International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:2, No:5, 2008

Feature Subset Selection Using Ant Colony Optimization

Ahmed Al-Ani

***Abstract*—**Feature selection is an important step in many pattern classification problems. It is applied to select a subset of features, from a much larger set, such that the selected subset is sufficient to perform the classification task. Due to its importance, the problem of feature selection has been investigated by many researchers. In this paper, a novel feature subset search procedure that utilizes the Ant Colony Optimization (ACO) is presented. The ACO is a metaheuristic inspired by the behavior of real ants in their search for the shortest paths to food sources. It looks for optimal solutions by considering both local heuristics and previous knowledge. When applied to two different classification problems, the proposed algorithm achieved very promising results.

***Keywords*—**Ant Colony Optimization, ant systems, feature selection, pattern recognition.

I. Introduction

THE problem of feature selection has been widely investigating due to its importance to a number of disciplines such as pattern recognition and knowledge discovery. Feature selection allows the reduction of feature space, which is crucial in reducing the training time and improving the prediction accuracy. This is achieved by removing irrelevant, redundant, and noisy features (i.e., selecting the subset of features that can achieve the best performance in terms of accuracy and computational time).

As described in their paper, Blum and Langley [1] argued that most existing feature selection algorithms consist of the following four components:

- *Starting point in the feature space*. The search for feature subsets could start with (i) no features, (ii) all features, or (iii) random subset of features. In the first case, the search proceeds by adding features successively, while in the second case, features are successively removed. When starting with a random subset, features could be successively added/removed, or reproduced by a certain procedure.
- *Search procedure*. Ideally, the best subset of features can be found by evaluating all the possible subsets, which is known as exhaustive search. However, this becomes prohibitive as the number of features increases, where there are 2^N possible combinations for N features. Accordingly, several search procedures have been developed that are more practical to implement, but they are not guaranteed to find the optimal subset of features. These search procedures differ in their computational cost and the optimality of the subsets they find.
- *Evaluation function*. An important component of any feature selection method is the evaluation of feature subsets. Evaluation functions measure how good a specific subset can be in discriminating between classes, and can be divided into two main groups: filters and wrappers. Filters operate independently of any learning algorithm, where undesirable features are filtered out of the data before learning begins [2]. On the other hand, performance of classification algorithms is used to select features for wrapper methods [3, 4].
- *Criterion for stopping the search*. Feature selection methods must decide when to stop searching through the space of feature subsets. Some of the methods ask the user to predefine the number of selected features. Other methods are based on the evaluation function, like whether addition/deletion of any feature does not produce a better subset, or an optimal subset according to some evaluation strategy is obtained.

In this paper, we will mainly be concerned with the second component, which is the search procedure. In the next section, we give a brief description of some of the available search procedure algorithms and their limitations. An explanation of the Ant Colony Optimization (ACO) is presented in section three. Section four describes the proposed search procedure algorithm. Experimental results are presented in section five and a conclusion is given in section six.

II. The Available Search Procedures

A number of search procedure methods have been proposed in the literature. Some of the most famous ones are the stepwise, branch-and-bound, and Genetic Algorithms (GA).

The stepwise search adds/removes a single feature to/from the current subset [5]. It considers local changes to the current feature subset. Often, a local change is simply the addition or deletion of a single feature from the subset. The stepwise, which is also called the Sequential Forward Selection (SFS)/ Sequential Backward Selection (SBS) is probably the simplest search procedure and is generally sub-optimal and suffers from the so-called "nesting effect". It means that the features that were once selected/deleted cannot be later discarded/re-

A. Al-Ani is with the Faculty of Engineering, University of Technology, Sydney, GPO Box 123, Broadway, Australia (e-mail: ahmed@eng.uts.edu.au).

selected. To overcome this problem, Pudil et al. [6] proposed a method to flexibly add and remove features, which they called "floating search".

The branch and bound algorithm [7] requires monotonic evaluation functions and is based on discarding subsets that do not meet a specified bound. When the size of feature set is moderate, the branch and bound algorithm may find a practicable solution. However, this method becomes impracticable for feature selection problems involving a large number of features, especially because it may need to search the entire feasible region to find the optimal solution. Also, it may not be possible to use the branch and bound algorithm in wrapper methods because of the monotonic constraint of the evaluation function, where the classification accuracy is not guaranteed to increase by including more features.

Another search procedure is based on the Genetic Algorithm (GA), which is a combinatorial search technique based on both random and probabilistic measures. Subsets of features are evaluated using a fitness function and then combined via cross-over and mutation operators to produce the next generation of subsets [8]. The GA employ a population of competing solutions, evolved over time, to converge to an optimal solution. Effectively, the solution space is searched in parallel, which helps in avoiding local optima. A GA-based feature selection solution would typically be a fixed length binary string representing a feature subset, where the value of each position in the string represents the presence or absence of a particular feature. According to [9,10], the GA was able to achieve better performance than other conventional methods.

We propose in this paper a subset search procedure that utilizes the ant colony optimization algorithm and aims at achieving similar or better results than GA-based feature selection.

III. Ant Colony Optimization

In real ant colonies, a pheromone, which is an odorous substance, is used as an indirect communication medium. When a source of food is found, ants lay some pheromone to mark the path. The quantity of the laid pheromone depends upon the distance, quantity and quality of the food source. While an isolated ant that moves at random detects a laid pheromone, it is very likely that it will decide to follow its path. This ant will itself lay a certain amount of pheromone, and hence enforce the pheromone trail of that specific path. Accordingly, the path that has been used by more ants will be more attractive to follow. In other words, the probability with which an ant chooses a path increases with the number of ants that previously chose that path. This process is hence characterized by a positive feedback loop [11].

Dorigo et. al. [12] adopted this concept and proposed an artificial colony of ants algorithm, which was called the Ant Colony Optimization (ACO) metaheuristic, to solve hard combinatorial optimization problems. The ACO was originally applied to solve the classical traveling salesman problem [11], where it was shown to be an effective tool in finding good solutions. The ACO has also been successfully applied to other optimization problems including data mining, telecommunications networks, vehicle routing, etc [13, 14, 15].

In order to solve an optimization problem, a number of artificial ants are used to iteratively construct solutions. In each iteration, an ant would deposit a certain amount of pheromone proportional to the quality of the solution. At each step, every ant computes a set of feasible expansions to its current partial solution and selects one of these depending on two factors: local heuristics and prior knowledge.

For the classical Traveling Salesman Problem (TSP) [11], each artificial ant represents a simple "agent". Each agent explores the surrounding space and builds a partial solution based on local heuristics, i.e., distances to neighboring cities, and on information from previous attempts of other agents, i.e., pheromone trail or the usage of paths from previous attempts by the rest of the agents. In the first iteration, solutions of the various agents are only based on local heuristics. At the end of the iteration, "artificial pheromone" will be laid. The pheromone intensity on the various paths will be proportional to the optimality of the solutions. As the number of iterations increases, the pheromone trail will have a greater effect on the agents' solutions.

It is worth mentioning that ACO makes probabilistic decision in terms of the artificial pheromone trails and the local heuristic information. This allows ACO to explore larger number of solutions than greedy heuristics. Another characteristic of the ACO algorithm is the pheromone trail evaporation, which is a process that leads to decreasing the pheromone trail intensity over time. According to [12], pheromone evaporation helps in avoiding rapid convergence of the algorithm towards a sub-optimal region.

Please note that searching the feature space in the problem of feature selection is quite different from the other optimization problems that researchers attempted to solve using ACO. In the next section, we present our proposed ACO algorithm, and explain how it is used for searching the feature space and selecting an "appropriate" subset of features.

IV. The Proposed Search Procedure

For a given classification task, the problem of feature selection can be stated as follows: given the original set, $\mathcal{F}$, of n features, find subset $\mathcal{S}$, which consists of m features ($m < n$, $\mathcal{S} \subset \mathcal{F}$), such that the classification accuracy is maximized.

The feature selection representation exploited by artificial ants includes the following:

- n features that constitute the original set, $\mathcal{F} = \{f_1, \ldots, f_n\}$.
- A number of artificial ants to search through the feature space (na ants).
- $\mathcal{T}_i$, the intensity of pheromone trail associated with feature f_i, which reflects the previous knowledge about the importance of f_i.
- For each ant j, a list that contains the selected feature

subset, $S_j = \{s_1, \ldots, s_m\}$.

We propose to use a hybrid evaluation measure that is able to estimate the overall performance of subsets as well as the local importance of features. A classification algorithm is used to estimate the performance of subsets (i.e., wrapper evaluation function). On the other hand, the local importance of a given feature is measured using the Mutual Information Evaluation Function (MIEF) [16], which is a filter evaluation function.

In the first iteration, each ant will randomly choose a feature subset of m features. Only the best k subsets, $k < na$, will be used to update the pheromone trial and influence the feature subsets of the next iteration. In the second and following iterations, each ant will start with $m - p$ features that are randomly chosen from the previously selected k-best subsets, where p is an integer that ranges between 1 and $m - 1$. In this way, the features that constitute the best k subsets will have more chance to be present in the subsets of the next iteration. However, it will still be possible for each ant to consider other features as well. For a given ant j, those features are the ones that achieve the best compromise between pheromone trails and local importance with respect to S_j, where S_j is the subset that consists of the features that have already been selected by ant j. The Updated Selection Measure (USM) is used for this purpose and defined as:

$$USM_i^{S_j} = \begin{cases} \dfrac{(\tau_i)^{\eta}\left(LI_i^{S_j}\right)^{\kappa}}{\sum_{g \notin S_j}(\tau_g)^{\eta}\left(LI_g^{S_j}\right)^{\kappa}} & \text{if } i \notin S_j \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

where $LI_i^{S_j}$ is the local importance of feature f_i given the subset S_j. The parameters η and κ control the effect of pheromone trail intensity and local feature importance respectively. $LI_i^{S_j}$ is measured using the MIEF measure and defined as:

$$LI_i^{S_j} = I(C; f_i) \times \left[\frac{2}{1 + \exp(-\alpha D_i^{S_j})} - 1\right] \quad (2)$$

where

$$D_i^{S_j} = \min_{f_s \in S_j}\left[\frac{H(f_i) - I(f_i, f_s)}{H(f_i)}\right] \times \frac{1}{|S_j|}\sum_{f_s \in S_j}\left[\beta\left(\frac{I(C;\{f_i, f_s\})}{I(C; f_i) + I(C; f_s)}\right)^{\gamma}\right] \quad (3)$$

the parameters α, β, and γ are constants, $H(f_i)$ is the entropy of f_i, $I(f_i; f_s)$ is the mutual information between f_i and f_s, $I(C; f_i)$ is the mutual information between the "class labels" and f_i, and $|S_j|$ is the cardinal of S_j. For detailed explanation of the MIEF measure, the reader is referred to [16].

Below are the steps of the algorithm:

1. Initialization:
   - Set $\tau_i = cc$ and $\Delta\tau_i = 0$, ($i = 1, \ldots, n$), where cc is a constant and $\Delta\tau_i$ is the amount of change of pheromone trial quantity for feature f_i.
   - Define the maximum number of iterations.
   - Define k, where the k-best subsets will influence the subsets of the next iteration.
   - Define p, where $m - p$ is the number of features each ant will start with in the second and following iterations.
2. If in the first iteration,
   - For $j = 1$ to na,
     - Randomly assign a subset of m features to S_j.
   - Goto step 4.
3. Select the remaining p features for each ant:
   - For $mm = m - p + 1$ to m,
     - For $j = 1$ to na,
       - Given subset S_j, Choose feature f_i that maximizes $USM_i^{S_j}$.
       - $S_j = S_j \cup \{f_i\}$.
   - Replace the duplicated subsets, if any, with randomly chosen subsets.
4. Evaluate the selected subset of each ant using a chosen classification algorithm:
   - For $j = 1$ to na,
     - Estimate the Mean Square Error (MSE_j) of the classification results obtained by classifying the features of S_j.
   - Sort the subsets according to their MSE. Update the minimum MSE (if achieved by any ant in this iteration), and store the corresponding subset of features.
5. Using the feature subsets of the best k ants, update the pheromone trail intensity and initialize the subsets for next iteration:
   - For $j = 1$ to k, /* update the pheromone trails */

$$\Delta\tau_i = \begin{cases} \dfrac{\max_{g=1:k}(MSE_g) - MSE_j}{\max_{h=1:k}\left(\max_{g=1:k}(MSE_g) - MSE_h\right)} & \text{if } f_i \in S_j \\ 0 & \text{Otherwise} \end{cases} \quad (4)$$

$$\tau_i = \rho.\tau_i + \Delta\tau_i \quad (5)$$

   where ρ is a constant such that (1 - ρ) represents the evaporation of pheromone trails.
   - For $j = 1$ to na,
     - From the features of the best k ants, randomly produce $m - p$ feature subset for ant j, to be used in the next iteration, and store it in S_j.
6. If the number of iterations is less than the maximum number of iterations, or the desired MSE has not been achieved, goto step 3.

It is worth mentioning that there is little difference between the computational cost of the proposed algorithm and the GA-based search procedure. This is due to the fact that both of them evaluate the selected subsets using a "wrapper approach", which requires far more computational cost than the "filter approach" used in the proposed algorithm to evaluate the local importance of features.

V. Experimental Results

A. Classification of Speech Segments

We conducted an experiment to classify speech segments according to their manner of articulation. Six classes were considered: vowel, nasal, fricative, stop, glide, and silence. We used speech signals from the TIMIT database, which has predefined segment boundaries.

Three different vectors of features were extracted from each speech frame: 16 log mel-filter bank (MFB), 12 linear predictive reflection coefficients (LPR), and 10 wavelet energy bands (WVT). A context dependent approach was adopted to perform the classification. So, the features used to represent each speech segment Seg_n were the average frame features over the first and second halves of segment Seg_n and the average frame features of the previous and following segments (Seg_{n-1} and Seg_{n+1} respectively). Hence, the baseline feature vectors based on MFB, LPR, and WVT consist of 64, 48 and 40 features respectively.

An Artificial Neural Network (ANN) was used to classify the features of each baseline vector into one of the six manner-of-articulation classes. Segments from 152 speakers (56456 segments) were used to train the ANNs, and from 52 speakers (19228 segments) to test them. The obtained classification accuracy for MFB, LPR and WVT were 87.13%, 76.86% and 84.57% respectively. It is clear that MFB achieved the best performance among the three baseline vectors; however, it used more features. The LPR on the other hand was outperformed by WVT despite the fact that it used more features.

The three baseline feature vectors were concatenated to form a new set of 152 features. The GA and the proposed ACO algorithms are used to select from these features. The GA-based selection is performed using the following parameter settings: population size = 30, number of generations = 20, probability of crossover = 0.8, and probability of mutation = 0.05. The obtained strings are constrained to have the number of '1's matching a predefined number of desired features. The *MSE* of an ANN trained with randomly chosen 2000 segments is used as the fitness function.

The parameters of the ACO algorithms described in the previous section are assigned the following values:

- $\eta = \kappa = 1$, which basically makes the trail intensity and local measure equally important.
- $\alpha = 0.3$, $\beta = 1.65$ and $\gamma = 3$, are found to be an appropriate choice for this and other classification tasks.
- The number of ants, $na = 30$, and the maximum number of iterations is 20. These values are chosen to justify the comparison with GA.
- $k = 10$. Thus, only the best $na/3$ ants are used to update the pheromone trails and affect the feature subsets of the next iteration.
- $m - p = \max(m - 5, \text{round}(0.65 \times m))$, where p is the number of the remaining features that need to be selected in each iteration. It can be seen that p will be equal to 5 if $m \geq 13$. The rational behind this is that evaluating the importance of features locally becomes less reliable as the number of selected features increases. In addition, this will reduce the computational cost especially for large values of m.
- The initial value of trail intensity $cc = 1$, and the trail evaporation is 0.25, i.e., $\rho = 0.75$.
- Similar to the GA-based feature selection, the *MSE* of an ANN trained with randomly chosen 2000 segments is used to evaluate the performance of the selected subsets in each iteration.

The selected features of each method are classified using ANNs, and the obtained classification accuracies of the testing segments are shown in Fig. 1. The following points can be deduced:

- Both feature selection methods were able to achieve classification accuracy similar to that of the LPR baseline feature vector with far less number of features ($|S_j| < 15$ features for GA, and $|S_j| < 10$ features for ACO).
- The ACO was able to achieve similar classification accuracy to that of the WVT baseline feature vector with smaller number of features ($|S_j| < 35$). On the other hand, the 40 features selected using GA was not enough to match the performance of WVT.
- When ACO and GA are used to select 64 features, they both achieved similar or slightly better performance than that of the MFB baseline feature vector.
- The overall performance of ACO is better than that of GA, where the average classification accuracy of ACO and GA over all the cases are: 84.23% and 83.47% respectively.

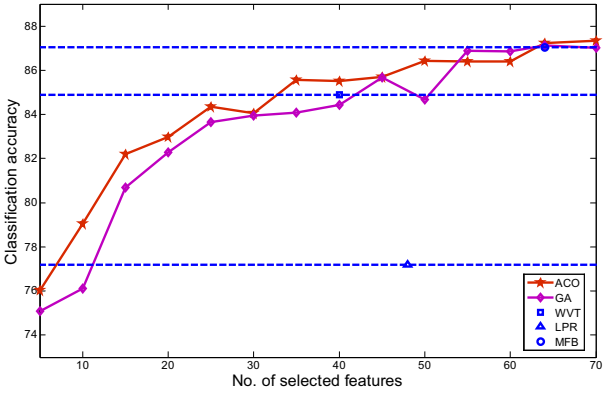

Fig. 1 Performance of ACO and GA (Speech segment classification)

B. Texture Classification

The second experiment was carried in texture classification. Nine textures were considered: bark, brick, bubbles, leather, raffia, water, weave, wood and wool [18]. Gaussian noise, with different signal-to-noise ratios, has been added to (1024 × 1024 pixels) images of each texture class to form the training and testing sets. 961 patterns were obtained from each image using (64 × 64) windows with an overlap of 32 pixels.

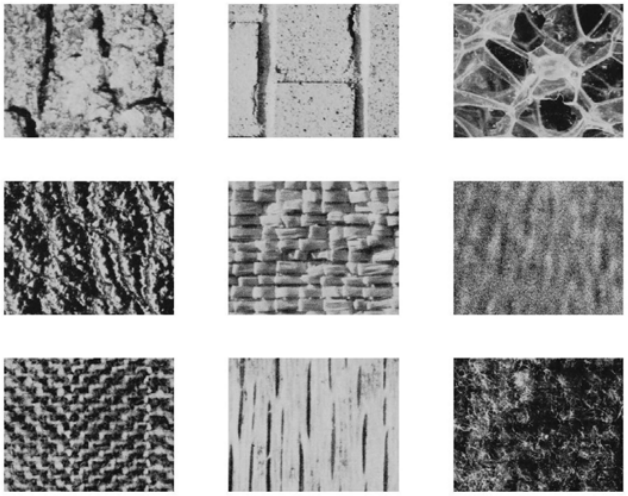

Fig. 2 256 × 256 windows of the clean texture images

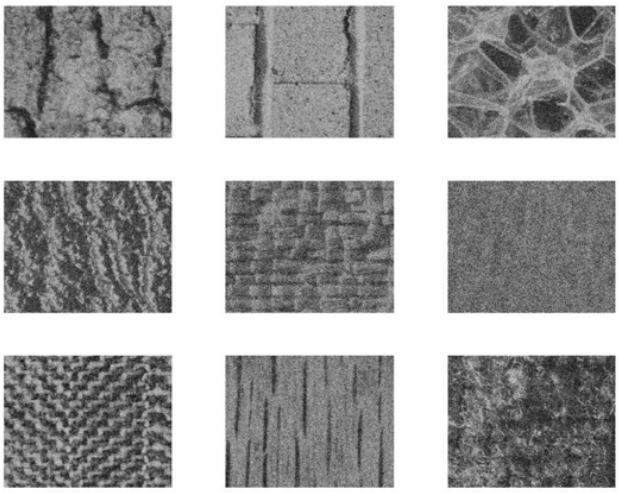

Fig. 3 256 × 256 windows of the noisy texture images

Figs. 2 and 3 show the clean and noisy texture images used.

Four 9 dimensional feature vectors were calculated using statistics of sum and difference histogram (*SDH*) of the co-occurrence matrix with different directions: vertical, horizontal, and the two diagonals (SDH_1, SDH_2, SDH_3 and SDH_4). For each direction, nine features were extracted: mean, variance, energy, correlation, entropy, contrast, homogeneity, cluster shade, and cluster prominence. The fractal dimension (*FD*) has been used to form the tenth feature of each vector. The energy contents (*E*) of texture images have been used to form another feature vector using 9 different masks, and its tenth feature was *FD*.

Each one of these five baseline feature vectors was used as input to an ANN. The numbers of training and testing patterns were 71354 and 23785 respectively. The classification accuracies obtained were 76.17%, 76.04%, 74.06%, 75.23%, and 89.39%. It is clear that the *E* vector performed extremely well compared to the other four vectors, where the ratio between the error rate of the second best vector (SDH_1) and that of *E* is 2.25. It is worth mentioning that the first four feature vectors (SDH_1, …, SDH_4) were found to exhibit a high degree of correlation.

The five baseline feature vectors were concatenated to form a feature set of 50 features. Both ACO and GA were applied to select from those features using the same parameters of the speech segment experiment. Fig. 4 shows the classification accuracy of the selected features.

Since the objective of feature selection is to improve classification accuracy, then a good feature selection method must achieve a similar performance to that of the *E* baseline feature vector with smaller number of features. GA was able to achieve this target by selecting 6 features. On the other hand, the 6 features selected by ACO were able to outperform *E*, and hence achieve better result compared to their GA counterparts. This represents a very good improvement when compared to the baseline feature vectors.

In addition, Fig. 4 shows that the ACO gives better results than GA in almost all cases. The performance of the whole feature set, which consist of 50 features, is indicated by the horizontal dash-dotted line in the figure. The ACO achieved similar performance using 20 features only, while GA could not match that and it needed more features to achieve such performance.

The above two experiments show the superiority of the proposed ACO algorithm, since it achieved similar or better performance compared to the baseline feature vectors with a lower number of features, and it outperformed the GA in almost all considered cases.

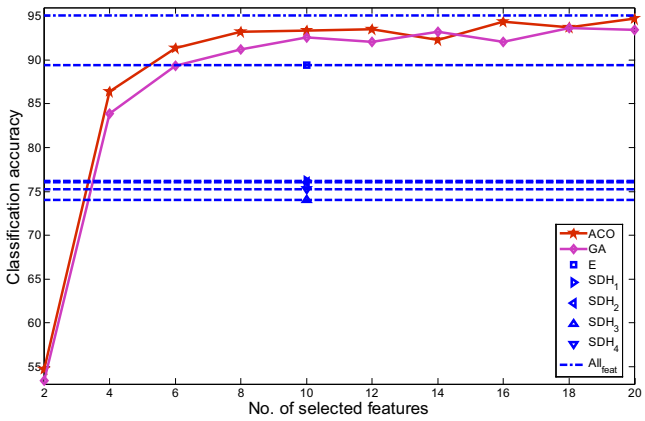

Fig. 4 Performance of ACO and GA (texture classification)

VI. Conclusion

In this paper, we presented a novel feature selection search procedure based on the Ant Colony Optimization metaheuristic. The proposed algorithm utilizes both local importance of features and overall performance of subsets to search through the feature space for optimal solutions. When used to select features for speech segment and texture classification problems, the proposed algorithm outperformed

GA-based feature selection. The proposed algorithm will be further studied and applied to other classification problems in the future.

REFERENCES

[1] A.L. Blum and P. "Langley. Selection of relevant features and examples in machine learning". *Artificial Intelligence*, 97:245–271, 1997.
[2] M.A. Hall. *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.
[3] R. Kohavi. *Wrappers for performance enhancement and oblivious decision graphs*. PhD thesis, Stanford University, 1995.
[4] P. Gallinari T. Cibas, F.F. Soulie and S. Raudys. "Variable selection with neural networks". *Neurocomputing*, 12:223_248, 1996.
[5] J. Kittler. "Feature set search algorithms". In C. H. Chen, editor, *Pattern Recognition and Signal Processing*. Sijhoff and Noordhoff, the Netherlands, 1978.
[6] P. Pudil, J. Novovicova, and J. Kittler. "Floating search methods in feature selection". *Pattern Recognition Letters*, 15:1119-1125, 1994.
[7] P.M. Narendra and K. Fukunaga. "A branh and bound algorithm for feature subset selection". *IEEE Transactions on Computers*, C-26: 917-922, 1977.
[8] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," *IEEE Transactions on Intelligent Systems*, 13: 44–49, 1998.
[9] M. Gletsos, S.G. Mougiakakou, G.K. Matsopoulos, K.S. Nikita, A.S. Nikita, and D. Kelekis. "A Computer-Aided Diagnostic System to Characterize CT Focal Liver Lesions: Design and Optimization of a Neural Network Classifier" *IEEE Transactions on Information Technology in Biomedicine*, 7: 153-162, 2003.
[10] I.-S. Oh, J.-S. Lee, and B.-R. Moon, "Hybrid Genetic Algorithms for Feature Selection" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26: 1424-1437, 2004.
[11] M. Dorigo, V. Maniezzo, and A. Colorni. "Ant System: Optimization by a colony of cooperating agents". *IEEE Transactions on Systems, Man, and Cybernetics* – Part B, 26:29–41, 1996.
[12] T. Stützle and M. Dorigo. "The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances". In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, Kluwer Academic Publishers, Norwell, MA, 2002.
[13] G. Di Caro and M. Dorigo. "AntNet: Distributed stigmergetic control for communications networks". *Journal of Artificial Intelligence Research*, 9:317–365, 1998.
[14] R.S. Parpinelli; H.S. Lopes; A.A. Freitas, "Data mining with an ant colony optimization algorithm", *IEEE Transactions on Evolutionary Computation*, 6: 321 - 332 2002.
[15] G. Di Caro and M. Dorigo. "AntNet: Distributed stigmergetic control for communications networks". *Journal of Artificial Intelligence Research*, 9:317–365, 1998.
[16] R. Montemanni, L.M. Gambardella, A.E. Rizzoli and A.V. Donati. "A new algorithm for a Dynamic Vehicle Routing Problem based on Ant Colony System". *Proceedings of ODYSSEUS 2003*, 27-30, 2003.
[17] A. Al-Ani, M. Deriche and J. Chebil. "A new mutual information based measure for feature selection", *Intelligent Data Analysis*, 7: 43-57, 2003.
[18] Signal and Image Processing Institute, USC. USE-SIPI image database, 1981. http://sipi.usc.edu/services/database/.