTB	0.01948	0.05180
(2 nod) [40]	98.05%	94.81%
Case #1	0.16216	0.74872
(FERET)	83.78%	25.13%
Case #2a	0.24000	0.70736
(FERET)	76.00%	29.26%
Case #2b	0.28000	0.70008
(FERET)	72.00%	29.99%
Case #2c	0.50667	0.73907
(FERET)	49.33%	26.09%
Case #2d	0.20500	0.69518
(FERET)	79.50%	30.48%

3. Experimental results

In this section the experimental results are presented. The ORL Database, FERET Database and Ear Database from the USTB were used to compare results with other works. The second database is divided into 4 cases because in other works [2,6,8,14,27,29,32,48] different number of persons were used. Each case for the FERET Database is described below:

1. Case #1: 74 persons are used, each person has 4 images.
2. Case #2: Each person has 7 images.
   (a) 50 persons
   (b) 100 persons
   (c) 150 persons
   (d) 200 persons

In the performed experiments, 30 Non-Optimized trainings and 5 evolutions were achieved for the ORL Database and for each case established for the FERET Database. For the Ear Database only the 5 evolutions were performed, because the non-optimized trainings were already performed in this way in [40].

3.1. Non-optimized results

When the design of the architecture of a neural network is established randomly (another method could be just trial and error) i.e. a method to find an optimal design of the architecture is not used, then the "Non-Optimized" term is used, but to find the best designs of architectures when these are randomly established is very difficult, unless we have worked previously with the same database, but if a general method is proposed, it is important to find the optimal architectures for each different database used. The search space to perform the non-optimized training is the same used to perform the optimization using the proposed method. This is because the authors want to perform a fair comparison between the non-optimized and the optimized results. In this section the non-optimized results are shown, and the design of the modular granular neural network architectures is randomly established. The best and average results for each database without the proposed MOHGA method are shown in Table 4.

3.2. Optimized results

The results obtained with the larger database (FERET Database, Case 2d) are shown in this section. In Section 3.3, the final results obtained with the ORL Database, Ear Database (USTB) and the other cases are also shown. First, it is important to say that in all the cases for us, the best solution is taken from the already found non-dominated solutions (Pareto front) and as external criterion it is established that a recognition rate higher than 95% is needed, and with the smallest data set for training. For Case 2d, the best architecture and memory of parameters of the evolution with the minimum average of error are shown.

3.2.1. Optimized case #2d results

The non-dominated solutions found are presented in Table 5. The recognition errors, and their respective recognition rate and the number of non-dominated solutions found in each evolution are shown. The final memory of parameters is presented in Table 6.

Table 5
FERET database (Case #2d, non-dominated solutions).

Evolution Images for training	1	2	3	4	5
1	0.53083 (46.92%)	0.53250 (46.75%)	0.56000 (44.00%)	0.53250 (46.75%)	0.53250 (46.75%)
2	0.32800 (67.20%)	0.37600 (62.40%)	0.36700 (63.30%)	0.37600 (62.40%)	0.37600 (62.40%)
3	0.27625 (72.38%)	0.24250 (75.75%)	0.21750 (78.25%)	0.24250 (75.75%)	0.24250 (75.75%)
4	0.19500 (80.50%)	0.16500 (83.50%)	0.17500 (82.50%)	0.16500 (83.50%)	0.16500 (83.50%)
5	0.11500 (88.50%)	0.07250 (92.75%)	0.08000 (92.00%)	0.07250 (92.75%)	0.07250 (92.75%)
6	0.10000 (90.00%)	0.03000 (97.00%)	0.03000 (97.00%)	0.06000 (94.00%)	0.03000 (97.00%)
Average	0.25751 (74.25%)	0.23642 (76.36%)	0.23825 (76.18%)	0.24142 (75.86%)	0.23642 (76.36%)
Execution time (HH:MM:SS)	30:56:12	31:32:34	30:13:54	33:11:23	32:14:43
Number of non-dominated solutions	6	6	7	7	6

Table 6
Parameter memory (Case #2d, Evolution #2).

Selection	Selection rate	Crossover	Crossover rate	Mutation rate
Stochastic universal sampling	0.9213	Multi point	0.5213	0.0094
Stochastic universal sampling	0.8513	Multi point	0.4556	0.1323
Roulette wheel	0.6833	Single point	0.9093	0.0048
Stochastic universal sampling	0.8161	Multi point	0.5324	0.0234
Roulette wheel	0.4633	Multi point	0.8599	0.0120

Table 7
The best solution (Case #2d, Evolution #2).

Solution	Persons per module	Num. hidden layers and num. of neurons	Number of images for training	Error / Rec. Rate
6	Module #1 (1 to 17) Module #2 (18 to 25) Module #3 (26 to 41) Module #4 (42 to 75) Module #5 (76 to 114) Module #6 (115 to 136) Module #7 (137 to 149) Module #8 (150 to 186) Module #9 (187 to 200)	*3*(102,99,21) *2*(132,101) *2*(13,156) *2*(43,109) *2*(66,123) *2*(92,31) *3*(39,123,111) *2*(156,131) *3*(95,31,101)	6 (1, 2, 3, 4 5 and 7)	0.03000 (97.00%)

In Table 7, the best architecture of the modular granular neural networks and its obtained results in evolution #2 are presented.

A graphical comparison between non-optimized and optimized trainings is illustrated in Fig. 12.

3.3. Summary of results

The main purpose of this section is to show the summary of results, execution times and a brief comparison with other works using the same databases with the same number of persons and images per person, and also the recognition rates are shown. As previously mentioned, it is important to say that in all the cases for us, the best solution is the one that has a recognition rate higher than 95%, and with the smallest possible data set for training. In Table 8 the best solutions for each case are presented.

To calculate the total number of images used for each database (in each case) we need to know the number of persons used and the number of images per each person. The total numbers of images used for each database sorted in ascending order are shown in Table 9.

The total number of images is very important when the proposed method is used, because the execution time mainly depends on this number, and also the number of generations and the number of individuals used for the evolution are considered. The execution time complexity of this algorithm would be $O(GMN^2)$, where G is the number of generations, M is the number of Individuals used in the MOGHA, and N^2 is the total of images. The number of fitness functions in this case is not used, because this is not a factor which increases the complexity. The execution times for each evolution of each

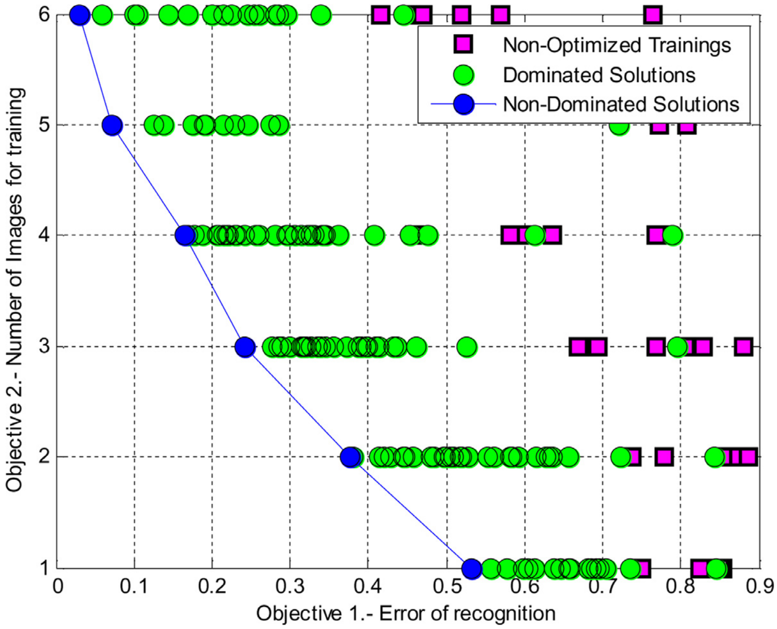

Fig. 12. Comparison evolution #2 and non-optimized trainings (Case #2d).

Table 8
Summary of results.

Database	Number of images (***nod***)	Error of recognition (Recognition rate)
ORL	2	0.02188 (97.81%)
Case #1	3	0 (100%)
Case #2a	5	0.03000 (97.00%)
Case #2b	5	0.04000 (96.00%)
Case #2c	6	0.01333 (98.67%)
Case #2d	6	0.03000 (97.00%)
Ear database (USTB)	2	0.01299 (98.05%)

Table 9
Total images for each case.

Database	Persons	Images per person	Total of images
Case #1 (FERET)	74	4	296
Ear Database (USTB)	77	4	308
Case #2a (FERET)	50	7	350
ORL Database	40	10	400
Case #2b (FERET)	100	7	700
Case #2c (FERET)	150	7	1050
Case #2d (FERET)	200	7	1400

database are presented in Table 10. The best, average and worst execution times of each case are illustrated in Fig. 13. It is important to say that the calculation of the computational complexity can be difficult and perhaps the time variation can be large, because of the usage of modular neural networks and the back propagation algorithm, and perhaps the training phase can be (depending on hidden layers, the number of neurons, the learning algorithm, the error goal, or the initial weights of the back propagation algorithm) faster or slower.

The specifications of the computer used to perform the experimental results presented in this work are summarized in Table 11.

Table 10
Execution times.

Database	Evolution					Best	Average	Worst
	1	2	3	4	5			
Case #1 (FERET)	8:18:27	8:45:23	8:57:07	8:12:25	9:16:47	8:12:25	8:42:02	9:16:47
Ear Database (USTB)	9:24:33	9:18:37	8:55:16	9:21:12	9:25:59	8:55:16	9:17:07	9:25:59
Case #2a (FERET)	8:42:33	9:39:38	10:52:30	9:59:47	8:03:55	8:03:55	9:27:41	10:52:30
ORL Database	9:11:33	10:02:38	10:34:30	9:43:47	9:22:55	9:11:33	9:47:05	10:34:30
Case #2b (FERET)	15:41:30	16:26:58	15:37:06	16:55:36	14:44:14	14:44:14	15:53:05	16:55:36
Case #2c (FERET)	23:46:18	23:15:52	23:30:52	19:46:09	19:09:49	19:09:49	21:53:48	23:46:18
Case #2d (FERET)	30:56:12	31:32:34	30:13:54	33:11:23	32:14:43	30:13:54	31:37:45	33:11:23

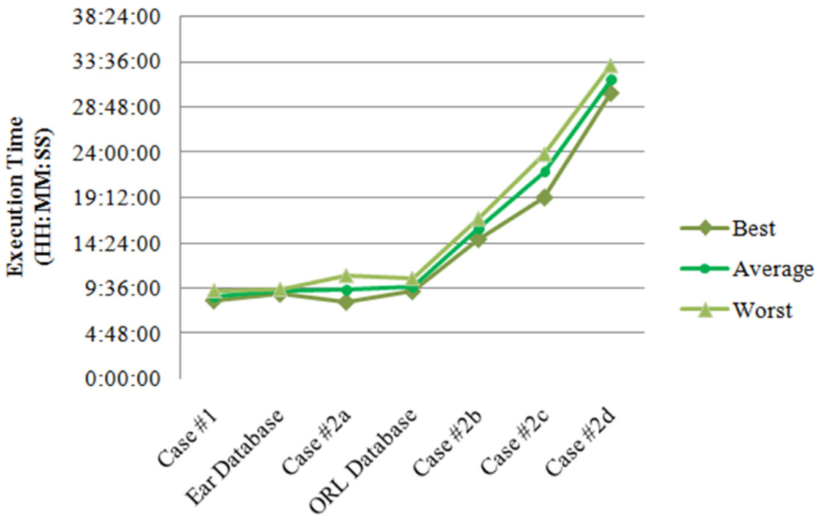

Fig. 13. Best, average and worst (execution times).

Table 11
Computer specifications.

Computers	1
RAM memory	12 Gb
Processor	Intel Core i7-2600
Processors Speed	3.4GHz
Operating system	Windows 7 Home Premium 64-Bit
Software	Matlab R2009b

In Table 12, the results obtained using the whole ORL Database are presented. In [29] the superior principal component (SPCA) analysis is used. In [14] the principal components analysis (PCA) is used. In [27] a self-organizing map (SOM) and a convolution network (CN) are used. In [2] eigenfaces and artificial neural networks are combined. In [8] the principal components analysis (PCA) and a linear discriminant analysis (LDA) are used. In [6] a new neural network approach is proposed based on conjugate gradient algorithms (cga) and a principal component analysis. In [7] a linear discriminant analysis (LDA) is used. In [32] a comparison of fuzzy edge detectors based on the image recognition rate as performance index calculated with neural networks is proposed. In [48] a scale invariant feature transform (SIFT) is used. This is a summary of the works considered in the comparison of results.

In Table 13, the results obtained for case #1 using the FERET Database are presented. In [32] a comparison of fuzzy edge detectors based on the image recognition rate as a performance index calculated with neural networks is proposed.

In Table 14, the results obtained for case #2 using the FERET Database are illustrated. In [48] a scale invariant feature transform (SIFT) is proposed.

In Table 15, the optimized results achieved for the Ear Database are presented. In [40], the granulation also used in this paper was originally proposed, but the optimization was performed using a single objective hierarchical genetic algorithm.

4. Statistical comparison of results

In this section, statistical comparisons are presented to verify that the proposed method is better than other works in the state of the art. Statistical t tests were performed to find the t values, because with sufficiently high values of t, the null

Table 12
Table of comparison (ORL database).

Method	Images for training									
	1	2	3	4	5	6	7	8	9	10
Mane et al. [29] (SPCA)	–	–	–	–	91.75	94.00	96.50	97.75	99.50	100
Duan et al. [14] (APCA)	–	–	–	–	88.50	93.75	95.00	–	–	–
Lawrence et al. [27] (SOM + CN)	70	83	88.2	92.9	96.2	–	–	–	–	–
Agarwal et al. [2] (Eigenfaces + ANNs)	–	–	–	–	–	98.04	–	–	–	–
Ch'ng et al. [8] (PCA + LDA)	–	–	–	–	96.5	–	–	–	–	–
Azami et al. [6] (CGA + PCA)	–	–	–	–	96.5	–	–	–	–	–
Bhattacharyyaet al. [7] (LDA)	–	–	–	–	–	–	–	–	92.5	–
Mendoza et al. [32] (MG + FIS2)	–	–	–	–	–	–	–	97.50	–	–
Wanget al. [48] (SIFT)	–	–	–	–	94	–	–	–	–	–
Proposed method	92.50	97.81	99.29	99.58	100	100	100	100	100	–

Table 13
Summary of comparisons for the FERET database in Case #1.

Method	Number of persons	Number of images	Images for training		
			1	2	3
Mendoza et al. [32] (Sobel + FIS2)	74	4	–	–	87.84
Proposed method	74	4	59.46	75.68	100

Table 14
Summary of comparisons for the FERET database in Case #2.

Method	Number of persons	Number of images	Images for training					
			1	2	3	4	5	6
Wanget al. [48] (SIFT)	50	7	–	–	–	86	–	–
Proposed method	50	7	69.33	83.20	91	92.67	97	100
Wanget al. [48] (SIFT)	100	7	–	–	–	79.7	–	–
Proposed method	100	7	64	75.60	84.25	91.67	96	99
Wanget al. [48] (SIFT)	150	7	–	–	–	79.1	–	–
Proposed method	150	7	56.89	71.20	83.50	83.50	92.67	98.67
Wanget al. [48] (SIFT)	200	7	–	–	–	75.7	–	–
Proposed method	200	7	46.92	67.20	78.25	83.50	92.75	97

Table 15
Table of comparison (Ear database, USTB).

Method	Images for training		
	1	2	3
Sanchez et al. [40]	–	98.05%	100%
Proposed method	81.82%	98.05%	100%

hypothesis (which states that there is no difference between the mean values obtained using the two compared methods) can be rejected. The values of the recognition rates previously presented were used to perform these statistical comparisons. The *t* tests were performed comparing with the other works that show more than one result in their work.

4.1. Comparison of results for the ORL database

The different values obtained in the *t*-test for the ORL Database between the proposed method versus [29,14,2,32,27,8,6] and the non-optimized trainings are shown in Table 16. The t-values are respectively 2.54, 3.76, 14.45, 4.05, 2.19, 8.53, 19.13 and 13.58, which means that there is sufficient statistical evidence to say that the face recognition results are improved using the proposed method in this paper.

4.2. Results comparison for case #1

The different values obtained in the *t*-test for the FERET Database (Case #1) between the proposed method versus [32] and the non-optimized trainings are presented in Table 17. The t-values are 9.52 and 8.93, which means that there is sufficient statistical evidence to say that the face results are improved using the proposed method.

Table 16
Values of the ORL database.

Test	N	Mean	Standard deviation	Error standard deviation of the mean	Estimated difference	t value	P value	Degree of freedom
Proposed method	6	99.917	0.204	0.083	–	–	–	–
Mane et al. [29] (SPCA)	6	96.58	3.21	1.3	3.33	2.54	0.05208786	5
Duan et al. [14] (APCA)	3	92.42	3.45	2.0	7.50	3.76	0.06392092	2
Agarwal et al. [2] (Eigenfaces + ANNs)	6	97.264	0.401	0.16	2.653	14.45	1.811E-06	7
Mendoza et al. [32] (FIS2)	4	94.69	2.58	1.3	5.23	4.05	0.02711166	3
Proposed method	25	96.32	4.87	0.97	–	–	–	–
Lawrence et al. [27] (SOM + CN)	5	86.1	10.3	4.6	10.26	2.19	0.09397884	4
Proposed method	5	99.9	0.224	0.10	–	–	–	–
Ch'ng et al. [8] (PCA + LDA)	4	94.75	1.19	0.60	5.150	8.53	0.00338017	3
Azami et al. [6] (CGA + PCA)	5	95.910	0.409	0.18	3.990	19.13	1.3186E-06	6
Proposed method (Optimized)	26	96.47	4.82	0.95	–	–	–	–
Non-optimized trainings	30	44.0	20.5	3.7	52.43	13.58	4.5586E-15	32

Table 17
Values of Case #1.

Test	N	Mean	Standard deviation	Error standard deviation of the mean	Estimated difference	t value	P-value	Degree of freedom
Proposed method	5	99.730	0.604	0.27	–	–	–	–
Mendoza et al. [32] (FIS2)	4	83.45	3.38	1.7	16.28	9.52	0.00246002	3
Proposed method	15	76.5	18.3	4.7	–	–	–	–
Non-optimized trainings	30	25.1	18.0	3.3	51.37	8.93	1.1035E-09	27

Table 18
Values of Case #2.

Test	N	Mean	Standard deviation	Error standard deviation of the mean	Estimated difference	t value	P value	Degree of freedom
Proposed method	20	88.35	3.80	0.85	–	–	–	–
Wang et al. [48] (SIFT)	4	80.13	4.29	2.1	8.22	3.56	0.02356	5
Proposed method	30	87.4	11.3	2.1	–	–	–	–
Non-optimized trainings	30	29.3	13.8	2.5	58.16	17.88	8.1515E-25	55
Proposed method	30	83.50	13.0	2.4	–	–	–	
Non-optimized trainings	30	30	15.9	2.9	53.46	14.24	2.9288E-20	55
Proposed method	28	80.2	14.9	2.8	–	–	–	–
Non-optimized Trainings	30	26.1	11.1	2.0	54.14	15.62	7.1767E-21	49
Proposed method	30	75.8	17.2	3.1	–	–	–	–
Non-optimized trainings	30	30.5	16.9	3.1	45.32	10.29	1.0693E-14	57

Table 19
Values of the USTB database (Using 2 images).

Test	N	Mean	Standard deviation	Error standard deviation of the mean	Estimated difference	t value	P value	Degree of freedom
Proposed method	7	96.66	0.694	0.26	1.845	2.71	0.01042	34
Sanchez et al. [40]	30	94.82	3.45	0.63				

4.3. Results comparison case #2

The different values obtained in the *t*-tests for the FERET Database (Case #2) between the proposed method versus [48] and the non-optimized trainings are the ones shown in Table 18. The t-values are 3.56, 17.88, 14.24, 15.62 and 10.29, which means that there is sufficient evidence to say that the face results are improved using the proposed method.

4.4. Results comparison for the ear database (USTB)

The values obtained in the *t*-test for the Ear Database (USTB) between the proposed method versus the optimization presented in [40] using 2 images per person (***nod***) are presented in Table 19. In this case, when 3 images are used, the means are 100% and 99.69% for the proposed method and the results achieved in [40] respectively and the t value is of 2.71.

5. Conclusions

In this paper, a new method for modular granular neural network (MGNN) optimization using a multi-objective approach is proposed. The proposed multi-objective hierarchical genetic algorithm (MOHGA) is based on a micro-genetic algorithm because the optimization with a small population is always a good option with applications that use large size data sets as is the case in human recognition. The proposed MOHGA performs the optimization of MGNN architectures optimizing some parameters, such as: the number of sub granules (modules), number of data for the training phase, goal error, learning algorithm, number of hidden layers and their respective number of neurons. In this case, two fitness functions are used; the minimization of the number of data for the training phase and the minimization of the recognition error. The multi-objective algorithms have many advantages, and one can be the search of multiple solutions concurrently in a single evolution. In this case, when these are applied to human recognition, then multiple architectures of modular granular neural networks using different sizes for the data sets for the training phase are obtained. In all the presented cases to perform the experiments, better results are obtained when the proposed method is used; this is because the MGNNs have the ability to learn complex data, the efficiency of the information granulation and also, the optimization of each architecture design depending on the database and the data for the training phase using the proposed MOHGA. We conclude that the MGNNs are a good choice to perform tasks such as pattern recognition, but their architecture must be optimally established to obtain better results because the methods, such as the trial and error or randomly establishing the architecture can be very tedious and do not guarantee that we achieve the optimal architecture. For this reason, when the MGNN and the MOHGA are used together they allow obtaining optimal architectures and better results than other techniques. To select the best solution of the Pareto front for each case, the correct parameters are established; the best solution is one that has a recognition rate higher than 95%, and with the smallest possible data set for training. This is because in the future, a system for human recognition using different biometric measures is going to be developed and this recognition is enough to obtain good results. As other possible future work, other objective functions can be considered (perhaps the minimization of the number of subgranules) and/or the use of sub-populations in the MOHGA.

References

[1] A. Abraham, L. Jain, R. Goldberg, Evolutionary Multiobjective Optimization, Springer, 2005.
[2] M. Agarwal, N. Jain, M. Kumar, H. Agrawal, Face recognition using eigen faces and artificial neural network, Int. J. Comput. Theory Eng. 2 (4) (2010) 624–629.
[3] M. Ali, P. Siarry, M. Pant, An efficient Differential Evolution based algorithm for solving multi-objective optimization problems, Eur. J. Oper. Res. 217 (2) (2012) 404–416.
[4] J. Anitha, R. Das, M.K. Pradhan, Multi-objective optimization of electrical discharge machining processes using artificial neural network, Jordan J. Mech. Ind. Eng. 1 (1) (2016) 11–18.
[5] AT&T Laboratories Cambridge, the ORL database of faces, Found on the Web page: https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html (Accessed on April 20, 2014).
[6] H. Azami, M. Malekzadeh, S. Sanei, A new neural network approach for face recognition based on conjugate gradient algorithms and principal component analysis, J. Math. Comput. Sci. 6 (2013) 166–175.
[7] S.K. Bhattacharyya, K. Rahul, Face recognition by linear discriminant analysis, Int. J. Commun. Netw. Secur. 2 (2) (2013) 31–35.
[8] S.I. Ch'ng, K.P. Seng, L. Ang, Modular dynamic RBF neural network for face recognition, in: Open Systems (ICOS), 2012 IEEE Conference on, 2012, pp. 1–6.
[9] C.A. Coello, G. Toscano Pulido, in: A Micro-Genetic Algorithm For Multiobjective Optimization, EMO, Zurich, Switzerland, 2001, pp. 126–140. 2001.
[10] C.A. CoelloCoello, G. Toscano Pulido, Multiobjective optimization using a micro-genetic algorithm, in: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001), San Francisco, USA, 2001, pp. 274–282.
[11] C.A. CoelloCoello, G. Toscano Pulido, Multiobjective structural optimization using a micro-genetic algorithm, Struct. Multidiscipl. Optim. 30 (5) (2005) 388–403.
[12] D.W. Corne, J.D. Knowles, M.J. Oates, The Pareto envelope-based selection algorithm for multiobjective optimization, in: Proceedings of the Parallel Problem Solving from Nature VI Conference, 2000, pp. 839–848.
[13] M. Dorigo, Optimization, learning and natural algorithms PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy, 1992.
[14] H. Duan, R. Yan, K. Lin, Research on face recognition based on PCA, in: International Seminar on Future Information Technology and Management Engineering, Leicestershire, United Kingdom, 2008, pp. 29–32.
[15] C.M. Fonseca, P.J. Fleming, Genetic algorithms for multiobjective optimization: formulation, discussion and generalization, in: Proceedings of the Fifth International Conference on Genetic Algorithms, 1993, pp. 416–423.
[16] J.C. Fuentes Cabrera, C.A. Coello Coello, Micro-MOPSO: A multi-objective particle swarm optimizer that uses a very small population size, Multi-Objective Swarm Intell. Syst. (2010) 83–104.
[17] D.E. Goldberg, Genetic Algorithms in Search Optimization and Machine Learning, Addison-Wesley, 1989.
[18] A. Gurpreet Kaur, A review on ear based biometric identification system, Int. J. Adv. Res. Comput. Sci. Softw. Eng. 6 (3) (2016) 548–552.
[19] J. Horn, N. Nafpliotis, D.E. Goldberg, A niched Pareto genetic algorithm for multiobjective optimization, in: Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 1, 1994, pp. 82–87.
[20] S.T. Hsieh, S.Y. Chiu, S.J. Yen, An improved multi-objective genetic algorithm for solving multi-objective problems, Appl. Math. Inf. Sci. 7 (5) (2013) 1933–1941.
[21] A.K. Jain, K. Nandakumar, A. Ross, 50 years of biometric research: accomplishments, challenges, and opportunities, Pattern Recognit. Lett. 79 (2016) 80–105.
[22] A.K. Jain, S. Pankanti, S. Prabhakar, L. Hong, A. Ross, in: Biometrics: A Grand Challenge, ICPR, Cambridge, England, UK, 2004, pp. 935–942. (2)2004.
[23] A.K. Jain, A. Ross, S. Prabhakar, An introduction to biometric recognition, IEEE Trans. Circuits Syst. Video Techn. 14 (1) (2014) 4–20.
[24] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proceedings of the 1995 IEEE International Conference on Neural Networks, Piscataway, New Jersey, 1995, pp. 1942–1948.
[25] K. Krishnakumar, Micro-genetic algorithms for stationary and non-stationary function optimization, SPIE Proc.: Intell. Control Adaptive Syst. 1196 (1989) 289–296.
[26] A. Lančinskas, P. Martinez Ortigosa, J. Žilinskas, Multi-objective single agent stochastic search in non-dominated sorting genetic algorithm, Nonlinear Anal.: Modell. Control 18 (3) (2013) 293–313.
[27] S. Lawrence, C.L. Giles, A.C. Tsoi, A.D. Back, Face recognition: a convolutional neural network approach, IEEE Trans. on Neural Netw. 8 (1997) 98–113.

[28] K.F. Man, K.S. Tang, S. Kwong, Genetic Algorithms: Concepts and Designs, Springer, 1999.
[29] A.V. Mane, R.R. Manza, K.V. Kale, Human face recognition using superior principal component analysis (SPCA), Int. J. Comput. Theory Eng. 2 (5) (2010) 688–691.
[30] S. Mane, S. Sonawani, S. Sakhare, Hybrid multi-objective optimization approach for neural network classification using local search, Innovations Comput. Sci. Eng. (2016) 171–179.
[31] P. Melin, D. Sánchez, O. Castillo, Genetic optimization of modular neural networks with fuzzy response integration for human recognition, Inf. Sci. 197 (2012) 1–19.
[32] O. Mendoza, P. Melin, O. Castillo, J.R. Castro, Comparison of fuzzy edge detectors based on the image recognition rate as performance index calculated with neural networks, in: Soft Computing for Recognition Based On Biometrics 2010, Springer, 2010, pp. 389–399.
[33] W. Mohamed, M. Heshmat, M. Girgis, S. Elaw, A new method for face recognition Using variance estimation and feature extraction, Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS) 2 (2) (2013) 134–141.
[34] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss, The FERET evaluation methodology for face-recognition algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 22 (10) (2002) 1090–1104.
[35] D. Pirale, M. Nirgude, Biometric techniques using neural networks, Int. J. Adv. Res. Comput. Commun. Eng. 5 (4) (2016) 619–625.
[36] K.S. Priya, B. AnuMol, Iris recognition, Int. J. Emerg. Trends Sci. Technol. 3 (5) (2016) 758–761.
[37] P.C. Ribas, L. Yamamoto, H.L. Polli, L.V.R. Arruda, F. Neves- Jr, A micro-genetic algorithm for multi-objective scheduling of a real world pipeline network, Eng. Appl. Artif. Intell. 26 (1) (2013) 302–313.
[38] T.C. Sabareeswari, S. Lenty Stuwart, Identification of a person using multimodal biometric system, Int. J. Comput. Appl. 3 (9) (2010) 12–16 (0975–8887).
[39] Y. Saji, M.E. Riffi, B. Ahiod, Multi-objective ant colony optimization algorithm to solve a nurse scheduling problem, Int. J. Adv. Res. Comput. Sci. Soft. Eng. 3 (8) (2013) 311–320.
[40] D. Sánchez, P. Melin, Optimization of modular granular neural networks using hierarchical genetic algorithms for human recognition using the ear biometric measure, Eng. Appl. Artif. Intell. 27 (2014) 41–56.
[41] N. Srinivas, K. Deb, Multiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput. 2 (3) (1994) 221–248.
[42] R. Storn, K.V. Price, Minimizing the real functions of the ICEC'96 contest by differential evolution, in: International Conference on Evolutionary Computation, 1996, pp. 842–844.
[43] H. Turabieh, Comparison of NEAT and backpropagation neural network on breast cancer diagnosis, Int. J. Comput. Appl. 139 (8) (2016) 40–44.
[44] University of Science & Technology Beijing (USTB), Database Ear Recognition Laboratory. Found on the Web page: http://www1.ustb.edu.cn/resb/en/index.htm (Accessed: August 5, 2013).
[45] J.C. Vázquez, M. Lopez, P. Melin, Real time face identification using a neural network approach, in: Soft Computing for Recognition Based On Biometrics 2010, Springer, 2010, pp. 155–169.
[46] P.A. Viola, M.J. Jones, Rapid object detection using a boosted cascade of simple features, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1, 2001, pp. 511–518.
[47] P.A. Viola, M.J. Jones, Robust real-time face detection, Int. J. Comput. Vision 57 (2) (2004) 137–154.
[48] Y.Y. Wang, Z.M. Li, L. Wang, M. Wang, A scale invariant feature transform based method, J. Inf. Hiding Multimedia Signal Process. Ubiquitous Int. 4 (2) (2013) 73–89.
[49] X. Xu, S. Tian, ELM-RBF neural networks using micro-genetic algorithm for optimization, Int. J. Hybrid Inf. Technol. 9 (12) (2016) 27–36.
[50] X.S. Yang, A new metaheuristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Granada, Spain, 2010, pp. 65–74.
[51] X.S. Yang, Bat algorithm for multi-objective optimization, Int. J. Bio-Inspired Comput. 3 (5) (2011) 267–274.
[52] X.S. Yang, Firefly algorithms for multimodal optimization, in: O. Watanabe, T. Zeugmann (Eds.), Proc. 5th Symposium on Stochastic Algorithms, Foundations and Applications, 2009, pp. 169–178. Lecture Notes in Computer Science, 5792.
[53] X.S. Yang, Multiobjective firefly algorithm for continuous optimization, Eng. Comput. 29 (2) (2013) 175–184 April 2013.
[54] X.S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, Bristol, UK, 2008.