SPEEDING GA-BASED ATTRIBUTE SELECTION FOR IMAGE INTERPRETATION

by

Q. Zhang
R. S. Michalski

Reports of the Machine Learning and Inference Laboratory, MLI 97-5, George Mason University, Fairfax, VA, 1997.

SPEEDING GA-BASED ATTRIBUTE SELECTION FOR IMAGE INTERPRETATION

Qi Zhang   Ryszard S. Michalski*

Machine Learning and Inference Laboratory
George Mason University

***Also with GMU Departments of Computer Science and Systems Engineering, and the Institute of Computer Sciences, Polish Academy of Science**

P97-6
MLI 97-5

March 1997

SPEEDING GA-BASED ATTRIBUTE SELECTION FOR IMAGE INTERPRETATION

ABSTRACT

This paper addresses the problem of GA-based attribute selection. Previous work in this direction considered only how to represent a problem so that a genetic algorithm could work on it and then search for a satisfactory attribute subset. Even though good experimental results were reported, they were usually acquired at the cost of time. This paper presents a novel approach to this problem. In particular, it introduces attribute information during genetic evolution in order to make some promising attributes more likely appear in a new generation. In this way, the evolution process is faster and satisfactory results can be achieved with less time. Preliminary experimental results in image interpretation show that this approach is promising.

KEYWORDS: evolutionary computation, attribute selection, genetic algorithm, machine vision, image interpretation, learning.

ACKNOWLEDGEMENTS

This research was conducted in the Machine Learning and Inference Laboratory at George Mason University. The Laboratory's activities are supported in part by the Defense Advanced Research Projects Agency under grant F49620-95-1-0462 administered by the Air Force Office of Scientific Research, in part by the National Science Foundation under grants DMI-9496192 and IRI-9020266, and in part by the Office of Naval Research under grant N00014-91-J-1351.

SPEEDING GA-BASED ATTRIBUTE SELECTION FOR IMAGE INTERPRETATION

Q. Zhang R.S. Michalski*

Machine Learning and Inference Laboratory
George Mason University
Fairfax, VA 22030, USA
{qzhang, michalski}@aic.gmu.edu
http://www.mli.gmu.edu

*Also with GMU Departments of Computer Science and Systems Engineering, and the Institute of Computer Sciences, Polish Academy of Science

ABSTRACT

This paper addresses the problem of GA-based attribute selection. Previous work in this direction considered only how to represent a problem so that a genetic algorithm could work on it and then search for a satisfactory attribute subset. Even though good experimental results were reported, they were usually acquired at the cost of time. This paper presents a novel approach to this problem. In particular, it introduces attribute information during genetic evolution in order to make some promising attributes more likely appear in a new generation. In this way, the evolution process is faster and satisfactory results can be achieved with less time. Preliminary experimental results in image interpretation show that this approach is promising.

KEYWORDS: evolutionary computation, attribute selection, genetic algorithm, machine vision, image interpretation, learning.

1 INTRODUCTION

Image interpretation is a subarea of machine vision and refers to those methodologies that exploit computer technologies and automatically interpret, fully or in part, images which are black/white, color or in other forms taken on the ground, by airplanes or satellites. It has important military and civil applications. Methodologies differ from each other in theories and techniques, but their very beginning step is the same: defining a set of attributes and selecting a satisfactory subset from it. The quality of selected attributes is crucial to the performance of a image interpretation system.

How to define attributes is beyond the scope of this paper. Selection of a satisfactory subset from available attributes is of much interest to researchers because normally there is nonlinear interaction among attributes and not any attribute subset has the same discriminating power and simply adding an attribute could degrade performance of a system [Kreithen, 1993]. From the perspective of machine learning, selecting a good subset of attributes is actually finding a good representation space which is crucial for any learning or classification task. Another benefit from attribute selection is that it could make a system work faster due to fewer attributes.

Many researchers contributed to this field [e.g., John et al., 1993; Koller and Sahami, 1996; Vafaie and DeJong, 1993; Bala et al., 1995]. Recently, GA-based approach to this problem attracted many researchers [e.g., Vafaie and DeJong, 1993; Bala et al., 1995]. This direction seems promising and some good results were reported [Bala et al., 1995]. However, previous GA-based work usually achieved its results at the expense of time. This paper presents a novel way of using genetic algorithms to address attribute selection, that is, achieving good results without consuming much time. We inserted easily available attribute information into GA evolution to speed the process of searching for a satisfactory subset from candidate attributes. This idea is demonstrated in a multistrategy learning system which combines the genetic algorithm [De Jong, 1996] and AQ15c [Michalski et al., 1986, Wnek et al., 1995]. Specifically, it starts with statistical analysis of available attributes and selects some potentially promising ones and makes them more likely appear in the first generation of genetic evolution. As for an evolved attribute subset, AQ15c is called to perform learning and testing upon actual examples. The testing accuracy is taken as the fitness value of this given subset and is also assigned, in a statistical way, to those attributes appearing in the subset. Statistically good attributes have more chances of being selected into an attribute subset of the next generation. This process ends while a satisfactory subset is found or the maximal number of generations is reached. Preliminary experiments done in attribute selection for interpretation of natural scene images [Michalski et al., 1996] showed an pronounced speeding of GA-based search for a good attribute subset.

2 BACKGROUND

The attribute is the basis of interpretation, classification or recognition. Without attributes, no man or machine could interpret, classify or recognize scenes or objects from images. However, it is known that usually not any subset of available attributes can bring the same performance to a system due to the nonlinear interaction among attributes and so simply adding an attribute would probably result in the performance degradation of a system [Kreithen et al., 1993]. Thus, the goal of attribute selection is to select the best subset or a satisfactory one according to some criterion. Image interpretation is a good application domain for attribute selection because often many numerical attributes are available and the amount of data is huge and so it is hard to find out, at a glance or by a simple computation, which subset could lead to better or the best performance.

A question arises naturally: what is the meaning of "good" when attributes being referred to? The authors observe that there are two kinds of "good" attributes: individually good and collectively good. A attribute is considered to be *individually* good if it itself satisfies some requirements based on analysis of its properties and given data. For example, orthogonality is a requirement whose definition is like "must measure *different* attributes of ..." [Kreithen et al., 1993]. Here by *different* Kreithen et al. meant different aspects or properties of data. Another example of such requirements is separability that can be defined as an attribute's ability of separating different object classes. However, there is a problem with selecting attributes directly according to these requirements: though finding such a set of attributes could often lead to satisfactory results, it is not guaranteed because of nonlinear interaction among attributes. The other problem is that on one hand requirements like orthogonality are not or hardly operational even by a human being and on the other hand requirements like separability seem to be operational but unable to be determined in reality because of the large number of attributes and object classes and noise in data. A attribute is considered to be *collectively* good if the set of attributes in which this attribute appears brings to a system good performance. This concept of "good" captures the cooperation and nonlinear interaction among attributes and is often the goal of so-called attribute selection. The relationship between this two kinds of good attributes is that an individually good attribute usually appears in the best or satisfactory attribute subset but a collectively good attribute is not necessarily individually good. The reason for the authors to introduce the concepts of individually good and collectively good is that their relationship has not drawn enough attention.

For attribute selection, a very naive method is to generate each possible subset of attributes and then test the performance. Clearly, the best subset could be determined because it is an exhaustive search. However its time cost is exponentially proportional to the number of attributes and it is almost never used in reality. Another method is ranking a candidate attribute based on some criterion followed by deleting some attributes with lower ranks [e.g., Baim, 1982]. This method ignores the nonlinear interactions among attributes. Some researchers [e.g., Imam and Vafaie, 1994] used heuristic search in attribute selection. They viewed the best or a satisfactory subset as a goal of a search process. This method usually runs fast; however, it often ends up with a very locally optimal attribute subset. Same as the ranking method, this approach is unable to capture nonlinear interaction among attributes and when the number of attributes is large, it is hard or impossible to find effective heuristics that could be used to guide a search process. Forsburg [1976] used an adaptive random search method which increased the probabilities of being selected of those attributes which appeared in generated knowledge descriptions (i.e., they are *relevant* attributes) to make them more likely be selected in the next subset.

Recently, genetic algorithms attracted many researchers working on this problem [e.g., Vafaie and DeJong, 1993; Bala et al., 1995]. This method takes advantage of the explorative power of genetic algorithms to search for a satisfactory attribute subset without exhaustive search. Compared to other methods, it usually provides better results, esp., in the case of a lot of attributes. Bala et al. (1995) was a good example of this direction and good experimental results were reported there. However, previous work in this area considered only how to represent a problem so that a genetic algorithm could work on it. It should be pointed out that such a GA-based search normally consumes much time, and it becomes even worse when the number of attributes is large, the amount of training data is huge and the testing method itself needs some time. This situation is often true in image interpretation and makes GA-based attribute selection impractical. This paper describes a new way of addressing this problem by introducing attribute information into genetic evolution.

3 ATTRIBUTE INFORMATION

The above observation of two kinds of good attributes and their relationship is the basis of this paper. A individually good attribute normally appears in an attribute subset which is the best or satisfactory; in other words, the best subset or a satisfactory subset usually can not exist without containing individually good attributes. So during genetic evolution, it may be better to let individually good attributes to more likely appear in generations. Based on this idea, the authors try to determine which attributes are individually good and increase their probabilities of appearing in an individual of a generation. In contrast, previous GA-based dealt with each attribute randomly or in an equally fair way. Even though in this way could the explorative power be strong, many individuals in a generation probably contain no or few individually good attributes and so the whole evolution is likely to consume more time before acquiring a satisfactory subset.

There are two places where attribute information can be introduced into GA evolution: forming the first generation and mutating within one individual of a population during evolution. The authors refer to these two sorts of information as *static* and *dynamic* respectively in this paper.

Static information is acquired from analysis of attributes based on training data. Separability is one property that could be evaluated from training data and thus we use it for static information. Other properties like orthogonality cannot be directly computed and must be determined by the designer and so they are not considered in this paper for the time of being. For each attribute, we try to determine its separability and assign a heuristic value to it. Attributes with high heuristics are considered individually good. We increase their probabilities of appearing in the individuals (each individual is an attribute subset) of the first generation of evolution. Notice that these attributes are not guaranteed to exist in the individuals of the first generation. The specific definition is the following:

Static information: Suppose that C is a class set and that i, j represent two classes respectively ($i, j \in C$). For an attribute f, calculate the average and standard deviation of each class in this attribute from training data, say $\overline{x_i}$ and $\overline{\sigma_i}$. Then the static information of f is:

$$\sum_{\{i,j\}\subseteq C; i<j} \frac{2}{\pi} \tan^{-1}\left[\frac{(\overline{x_{ik}} - \overline{x_{jk}})^2}{1000\sigma_{ik} * \sigma_{jk} + \varepsilon}\right].$$

In the above, only one of $\{i, j\}$ and $\{j, i\}$ is counted; ε is a very small number preventing the denominator from being zero.

Dynamic information is calculated in the evolution process. If an attribute subset, i.e., an individual in a population, results in high performance on testing data, then every attribute in this subset would get some credit. If averagely one attribute has high credit, then this attribute's probability of surviving the mutation in order to appear in a new individual in the next generation is increased. The dynamic information tries to capture both the concept of individually good and the concept of collectively good. Note that static information is from training data while dynamic information is from testing data.

Dynamic information: For an attribute f, add the testing accuracies of all the previous individuals (i.e., attribute subsets) since the first generation where f appeared and divide this sum by the number of such individuals. The result is defined as the dynamic information of attribute f.

In fact, there could be many ways to calculate static or dynamic information for attributes, if reasonable. For example, PROMISE [Baim, 1982] can be used for static information. The key idea here is to let potentially promising attributes have higher probabilities to appear in individuals.

4 METHODOLOGY

4.1 Application Domain

We applied a combination of GA and AQ15c for attribute selection in natural scene interpretation [Michalski et al., 1996], in which the system is asked to label the class of each area in a natural scene image (see Fig. 1).

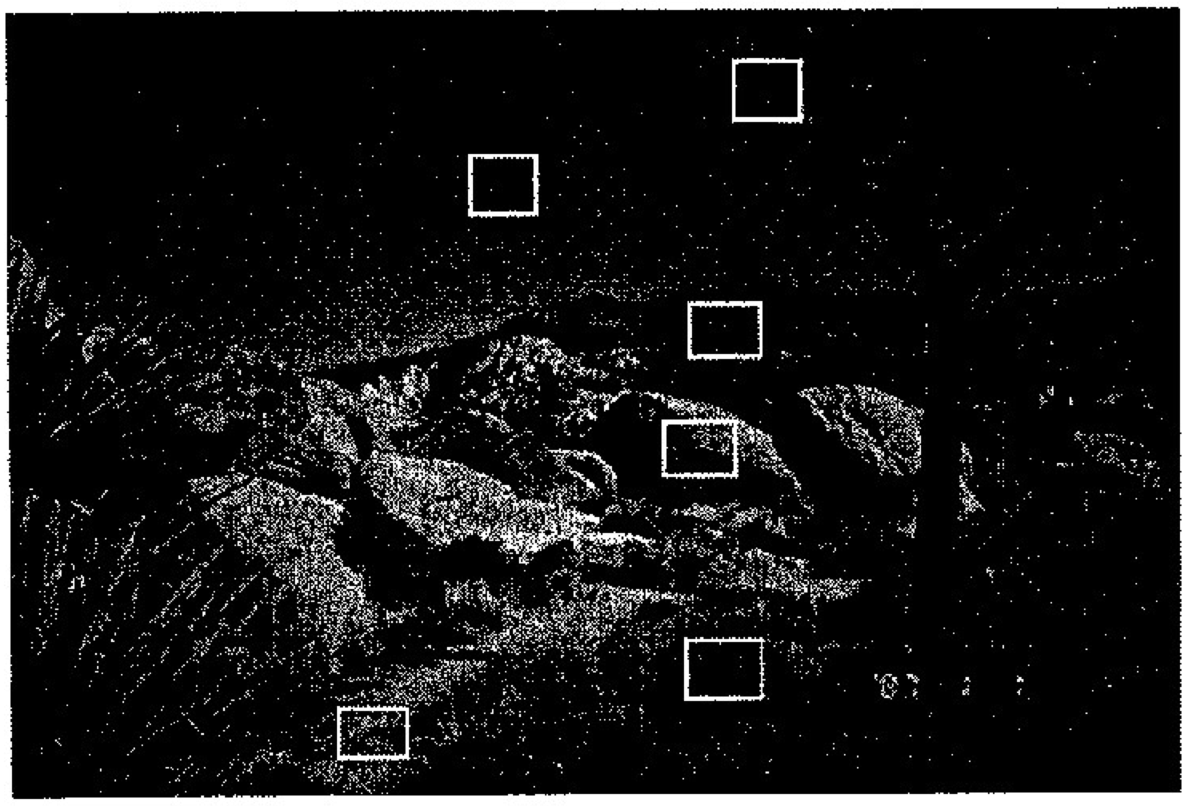

Fig. 1 A natural scene image.

4.2 Attribute Definitions

For each pixel in a image, it is taken as the pixel of interest and a set of attributes can be extracted. A total of 17 attributes were used in experiments. The first nine are computed according to some properties of the pixel itself: (1) red value; (2) green value; (3) blue value; (4) intensity; (5) saturation; (6) hue; (7) relative value of red = red - min(red, green, blue); (8) relative value of green = green - min(red, green, blue); (9) relative value of blue = blue - min(red, green, blue).

The other eight attributes in Fig. 2 are computed from the surrounding area of the pixel of interest and the area size is the size of each matrix in Fig. 2. Each matrix is also called a Laws mask [Laws, 1980]. Each matrix is used to detect some information around the pixel of interest. For instance, horizontal line operator is for detecting whether there are lines around the pixel of interest and this operator usually produces high values for grass pixels. The usage of each matrix is such: let the center of a matrix positioned at the pixel of interest, multiply each value in the matrix by the gray value of the pixel in the corresponding position, add all the products and the sum is the attribute value of the pixel of interest.

-1	-4	-6	-4	-1
0	0	0	0	0
2	8	12	8	2
0	0	0	0	0
-1	-4	-6	-4	-1

(10) horizontal line operator

-1	0	2	0	-1
-4	0	8	0	-4
-6	0	12	0	-6
-4	0	8	0	-4
-1	0	2	0	-1

(11) vertical line operator

-1	-4	-6	-4	-1
-2	-8	-12	-8	-2
0	0	0	0	0
2	8	12	8	2
1	4	6	4	1

(12) horizontal edge operator

-1	-2	0	2	1
-4	-8	0	8	4
-6	-12	0	12	6
-4	-8	0	8	4
-1	-2	0	2	1

(13) vertical edge operator

-1	0	2	0	-1
-2	0	4	0	-2
0	0	0	0	0
2	0	-4	0	2
1	0	-2	0	1

(14) horizontal V-shape operator

-1	-2	0	2	1
0	0	0	0	0
2	4	0	-4	2
0	0	0	0	0
-1	-2	0	2	1

(15) vertical V-shape operator

1	-4	6	-4	1
-4	16	-24	16	-4
6	-24	36	-24	6
-4	16	-24	16	-4
1	-4	6	-4	1

(16) frequency spot operator

1	-2	1
-2	4	-2
1	-2	1

(17) Laplacian operator

Fig. 2 Law masks for generating attributes.

4.3 Training Data, Testing Data and Discretization

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14	f15	f16	f17
6	6	5	6	2	6	1	0	0	5	5	5	3	2	9	1	5
8	7	6	7	2	6	2	1	0	6	5	5	3	4	9	1	5
8	7	6	7	2	6	2	1	0	6	6	5	5	7	6	2	6

Fig. 3. 17 attribute values of three selected rock pixels.

A 20 x 20 area from each kind of natural scene is selected from Fig. 1 (boxes) and 17 attributes are computed for each pixel in the selected areas. Fig. 3 gives some examples of training data.

60% of all the selected data are randomly taken for learning and the other 40% for testing [Weiss and Kulikowski, 1992]. Note that before learning rules for describing pixels by using AQ15c on training data and testing on testing data, each attribute is linearly discretized to one of fifteen levels for the experiments in this paper. Actually any other discretization scheme is applicable here. Since the purpose of the work presented is testing the effect of introducing attribute information on GA-based attribute selection, only the equal-interval scheme was adopted here. A learned pixel description (i.e. rule) by AQ15c is exemplified in Fig. 4.

Rock <:: [x1=5..14] [x5=0..4] [x13=3..10] [x14=1..7] [x15=9..12]

Fig. 4. One of the learned descriptions about rock pixels.

4.4 Acquisition of Attributes' Static Information

According to the definition above, calculate static information for every attribute from selected training data. See Fig. 5.

f1	f2	f3	f4	f5	f6	f7	f8	f9
8.390	5.324	8.328	4.769	5.116	11.122	9.591	12.456	5.000

f10	f11	f12	f13	f14	f15	f16	f17
6.661	8.912	8.199	7.536	8.075	6.582	8.967	3.957

Fig. 5 Static attribute information

4.5 Genetic Evolution for Attribute Selection

The genetic evolution proceeds according to the following steps. An individual in a generation could be considered as an array of 1s and 0s, in which 1 indicates the attribute is used in this individual and 0 not.

Step 1: Select the top 5 attributes according to static information, and then increase their probabilities of appearing in the first generation.

Step 2: Generate the first generation.

Step 3: For each individual in a generation, use AQ15c to learn pixel descriptions from training data and then do test on the testing data. The testing accuracy represents the fitness of an individual. Assign the testing accuracy to attributes in the individual and calculate attributes' dynamic information.

Step 4: Do fitness proportional selection and uniform crossover to generate new individuals.

Step 5: Mutate within each new individual. Two ways: (1) standard mutation, i.e., every attribute has 1/L probability of being mutated (L is the number of attributes); (2) dynamic information enhanced mutation, specifically, select the top 5 attributes according to dynamic information, and increase their probabilities of appearing in children.

Step 6: Go to Step 3 to continue the evolution from the new generation.

In step 1, when creating individuals in the first generation, an attribute is usually selected based on a uniform random number within 0.0 to 1.0 and the probability of selecting it is 0.5. Due to static information, the probabilities of those selected attributes which have high static information are increased (in our experiments, 0.75) and so they are more likely to appear in the first generation. In step 5, when doing mutation in an individual, usually an attribute has a probability of 0.5 being mutated. In this work, if an attribute has high dynamic information, then its probability of being mutated is decreased (0.7 in our experiments) if it already appears in the individual and is increased (0.3 in our experiments) if it dose not appear in this individual. The evolution stops when a predefined number of generation are produced or a predefined accuracy is achieved.

5 EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Experimental Results

The genetic algorithm in the experiments was built upon [De Jong, 1996]. The population size is 20 and the experiments were done on a Sun 4.

Three kinds of experiments were done: "traditional" means the way [Bala et. al., 1995] would do; "static" means only static information was introduced into genetic evolution; "static+dynamic" means both static and dynamic information were used. 10 runs were performed for every kind of experiment. The average testing accuracy is plotted as best-so-far in the Y axis against the number of births in the X axis. Results are given in Fig. 6.

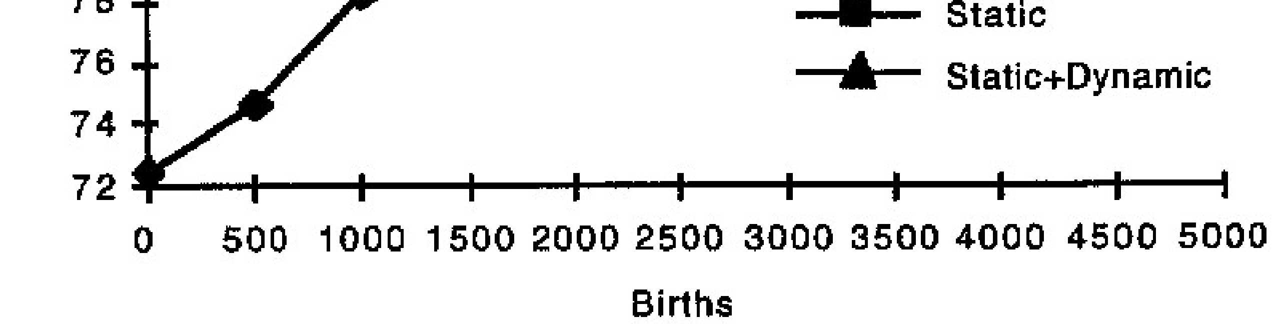

Fig. 6. The comparison of evolution speeds. Best-So-Far is testing accuracy.

5.2 Discussion

Fig. 6 shows a significant speeding effect due to introduced attribute quality information. Dynamic quality information did not result in much improvement at the early phase of evolution, because a few generations are not able to capture the statistical goodness of each attribute. When there were enough generations created, dynamic quality information worked to some degree. Note that both static and dynamic attribute information had a strong positive effect on evolution speed. It is possible to gain better performance if we do not give credit to each attribute in an individual during evolution but rather only these attributes which were actually used in obtained knowledge descriptions because only they contributed to acquired testing accuracy (Forsburg, 1976).

For the success of attribute quality information, the design of formulas of quality information seems crucial. We tried another formula for static quality information (not shown in this paper) but the speeding effect was not good. We set parameters t_1 and $t_2$in the above methodology to 5. We also tried selection of the top 3, 4, 6 attributes for probabilities to be increased. The results showed that the top 5 were the best for this given problem. We consider this issue to be problem-dependent and very important. If a system itself knows too much about the properties of candidate attributes, it could simply select them, and this way would clearly produce very good results quickly. Obviously, this is not always true. It is the case that only some of them may be selected. If too many attributes' probabilities are increased without well-founded understanding of them, then the evolution process is subject to going to and staying at some local optimum or spending more time in finding a satisfactory subset than without attribute quality information. On the other hand, selecting too few attributes may not produce the desired speeding effect. Thus selection of a appropriate number of attributes for probability increase is important.

Note that our methodology is similar to the work by Forsburg (1976) but different in many aspects. Attribute quality information is like *information content value* mentioned there; nonetheless the latter did not touch upon the concept of *static* quality information. Even though Forsburg adopted a random search, its theoretical properties were unclear. To some degree, the work there can be considered as a special case of our methodology with population size being one (no crossover, no mutation) and thus its search is not so powerful and systematic as genetic algorithms are. Further, we cannot evaluate its performance in terms of accuracy and speeding effect since they were not reported there.

6 CONCLUSION AND FUTURE WORK

This paper describes a promising way of speeding the GA-based attribute selection by introducing attribute information. It combines GA and AQ15c into a multistrategy learning system. Information of an attribute is introduced to determine whether to increase its survival probability. Experimental results are presented to show the feasibility of this new method. The preliminary results indicate the improvement in time in comparison with previous GA-based work in which the main focus was representing a problem so that a genetic algorithm could work on it.

There are some aspects which needs further work. Among them, selection of a appropriate number of attributes whose probabilities of survival are going to be increased is of special interest. The authors are going to find an automatic and adaptive way of determining this number so that the system could run more independently of the designer. Another work is to take as testbed more attributes and more application domains, esp. in the case of large number of attributes, to see the speeding effect because of the introduction of attribute information. Now the experimental results are still preliminary. Further, it is of interest to us to design other effective formulas to calculate attributes' information (esp., dynamic information) and capture more attribute properties in these formulas.

The authors believe that in application domains with a lot of numerical data like image interpretation, introduction of attribute information is a promising way to speeding GA-based attribute selection.

REFERENCES

Baim, P.W., 1982. The PROMISE Method for Selecting Most Relevant Attributes for Inductive Learning Systems. *Reports of the Intelligent Systems Group*, Department of Computer Science, University of Illinois at Urbana-Champaign, ISG 82-5.

Bala, J., Huang, J., Vafaie, H., DeJong, K. and Wechsler, H, 1995. Hybrid Learning Using Genetic Algorithms and Decision Trees for Pattern Classification. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence,* August, Motreal, Quebec, Canada.

De Jong, K., 1996. *Evolutionary Computation: Theory and Practice.* MIT Press.

Forsburg, S., 1976. AQPLUS: An adaptive random search method for selecting a best set of attribute from a large collection of candidates. Internal Report, Department of Computer Science, University of Illinois, Urbana, Illinois.

Imam, I.F., and Vafaie, H., 1994. An Empirical Comparison Between Global and Greedy-like Search for Attribute selection. *Proceedings of the Seventh Florida Artificial Intelligence Research Symposium*, May, Pensacola Beach, FL., pp. 66-70.

John, G., Kohavi, R., and Pfleger, K., 1994. Irrelevant Features and the Subset Selection Problem. *Proceedings of the Eleventh International Conference on Machine Learning,* July, Rutgers University in New Brunswick, New Jersey, pp. 121-129.

Koller, D., and Sahami, M., 1996. Toward Optimal Feature Selection. *Proceedings of the Thirteenth International Conference on Machine Learning,* July, Bari, Italy, pp. 284-292.

Kreithen, D.E., Halversen, S.D. and Owirka, F.J., 1993. Discriminating Targets from Clutter. *Journal of MIT Lincoln Lab.*, pp. 25-52.

Laws, K., 1980. Textured Image Segmentation. Ph.D. Thesis, University of Southern California, Department of Electrical Engineering.

Michalski, R.S., Mozetic, I., Hong, J., and Lavrac, N., 1986. The Multipurpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains. *Proceedings of the Fifth National Conference on Artificial Intelligence,* August, Philadelphia, PA., pp. 1041-1045.

Michalski, R.S., Zhang, Q., Maloof, M.A., and Bloedorn, E., 1996. The MIST Methodology and Its Application to Natural Scene Interpretation. *Proceedings of Image Understanding Workshop*, Palms Spring, CA., pp. 1473-1479.

Vafaie, H. and De Jong, K., 1993. Robust Feature Selection Algorithms. *Proceedings of the International Conference on Tools with AI,* Boston, MA., pp. 356-364.

Weiss, S.M., & Kulikowski, C.A., 1992. *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning and expert systems*, Morgan Kaufmann, San Mateo, CA.

Wnek, J., Kaufman, K., Bloedorn, E., and Michalski, R.S., 1995. Selective Induction and Learning System AQ15c: the Method and User's Guide. *Reports of the Machine Learning and Inference Lab.*, MLI 95-4, Machine Learning and Inference Lab., George Mason University, Fairfax, VA.