Use of Nature-inspired Meta-heuristics for Handwritten Digits Recognition

Salima Nebti[1], Abdellah Boukerram[1]
[1]Department of computer science
Ferhat Abbess University, Setif, 19000
Algeria
s.nebti@live.fr, boukerram@hotmail.com

ABSTRACT: *Character recognition is an important task in pattern analysis that aims to give significance to handwritten data without users' intervention. Although, an intensive research has been devoted to this problem, it remains a challenging task as humans need to interact with computer in the easiest way. This work attempts to incorporate some meta-heuristics as guidelines searching for the best solution of handwritten digits recognition problem namely the particle swarm optimizer and variations of the bees' algorithm. The bees' algorithm is a variant of evolutionary optimization that take inspiration from the foraging behavior of honey bees where individuals called bees are used to perform a neighborhood search in joint with a random search as an attempt to achieve a good balance between exploration and exploitation abilities. We show that this method can be adapted to handwritten characters recognition and can be effectively combined with a neural network classifier which results in a good quality on a wide range of real data against that of the k-nearest neighbour classifier and the back-propagation training algorithm.*

Keywords: Particle Swarm Optimization, The Bees Algorithm, The Artificial Bee Colony Optimization, The Multi-Layer Perceptron

Received: 11 September 2009, Revised 28 October, Accepted 8 November 2009

1. Introduction

Handwritten characters recognition has been paid a great attention of researchers for several decades and still an open field of research due to its great number of applications like automatic reading of bank checks, postal addresses and more especially for tablet PC applications…...

Handwritten characters recognition is the handwritten data transcription to numerical format to obtain its significance. Handwritten characters recognition systems can be distinguished into two types: the online recognition and off-line recognition. The first one treats the data coming from a sensor when the user writes, while the second one treats the obtained data from a handwritten document (scanned) [1]. Characters recognition is a difficult task due to the large variability of writing styles and it is more difficult on the level of text recognition where recognition passes by many processes that are: text segmentation into sentences, sentences into words and words could be segmented into letters. In all these processes, a contextual dictionary plays an essential role to correctly interpret the different found assumptions during recognition. Characters segmentation represents a major challenge for a handwritten recognition system mainly for numerical chains due to numbers overlapping or splitting into several fragments, also, there is not a contextual information that help recognition (I.e., any digit can follow any other). Characters segmentation is the process which finds an isolated segment that must be identified by significant features. The determination of these feature vectors known as Features extraction is a difficult task and constitutes an important step that improve the robustness of a recognition system. The feature vectors are supposed to be invariable for objects of the same class, i.e. they must be independent to translation, rotation and scaling.

After characters segmentation and features extraction a recognition step is used for characters classification. Many recognition methods can be found, the most used are neural networks thanks to their generalization capability. Despite, the amount of works done in character recognition, it is still required to define new methods to improve system robustness and generalization. Nature inspired computing is an emergent paradigm that has got a considerable interest. Swarm intelligence is a subfield of nature inspired computing more particularly the social behaviors of animals and their related disciplines. This work contributes in showing its suitability to solve characters recognition problem. We outline a new way for the design of decentralized cooperative strategies to deal with handwritten characters recognition through the use of particle swarm optimisation (PSO), the Bees' Algorithm (BA) and the Artificial Bee Colony optimization (ABC). These strategies show an emergent intelligent

behavior from cooperation of simple agents which is robust compared to intelligent individual agents [2].

In this work, solutions are considered as particles or honey bees, imitating the collective behaviour of birds flocking or honey bees when searching for food sources. This work is motivated by the social interactions of birds and bees in nature that are adapted here to find the promising handwritten models which deliver the minimum value for a given criterion function. Results are encouraging compared to the famous method: the K-Nearest Neighbour classifier.

The structure of this paper is as follows: section 2 summarizes the fundamental steps of handwritten digits recognition within a clustering scheme; sections 3, 4 and 5 review the used strategies already mentioned; section 6 provides and discusses the obtained results; finally, a summary of this paper and envisaged developments are given.

2. Problem Formulation

The proposed approach uses two different characteristic features: the Hu moment features and pixel based density zoning then adapts particle swarm optimisation or the bees' meta-heuristics so that to define a new dynamic of characters recognition in a way to reduce the following sum of squared error.

$$SSE = \sum_{i=1}^{M} \sum_{x \in C_i}^{N} \left\| x - \overline{C}_i \right\|^2 \tag{1}$$

Where M: is the number of feature vectors of the existing characters as models

N: is the number of feature vectors of the character to recognize

$\overline{C}_i$: is the feature vector of the character to recognize

SSE represents the sum-of-squared-error criterion which is generally used for clustering. The minimization of this criterion results into clusters having minimized intra-cluster variability and consequently a maximized inter-cluster variability.The algorithm divides the features' vectors into a predetermined number of characters while optimising the Euclidean distance between the training models and the character to recognise. Finally each character is assigned to the closest class of the model having the nearest centre.

The proposed approach can be summarized in the following diagram:

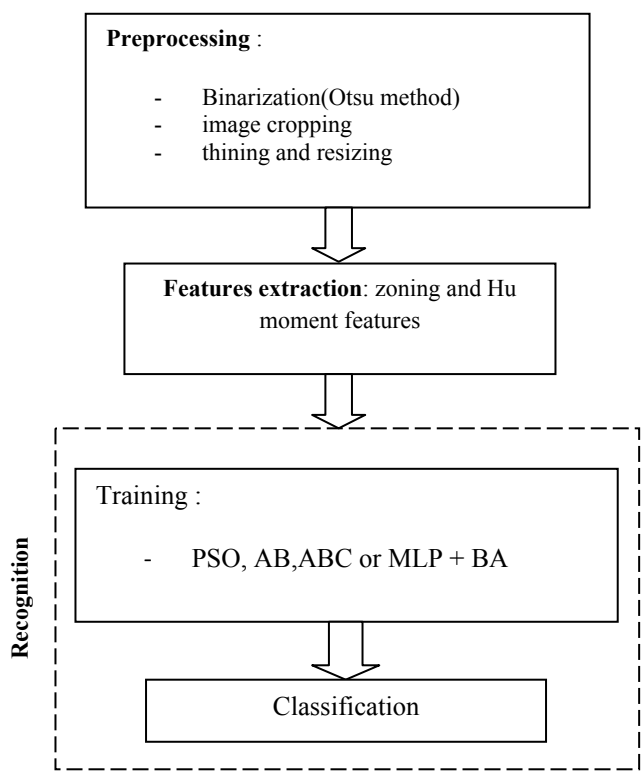

Figure 1. Structure of the recognition system

2.1 The Preprocessing Step

We perform the following preprocessing steps:

- Binarization using Otsu method
- Crop each character from its image by eliminating the blank spaces
- Skeletization (thinning) using morphological operations

2.2 Features Extraction Step

We have employed pixel based density zoning and the 7 invariant Hu moments.

Zoning

Zoning is based on imposing a grid N*M on the character image and thus dividing the character into a number of zones where characteristic features are extracted from each zone to form the feature vector. The goal of zoning is to obtain local characteristics instead of the global characteristics [4], [3].

In fact, Zoning is not a method of features extraction but an auxiliary process used in combination with other feature extractors. In this technique, the treated image is subdivided into zones like a rectangular grid. Then a certain method of characteristics extraction is applied separately to each zone. The final feature vector is formed by concatenation of the features of these zones [5].

Calculation of pixels density in each zone:

Density: represents the proportion of the number of black pixels forming the character on the total size of each zone [6].

The invariant Hu moments:

The following is a brief description of the invariant Hu moments [8]:

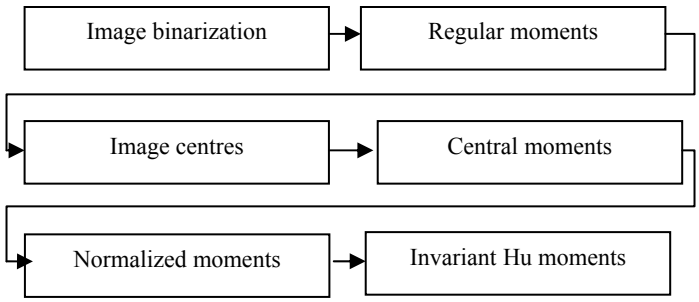

Figure 2. Evaluation of Hu moments

Hu introduced 7 nonlinear moments which are invariant to translations, scaling and rotations. To compute these features; we follow the steps bellow [7].

- image binarization
- regular moments are then evaluated under the format:

$$m_{pq} = \iint_{-\infty}^{+\infty} x^p y^p f(x,y)\, dx\, dy\, , p,q = 0,1,2, \dots \dots .$$

Where m pq is the moment of order (p+q) of image f(x, y).

- The central moments of f(x, y) are defined as follows:

$$\mu_{pq} = \iint_{-\infty}^{+\infty} (x - x_c)^p \ (y - y_c)^q \ f(x,y)\, dx\, dy, \qquad p,q = 0,1,2, \dots \dots$$

where $x_c = \frac{m_{10}}{m_{00}}$ et $y_c = \frac{m_{01}}{m_{00}}$ are image centre.

For digital images, the integrals are transformed into sums and thus give for normal and central moments respectively:

$$m_{pq} = \sum_{x}\sum_{y} x^p y^p f(x,y) \ \ ,p,q = 0,1,2,\dots\dots.$$

$$\mu_{pq} = \sum_{x}\sum_{y} (x - x_c)^p (y - y_c)^q f(x,y), \ \ p,q = 0,1,2,\dots\dots..$$

- the normalized central moments can be defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{y}}, y = \frac{(p+q+2)}{2}, p+q = 2,3,\dots\dots$$

- The 7 moments of Hu are defined in the following equations:

$$\emptyset_1 = \eta_{20} + \eta_{02}$$

$$\emptyset_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\emptyset_3 = (\eta_{30} - 3\,\eta_{12})^2 + (3\,\eta_{21} - \eta_{03})^2$$

$$\emptyset_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\emptyset_5 = (\eta_{30} - 3\eta_{12})\,(\eta_{30} + \eta_{12})\,[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\,] + (3\eta_{21} - \eta_{03})\,(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\,]$$

$$\emptyset_6 = (\eta_{20} - \eta_{02})\,[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\,] + 4\,\eta_{11}(\eta_{30} + \eta_{12})\,(\eta_{21} + \eta_{03})$$

$$\emptyset_7 = (3\eta_{21} - \eta_{03})\,(\eta_{30} + \eta_{12})\,[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\,] + (3\eta_{21} - \eta_{30})\,(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\,]$$

After determination of feature vectors it is possible to assign them to known classes that represent all the possible character models (here ten classes: from 0 to 9).

In our design, each solution either a particle or a honey bee is represented as a set of handwritten characters defined by their feature vectors (here the Hu invariant moments and pixel based density zoning). To solve the problem mentioned above using PSO or the Bees meta-heuristics, we must first decide about an appropriate representation. In another way, we should define what is a particle or a honey bee in the context of the tackled problem?

Hence, the following representation is used:

$\overline{C}_1$	$\overline{C}_2$	.....	$\overline{C}_M$

Figure 3. Structure of a potential solution

***Particle or a honey bee*:** represents a potential solution that is a set of character models used for classification. Each character image either the character being recognized or the character models is expressed in terms of Hu moment vectors pixel based density zoning.

Where C_1,C_2,...C_3 are the feature vectors associated to the selected characters from the used database here the MNIST database.

3. Classification

The proposed approach can be seen as a reactive multi-agent system composed of particles or bees agents, which behave according to the described PSO or Bees dynamics searching for the most accurate classification. The system is based on the following principle: agents use their own results and the ones of their neighboring agents to achieve in a collective way the recognition task. The key idea is thus: changing the feature vectors based on particle swarm optimization or the honey bees' dynamics to achieve better exploration of the search space. As a result, the features that optimizes the sum of squared error function is said to be the target characters.

The developed strategy can be summarized as follows. Particles or bees are initialized by feature vectors of randomly selected characters from the used database. Then the whole population undergoes evolution through an iterative process. This iterative process is the particle swarm optimization method or the bees' algorithms.

The following is a description of these algorithms for digits recognition.

Particle swarm optimizer based approach:

Particle Swarm Optimization (PSO) is a stochastic optimization method developed by Kennedy and Eberhart [9] in 1995 for non-linear functions. This method is originally derived from the social flocking behavior of birds during Reynold simulations of grouped birds' and fish benches [10]. This method is based on cooperation between simple individuals named as agents. These agents have little memory capacities in the sense that each agent remembers its personal best visited position and the best one found in its neighborhood then moves in a weighed trajectory among these two positions and its velocity. PSO is a simple and effective algorithm which starts with a number of particles positioned in a random way within the search space and initialized with random velocities (or speediness).

For a number of iterations or until a condition is met particles move according to the following rules:

$$V_{k+1} = wV_k + R_1 * C_1(P_i - X_k) + R_2 * C_2(P_g - X_k) \quad (1)$$
$$X_{k+1} = X_k + V_{k+1}.$$

Algorithm 1: The particle swarm optimizer for characters recognition

Transform the set of characters being recognized into a vector of Hu moment features or pixel based density zoning.
Initialize a population of particles where each particle has a size equal to the number of characters being recognized and each particle is initialized with randomly chosen characters from the MNIST database

For each character do // processing characters within a particle
{**For** each particle do
{While a number of iterations is not expired do
{Evaluate the Euclidean distance between the feature vectors of this character and the related character being recognized.
Update P_{best} and G_{best}
Update position and velocity according to eq.(3),
}}}
Use the best found features as centres to minimize the Sum of Squared Error (SSE) of the selected training characters.
The labels of characters that give the minimum SSE constitute the target solution.

Where: V is the particle velocity, X is its position, w: the inertia weight, R_1 and R_2 are random values between 0 and 1, C_1 and C_2 are respectively the personal and the collective factors, P_i and P_g are respectively the personal position and the best one in the defined neighborhood

The Bees Algorithm based approach:

The Bees Algorithm is a population based algorithm imitating the foraging behavior of honey bees. This algorithm follows a spatial neighborhood based search strategy which starts with randomly locating scout bees into the search space. To find an optimal solution, a quantity of the scout bees is specified as representative bees (sites) searching in the neighborhood of the best solutions and the remaining bees scout randomly for better exploration [11],[12].

In the first step, the fitnesses of the visited sites are computed. The sites that have the highest fitnesses are chosen for neighborhood search. Then, recruitment of bees around sites and elite sites is conducted. The recruitment is a neighborhood search; for each selected site, a patch size is defined. A number of bees that belong to the patch are selected for fitness evaluation. More bees are selected from the neighborhood of the elite patches. Selection can be directly according to the fitnesses of the related sites or alternatively, using their fitness values to determine the

Algorithm 2: The bees optimizer for characters recognition

Transform the set of characters being recognized into vectors of Hu moments or pixel based density zoning.
Initialise a population of bees having the same structure of particles in the first algorithm.

While a number of iterations is not expired do **{**
{**For** each character do // processing characters within a honey bee
{**For** each scout bee do
{**For** each elite bee do //the neighbouring bees of the current scout bee

{Evaluate the elite bees' fitnesses which are the Euclidean distances between
the feature vectors of training characters and the related character being recognized replace the current
scout bee by the current elite bee if it is better }}
{**For** each site bee do //a predefined number of neighbouring bees to
the current scout bee

{Evaluate the site bees' fitnesses which are the Euclidean distances
between the feature vectors of training characters and the related character
being recognized replace the current scout bee by the current site bee if it is better }}
Assign the remaining bees in a random way }}}
Use the best found features as centres to minimize the Sum of Squared Error (SSE) of the selected training characters.

The labels of the characters whose the SSE is minimum constitute the target solution.

probability of the bees being selected. Then, the best found bees into the patches will replace their related sites to form the next population. The remaining bees are randomly placed scouting for new and perhaps better solutions. This differential recruitment is a key operation of the Bees Algorithm. These steps are repeated until a stopping criterion is met [11].

The Artificial Bee Colony Optimization (ABC)

The ABC algorithm is a recent meta-heuristic inspired from the intelligent behavior of honey bees when searching for food sources [13]. The population in ABC algorithm is constituted of three sub-populations: employed bees, onlookers and scouts. The employed bees search for food sources and then return to dance on the hive area. The onlookers choose the food sources of employed bees depending on their dances. Scouts are the employed bees that abandoned their food sources to search new and perhaps better food sources [14].

In ABC algorithm, food sources are possible solutions to an optimization problem and their nectar amounts correspond to fitnesses of the associated solutions. This algorithm starts with randomly generated population of solutions. An employed bee generates a new food source (solution) based on its initial food source. If the new solution is better than the old ones then the employed bee guards it as its food source. Each onlooker chooses a food source based on the dance of employed bees. Then each of them creates a new food source and checks its fitness to guard or to forget it. The food sources whose fitness remains unchanged for a number of iterations are randomly replaced by artificial scout bees [13].

Algorithm 3: The artificial bee colony optimizer for characters recognition

Transform the set of characters being recognized into vectors of Hu moments or pixel based density zoning.
Initialize a population of Employed bees having the same structure of particles in the first algorithm // (the number of food sources is equal to the number of employed bees).

While a number of iterations is not expired do{
{**For** each character do // processing characters within a honey bee // Employed phase
{**For** each Employed bee do
{Determine a neighbor to its food source //the neighbor bee of the employed bee

Evaluate the fitness of this neighbor which is the Euclidean distance between the feature vectors of training characters and the related character being recognized Replace the current employed bee by its neighbor if it is better }}

// Onlooker phase

{**For** Each onlooker bee do

{Choose one of the food sources depending on the fitnesses employed bees.
Determine a neighbor to the onlooker bee

Evaluate the neighbour bee fitness which is the Euclidean distances between the feature vectors of training characters and the related character being recognized

Replace the current onlooker by its neighbor if it is better }}

initialize the food sources which were not improved for a number of iterations in a random way}}
Use the best found features as centres to minimize the Sum of Squared Error (SSE) of the selected training characters.

The labels of characters that give the minimum SSE constitute
the target solution.

The solution based on Multi-Layer Perceptrons:
The multi layer perceptron (MLP) is a supervised neural network widely used for pattern recognition, particularly for characters recognition, An MLP consists of an input layer whose size is equal to the input data size, one or more hidden layers whose size is determined by experiment and the output layer whose size is equal to the number of the target classes. However, MLPs neural networks present several limitations concerning how to find the suitable architecture (i.e. the number of layers and neurons), to choose the size and adequate quality of training examples, to precise good initial values for weights and algorithm parameters [15]. The MLP networks are also very sensitive to initialization; the training rate influences in a significant way the network performance [16],[17]. A large variety of methods have been proposed to cure these problems; among them the dynamic addition of neurons; a neuron is added each time the error is stagnated on undesirable value. The momentum is an effective alternative to accelerate the training and also to escape local minima [16]. Many other modifications of the retro-propagation algorithm were proposed, in order to accelerate the multi-layer perceptron training and thus its convergence speed. Methods of regularization or minimization such as Newton, or Quasi-*Newton*, were considered.

In this work we try to improve digits recognition quality by a serial combination of two different classifiers: Ten MLPs neural networks and the bees' algorithm based classifier. In so doing, we have considered ten MLPs to recognize the ten digits (0 ..9). the output of each MLP is 1 for the considered digit class and 0 for all the other digits. We have considered this scheme to find the digits which were not recognized during the test step. (The unrecognized digits are those not classified by any one of the ten networks).

The following is a description of this method.

Algorithm: The bees algorithm with MLP trained by back-propagation for digits recognition

1. Transform the set of characters being recognized into feature vectors.
2. Apply back-propagation to train the ten MLPs
3. Test the networks and determine the characters which were not recognized
4. Send these characters as inputs to the described above BA classifier.

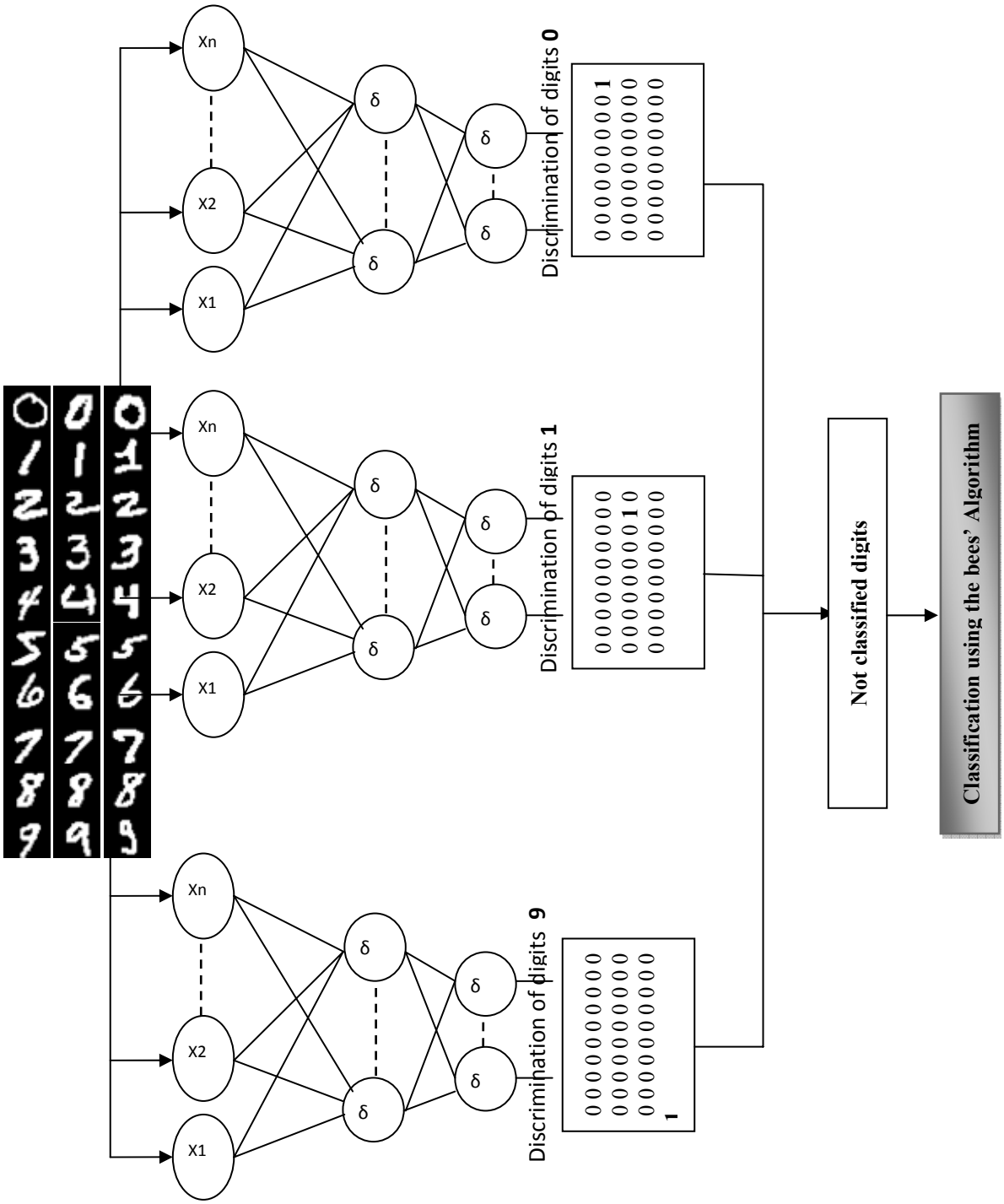

Figure 4. MLP cooperation for digits recognition

4. Results and Discussion

Different kinds of handwritten digits have been used like handwritten digits obtained from the MNIST database and scanned samples of handwritten digits.

The initial parameter settings during experiments were as summarized in the table below:

We have adapted the ABC algorithm with some modifications; where the BA neighborhood was used rather than a random selection of possible solutions as neighbors. Also the used fitness is the Sum of Squared Error between the characters being recognized and the training characters without any changes as in the ABC algorithm, so the objective function is minimized rather than maximized. The probability of the selected food sources by onlookers is: (the-fitness-of-an-employedbee / Sum-of-all-employed-bees-fitnesses). With these changes, the algorithm is more robust when applied to handwritten digits recognition.

The used classifier	Parameters setting
For all these algorithms : The number of training characters : 8000, The number of testing characters: 2000 which were selected from the MNIST database.	
PSO based Classifier	NumberOfparticles:200 The inertia weight : A decreasing inertia weight from 0.78 to 0.1 The cognitive and social factors :1.49
The BA based Classifier	Number of bees: 200, Number of sites: 40, Number of elite sites:20 Number of bees around sites:50 Number of bees around elite sites: 80 Size of neighborhood: 0.0234
The ABC based Classifier	Number of employed bees:100 Number of onlooker bees:100 Number of iterations for scouting: 20
The combined MLP and BA classifier	For each MLP: The number of Hidden neurons: 32, The number of Output neurons: 10 Number of epochs:5000, a sigmoid activation function

Table 1. Initial parameter setting

In table 2, we summarize the obtained results after 10 runs for each algorithm.

The used classifier	The recognition rate	
	Zoning features	Hu moments
PSO classifier	80.428 %	92.2%
ABC classifier	80 %	84.49%
AB classier	92.857 %	99.8%
K-nn	90%	98%
MLP (back-propagation)	80.01%	
The combined MLP and BA	99.82%	

Table 2. Comparative results

Table II presents the values of the recognition rate in 10 runs for all the conducted experiments. It can be seen that the BA based training approach gives the best results; PSO algorithm gives better results than ABC when dealing with handwritten digits recognition. Also, it has been found that the Hu moment feature extractor contributes to a better classification than the zoning feature extractor. In the second alternative the combined back-propagation with the BA gives the highest results compared to the famous back-propagation algorithm because the misclassified digits using the MLP will be then correctly classified using a good classifier such as the bees' algorithm.

Also, it is important to say that we have obtained a recognition rate of 100% on 7500 training digits and 2000 testing digits when ambiguous digits have been eliminated.

The graphs bellow show the behavior of the proposed classifiers i.e. the minimization of the sum squared error (SSE) on the training data during iterations when the Hu moment features are considered.

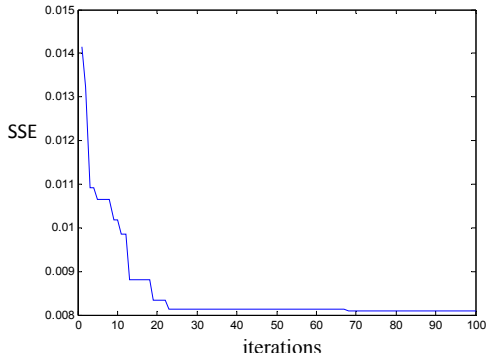

Figure 5. The ABC classifier behavior: Recognition Rate 80.9%

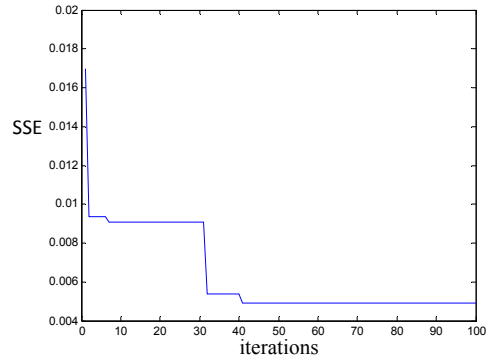

Figure 6. The BA classifier behavior: Recognition Rate 99.85%

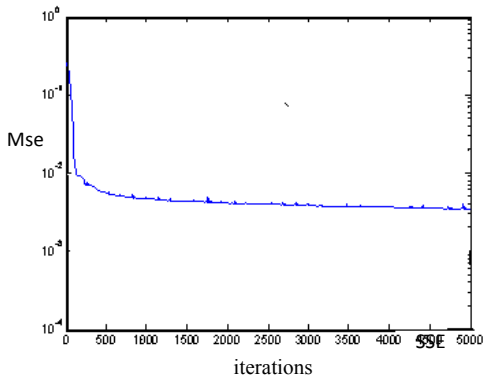

Figure 7. The combined Bp with BA classifier behavior: Recognition Rate 99.87%

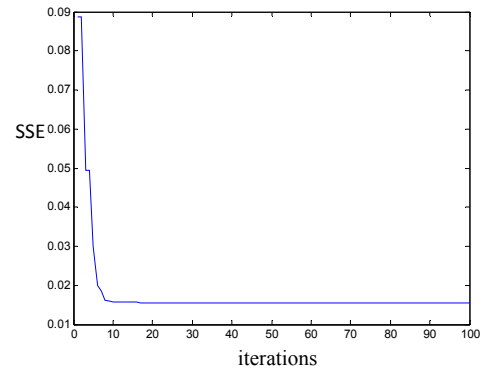

Figure 8. The PSO classifier behavior: Recognition Rate 90.65%

5. Conclusion

The focus of this paper is on the use of different population-based optimization algorithms searching for the best solution of handwritten digits recognition problem. To achieve this aim, these evolutionary optimizers have been adapted as statistical classifiers. Additionally, a combination of Multi-Layer Perceptron neural network (MLP) and the bees' algorithm has been proposed to enhance recognition accuracy. Furthermore, a new and simple cooperation of ten MLPs has been designed in such a way that allows determination of the digits which were not correctly classified. The quality of solutions has been estimated by the recognition rate on a large amount of testing data. Comparative study shows that our proposed strategies give promising results in terms of accuracy against that of k-nearest neighbor classifier and the well known back-propagation training algorithm. The use of the BA often gives the best results compared to PSO, ABC and the famous K nearest neighbor (K-nn) classifier. Also, it is found that the combined BA with Back-propagation for training a MLP gives the highest results in terms of accuracy and speediness due to hybridization with the potential of neural networks. We have found that the neural approach is the best, it takes an enormous time for training but once the training step finishes, the best weights are saved for a very quick recognition time, in contrast to the other presented approaches which also give good results but take a very long time for recognition which make their use impossible in real applications. As future works, we can easily extend this work to recognize the handwritten Latin or Arabic characters based on other structural and geometrical characteristic features. Also we can reduce the training time by selection of small digits set using these meta-heuristics as clustering methods before the training step. Also we guess that the use of these meta-heuristics for neural networks architecture adaptation or characters segmentation can give good quality results.

References

[1] Park, J. (1999). Hierarchical Character Recognition and its uses in handwritten word/phrase recognition, PHD thesis.

[2] Lhotská, L., Macaš, M., Strackeljan, J., Jantzen, J.(2005). Applied Nature Inspired System: Preliminary Investigation. *In*: European Symposium on Nature-inspired Smart Information Systems.

[3] Evangelos, Y. (2006). Performance Evaluation of feature extraction Algorithms for character Recognition. (Rathna Ramaswamy, Department Of Computer Science, University Of Nevada, Las Vegas).

[4] Vamvakas, G. Optical Character Recognition for Handwritten Characters , National Center for Scientific Research "Demokritos" Athens – Greece, Institute of Informatics and Telecommunications, http://www.iit.demokritos.gr/IIT_SS/Presentations/Off-Line%20Handwritten%20OCR.ppt

[5] Laaksonen, J. (1997). Subspace classifiers in recognition of handwritten digits, Ph.D. Thesis, Helsinki University of Technology.

[6] Parker, J.R. (1994). Practical computer vision using C. University of Calgary; Canada; P 40-75, 354-393.

[7] Sanjeev R., Kunte, R.D., Sudhaker,S. (2007). A simple and efficient optical character recognition system for basic symbols in printed Kannada text.

[8] Villegas-Cortez, J., Avile's-Cruz, C. (2005). Font recognition by invariant moment of global textures. *In*: Internat. Workshop on Very Low Bit-Rate Video-Coding.

[9] Kennedy. J., Eberhart, R. C. (1995). Particle Swarm Optimization, *In*: Proceedings of the IEEE International Conference on Neural Networks, v. 4, 1942 - 1948. IEEE Press.

[10] Reynolds, C.W. (1987). Flocks, herds and schools: A distributed behavioral model. ACM SIGGRAPH Computer Graphics, 21 (4) 25-34.

[11] Pham, D T., Ghanbarzadeh, A., Koç E., Otri.S., Rahim, S., Zaidi, M. (2006). The Bees Algorithm – A Novel Tool for Complex Optimisation Problems.Proceedings of IPROMS 2006 Conference, p.454-461.

[12] Pham. D T., Ghanbarzadeh, A, Koc E, Otri S, Rahim S and Zaidi M. (2005). The Bees Algorithm. Technical Note. Manufacturing Engineering Centre, Cardiff University, UK.

[13] Karaboga. D. (2005). An Idea Based On Honey Bee Swarm for Numerical Optimization, Technical ReportTR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

[14] Karaboga, D., Basturk, B. (2007). Artificial Bee Colony Algorithm on Training Artificial Neural Networks, Signal Processing and Communications Applications, 2007. SIU 2007, IEEE 15th. 11-13 June 2007, p. 1 – 4.

[15] Gerstner, W. (1988). Supervised learning for neural networks: a tutorial with JAVAexercises, Technical Report {http://diwww.epfl.ch/~gerstner/wg_pub.html}.

[16] Abraham, A. (2004). Meta-Learning Evolutionary Artificial Neural Networks, *Neurocomputing Journal*, Elsevier Science, Netherlands, V. 56c, p. 1-38.

[17] Aschenwald, J., Tappeiner, G., Tappeiner, U., Tasser, E., Holub, H. W. (2004). Statistical Aspects of Multilayer Perceptrons Under Data Limitations. Computational Statistics & Data Analysis, 46 (1) 173-188.