LEARNING TEXTURE CONCEPTS THROUGH MULTILEVEL SYMBOLIC TRANSFORMATIONS

by

J. Bala
R. S. Michalski

Proceedings of the Third International Conference on Tools for Artificial Intelligence,
San Jose, CA., Nov. 9-14, 1991.

Proceedings of Tools for AI - 91 Conference, San Jose Nov. 1991

Learning Textural Concepts Through Multilevel Symbolic Transformations

Jerzy W. Bala
Center for Artificial Intelligence
George Mason University
Fairfax, Virginia 22030

Ryszard S. Michalski
Center for Artificial Intelligence
George Mason University
Fairfax, VA 22030

Abstract

This paper presents the TEXTRAL system, used for determining structural visual properties of textures through symbolic transformations. The method consists of two phases: one that extracts information from raw textural images by applying convolution operators and learns an initial set of rules; and a second that iteratively extracts symbolic information from the transformed representation of initial image and learns another set of rules. The transformed symbolic representation is obtained by applying previously learned rules to a new image location and generating symbolic images based on rule assertions.

1 Introduction

Among the most informative properties in recognizing visual objects are their color and texture. The different textures in an image are usually very apparent to a human observer, but an automatic description of these patterns has proven to be complex. Texture provides very useful information for the automatic interpretation and recognition of the image by a computer. Textural features can be crucial for the segmentation of an image and can serve as the basis for classifying image parts. Many, if not all, objects in one's familiar environment can be recognized on the basis of just these two properties; i.e., without information about their shape, size or other characteristics. While measuring color is relatively easy, "measuring" texture is difficult. Texture may be described as the pattern of the spatial arrangement of different intensities (or colors) with two major characteristics: its coarseness and its directionality.

Traditionally, all methods of textural analysis have taken either the statistical approach [Haralick, 1973.], in which the statistical properties of the spatial distributions of the gray levels are used as the texture descriptors, or the structured [Rosenfeld, 1970] approach which conceives texture as an arrangement of a set of spatial subpatterns according to a certain placement rules. The statistical approach is usually motivated by a lack of strong regular patterns that are obvious in natural textures, and by a conjecture [Julesz, 1975] that second-order probability distributions suffice for human discrimination between two texture patterns. The structural texture models are best suited to situations in which complete descriptions of individual texture primitives are derivable from the image. This usually means that the texture primitives consist of relatively large numbers of pixels, and that the boundaries of the primitives are consistently discernable.

This paper presents a method for visual surface characterization and recognition based on adaptive image transformations and application of symbolic inductive learning. Some ideas of the presented method are based on [Michalski, 1972, 1973]. Our interest in this work is to produce symbolic descriptions of texture that are usable at the higher levels of a symbolic reasoning based vision system. These symbolic descriptions are used to isolate the texture primitives themselves in the original texture image. Once the texture primitives have been isolated, we compute "placement rules" via inductive learning techniques. These placement rules are used to generate symbolic images by labeling image elements (pixels) with class names. The labeling process is performed by matching new texture primitives extracted at the given image position with previously learned rules. The new set of rules is learned and the whole process is repeated until the desired performance level is achieved. The method has been tested on a number of different texture images from the Brodatz Album of Textures [Brodatz, 1966]. We present results of recognition for 12 homogeneous, noisy textures. The results show that the low-level vision symbolic computation can be successfully performed even for such textures.

2 Previous Accomplishments

A learning-based technique for texture domain was originally proposed by Michalski [Michalski, 1972, 1973], and was tested using ILLIAC III computer facilities. Early experiments produced very good results in discriminating even between very similar structural textures. Subsequently, this approach was applied to determine faults in laminates for aircraft wings using ultra sound images [Channic, 1989]. Recently, in different experiments [Pachowicz, 1989, Bala, 1990) the method was applied to raw, homogeneous, noisy textural images.

In one of our experiments [Pachowicz, 1989], the system was able to improve the average recognition rate (for six classes of texture acquired from very poor image data) from 70% of correct recognitions obtained for the k-NN pattern recognition method, to 80% for the symbolic machine learning approach, and to 91% for the symbolic machine learning approach incorporating optimization of texture class descriptions. The decrease of the average deviation of recognition rates has been observed in this experiment, making the system more stable for practical applications.

In a different experiment [Bala, 1990] a combination of structural and statistical features was used to tune the extraction and learning algorithm to produce acceptable rules. The structural features were derived for each pixel from a small neighboring area, and the statistical texture features were derived from co-occurrence matrices calculated in a larger neighborhood area. The experiment showed the capabilities and effectiveness of inductive learning techniques in a low-level vision domain.

3 Machine Learning Approach to Texture Recognition

This section describes the main idea of a symbolic learning approach to texture recognition (Figure 1).

Given a series of images classified by a human tutor into named surface and textural regions (module 1), the system (Figure 1) generates a procedure for classifying pixels into these regions.

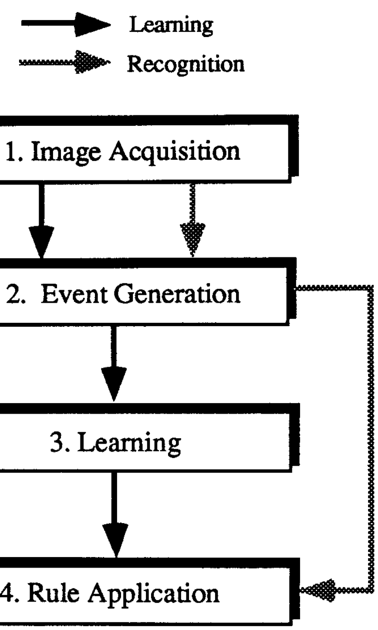


Figure 1. Texture recognition by machine learning.

Such a procedure consists of a sequence of operators that transform any given texture into a uniform set of labels characterizing the individual texture type. The major step in the procedure is the formulation of rules characterizing spatial properties of a texture. This is performed by the inductive learning algorithm (AQ method) [Michalski, 1986]. The system extracts a set of spatial texture samples, called events, from different texture regions (module 2). These events are input into the learning program (module 3) that formulates rules (covers).

The concept descriptions learned by the AQ algorithm are represented in VL_1, which is a simplified version of the Variable-Valued Logic System VL [Michalski, 1972], and are used to represent attributional concept descriptions. In the application of machine learning described in this paper, a concept represents a single texture class. A description of a concept is a disjunctive normal form which is called a cover. A cover is a disjunction of complexes. A complex is a conjunction of selectors. A selector is a form:

$$[L \# R] \qquad (1)$$

L is called the referee, which is an attribute.
R is called the referent, which is a set of values in the domain of the attribute L.
is one of the following relational symbols: =, <, >, >=, <=, <>.

In this learning method, each generated complex is associated with a pair of weights: *total* (t-weight) and *unique* (u-weight). In the experiments described in this paper we used a program called NEWGEM to generate rules. NEWGEM is one of the learning modules from the AQ family of learning programs. The following is an example of a NEWGEM complex (equality is used as a relational symbol):

[x1=1..3][x2=1][x4=0][x6=1..7][x8=1] *(t:6, u:2)* (2)

If this complex represents some textural information, *x1..x8* represent attributes extracted from a local area of a given texture class. The t-weight of a complex is the number of positive examples (examples from the class for which cover is generated) covered by the complex, and the u-weight is the number of the positive examples that are covered only by this complex. The complexes are ordered according to decreasing values of the t-weight. The following is an example of a cover generated by the AQ module

1. *[x1=5..10] [x2=5..13] [x3=13..54] [x4=3 [x6=0..4] [x7=6] (t: 6, u:5)*
2. *[x1=10..54] [x2=20..28] [x3=18..54] [x5=11..17] [x6=0..6] (t:5, u:5)*
3. *[x3=18..54] [x4=16..54] [x5=0..6] [x6=0..6] [x7=5..12] (t:5, u: 5)*

Learned rules (covers) are used to classify unknown instances. There are two methods for recognizing the concept membership of an instance: the strict match and the flexible match. In the strict match, one tests whether an instance strictly satisfies the condition part of a rule (a complex). In the flexible match, one determines the degree of closeness between the instance and the condition part. Such closeness is represented by a coefficient that can vary in range from 0. (does not match) to 1.0 (matches). In the strict matching one recognizes an instance if it is covered by the concept description. In the flexible matching, one determines the most closely related concept description. Given a selector in some attribute x whose domain is ordered list $<a_1,a_2,...,a_n>$, and an event where $x=a_k$, the normalized value for the selector $[x=a_j]$ is

$$1 - (| a_j - a_k | / n) \qquad (3)$$

If the selector has several values on its right hand side, the value closest to a_k is used. The complex is evaluated as an average value of evaluations of its selectors. The total evaluation of a class description for a given testing example is equal to the evaluation value of the best matching complex (the complex with the highest evaluation value).

Since an evaluation of an event to a cover is important in the regeneration phase of our method (this evaluation is used to determine class membership and to substitute pixel gray level value in the original textural area by the symbolic value that represents a class name).

The following is an example of an evaluation of the event *< 2, 4, 2, 1, 6, 10, 7, 5>* to the complex presented in (2):
Given:
complex:
[x1=1..3][x2=1][x4=0][x6=1..7][x8=1]
testing event:
event == < 2, 4, 2, 1, 6, 10, 7, 5,>
Number of possible levels per attribute:
L=55
Selector evaluations are:
eval(1)=1 (an attribute value is covered by the corresponding selector).
eval(2)=1-|1-4|/55=0.945
eval(3)=1 (selector for x3 is not present in the complex).
eval(4)=1-|0-1|/55=0.981
eval(5)=1 (same as for x3)
eval(6)=1-|7-10|/55=0.945

eval(7)=1 (same as for x3)
eval(8)=1-|5-1|/55=0.927
and complex evaluation is:

$$\text{Eval} = \left(\sum_{i=1}^{8} \text{eval}(i) \right) / 8 = 0.974$$

The total evaluation of a class description for a given testing example is equal to the evaluation value of the best matching complex, i.e. the complex with the highest evaluation value. The final recognition decision is made for a test dataset based on the classification of each instance from this set. The measure of recognition effectiveness is given as the percentage of the number of correctly classified test instances to the total number of instances of a dataset.

4 Learning Texture Concept in the TEXTRAL System

The learning method used by the TEXTRAL system consists of two basic phases (Figure 2). The first phase represents an application of various extraction operators used to learn the first set of rules. In this phase Law's masks [Laws, 1980] were used as the convolution operators to extract events that represent the energy content of the textural image. The second phase is the iterative process of the symbolic image regeneration (based on the assertion of rules learned in the previous iteration step) and learning a new set of rules by extracting symbolic events and applying AQ inductive method.

Before a learning process can take place, an experimenter outlines an area of the image that will be used for training (the *training area*). Within this area, the experimenter indicates sub-areas corresponding to the surfaces, whose identity the system is supposed to learn. The initial information is extracted from each textural class by applying a set of Law's convolution operators. This initial information is represented as the set of training events. A training *event* is a vector of values of *characteristic attributes* determined over the pixels in the training area. The Law's convolution operators were used in the TEXTRAL system in all experiments presented in this paper. These operators are shown in Figure 3. They are defined as follows:

R5R5 High frequency spot operator.

E5L5 Edge operators (vertical and horizontal).

E5S5 V-shape operators (vertical and horizontal).

L5S5 Line operators (vertical and horizontal).

Since three of these operators are directional, two versions, horizontal and vertical, of these directional operators were used to give a total number of seven attributes per event.

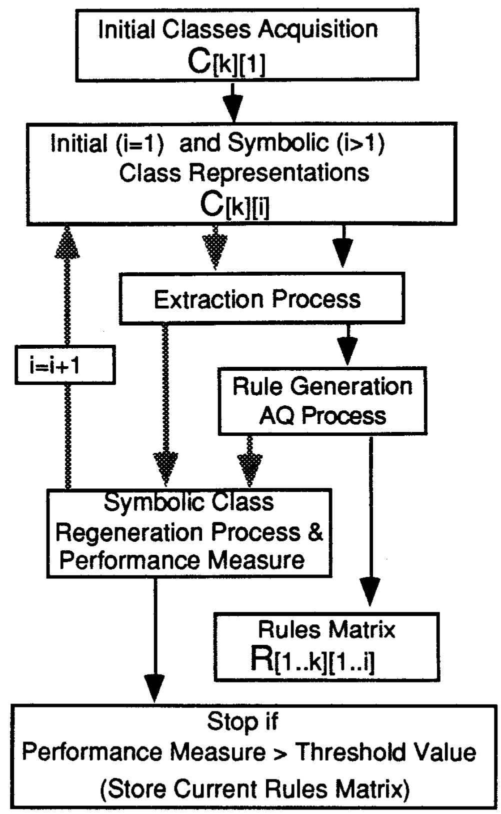


Figure 2. Learning Algorithm in the TEXTRAL system (*k* is the class index, *i* is the iteration index, shaded lines depict the regeneration process).

Before the extraction process each image is histogram-equalized. The extraction area of a given texture class is convoluted by each of the 7 operators yielding seven different convoluted areas. Each pixel in one of the 7 convoluted areas is replaced by an average of the absolute values in a local macrostatic window (10 by 10 pixels).

-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

L5S5 - vertical

-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

L5S5 - horizontal

-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

E5L5 - vertical

-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

E5L5 - horizontal

-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

E5S5 - horizontal

-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

E5S5 - vertical

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

R5R5

Figure 3. Operators used in initial extraction process.

The events are extracted only from a certain randomly chosen number of pixel positions for a given learning area. Each event has 7 attributes which represent pixel gray level values derived from 7 different areas (at the same $<x,y>$ location of a given area). The extracted events from each class are input into the AQ learning module. Generated rules describe discriminatory properties of a given set of classes and are stored in a rule matrix as $R[1..k][1]$ (with i index equal 1, Figure 2). These rules form the first column of this matrix. They are used to regenerate a symbolic representation of a given textural class.

The same extraction process is repeated, but this time to different pixel positions of the learning area. Each extracted event is evaluated by flexibly matching with $R[1..k][1]$ (rules learned from the initial extraction process). The result of this evaluation is used to determine the class membership of a given extracted event. The determined class name (symbolic value) is used as a pixel value in a transformed symbolic representation of a given textural area. Each pixel position in the learning area is assigned its symbolic value based on the dominant symbolic value assigned by the evaluation process in the neighborhood of this position. To accomplish this, a small window (e.g. 5 by 5 pixel positions) is scanned through the learning area, and the central pixel position in this window is substituted by the dominant symbolic value inside the 5 by 5 area. This operation of the "symbolic images smoothing" is necessary since not all pixel positions of the learning area are extracted for evaluation. This symbolic image represents a classification (or mis-classification) of the spatial characteristics of the first extraction and learning process ($i=1$).

A different extraction process (than the one used for the initial image) is used to derive event sets for the next iteration of the signature learning algorithm. The extraction operator is a simple window, 5 by 5 pixels wide, which extracts symbolic values of neighboring pixels $(x1,x2,..,x8)$ as depicted in Figure 4.

x1		x2		x3
x8		*		x4
x7		x6		x5

Figure 4. An operator used for the extraction process applied to the symbolic representation of the textural image area.

This operator is used in each subsequent learning and regeneration processes. Let us suppose that we have four classes (A,B,C,D). The extracted symbolic event might look like this:

$$<x1, x2, x3, x4, x5, x6, x7, x8> = <A, B, B, C, D, D, A, D> \quad (4)$$

As we can see the extracted volume of information from symbolic representation is substantially reduced. As the attributes values in symbolic representations we used linear types of attributes. Possible values of these attributes were chosen inside the range from 1 to 55 and the total number of these values is equal to the number of classes. The extracted symbolic event presented in (4) as input to a learning module might look like this:

$$\langle x1, x2, x3, x4, x5, x6, x7, x8\rangle = \langle 2, 10, 10, 20, 30, 30\ 2, 30\rangle \qquad (5)$$

where value 2 represents class A, 10 class B, 20 class C, and 30 class D.

The following shows an example of a 5 classes rule matrix (E denotes an empty entry ("no rules") in the matrix):

R(1,1)	R(1,2)	E	E
R(2,1)	E	E	E
R(3,1)	R(3,2)	R(3,3)	E
R(4,1)	R(4,2)	R(4,3)	R(4,4)
R(5,1)	R(5,2)	E	E

Each next iteration adds one column to the rule matrix. After each new symbolic image regeneration process is completed we compute statistics for all classes in form of a confusion matrix and estimate how well each class is represented by pixels with this class name value. A threshold value is introduced to determine how well a given class is represented in a transformed form by its name in a symbolic image. Let us say, if more than 90% of pixel location of a given class are assigned (by the matching process) its name we do not proceed with the next iteration step for this class. If more than 90% of the pixels are properly assigned, we say that this class is strongly represented by the rules derived from the previous extraction process. This also means that a given extraction process is relevant to the discriminatory characteristic of a given class. If less than 90% of pixel locations are assigned the name of the class, the iteration process is repeated. If a class is mis-represented in its symbolic representation (majority of pixels are assigned other class name) the next step of the learning algorithm has to proceed (probably yielding correct classification results in the next regeneration process). The ability to "force" the correct classification results by generating the next rule description of a given class (next step of regeneration and learning processes) is an important and novel feature of our method. This feature provides immunity to the extraction method chosen.

5 Using Surface Signatures in the Recognition Phase

The rule matrix is used to recognize unknown textural areas. The same sequence of extraction operators is used as describe previously (first convolution operators, as in Figure 3, followed by extraction window, as in Figure 4). The extraction events are flexibly matched against the first column of rule matrix. Based on matching results, the unknown textural image is transformed into its symbolic representation. The extraction process is repeated, but this time using a different extraction operator. During each iteration step there may be fewer classes to be matched with the still unknown class. The determination of class membership can be made during each iteration step depending on the matching results. In the case of the five class example (see rule matrix in the previous section), if the unknown class is matched strongly (some threshold value is established) to the second class of the first column of the signature matrix, the next iteration of the recognition algorithm is not needed.

6 Experimental Results

Using the method described in the previous sections we performed experiments with twelve textural classes (Figure 5). All textural classes were acquired from the Brodatz album of textures [Brodatz, 1966].

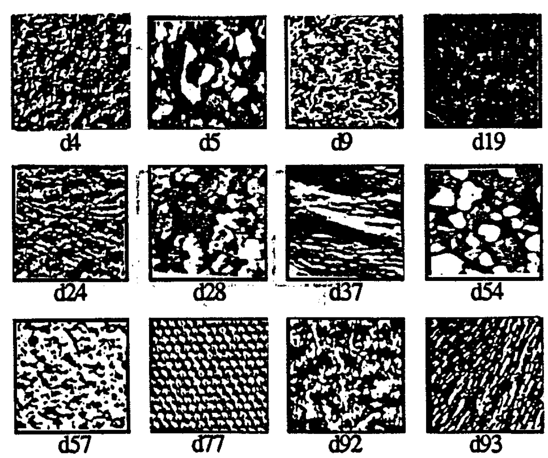
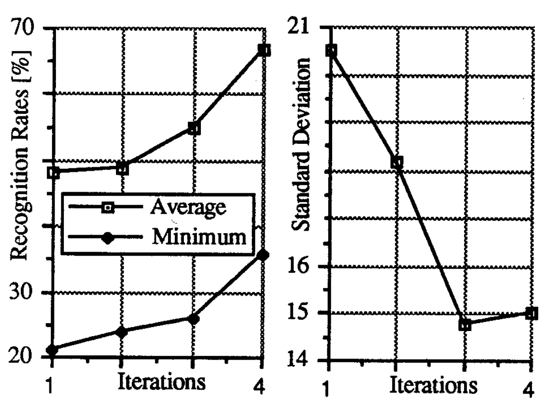


Figure 5. Samples of twelve texture classes (Approximately 50 by 50 pixels samples).

The learning area for each class was chosen to be a 100 by 100 square. From each class 100 events were derived to learn the rules in each iteration step. For each regeneration process 500 events were randomly chosen inside the learning areas. After completing the matching process, a 5 by 5 window was scanned through all pixel positions (100 by 100) of the learning area and the dominant recognized class inside this window was used as the symbolic value of the pixel represented by the central position inside this window (see section 5). By using this technique all 100 by 100 pixel positions were assigned their symbolic value (although only 500 had been used for regeneration/matching process).

Graph 1 presents the results of the 12 textures experiments. Testing area (different than learning area) in each texture class was used to extract 200 testing events. In each iteration these events were mached to the rule matrix (as described in section 5). Each class was correctly recognized in each of four iterations. The average recognition rate was increased from 48% to 58%. At the same time, standard deviation decreased from above 20 to 15. Minimum recognition rate increased from 21% to 36%. All these significant changes were obtained in only four iterations.



Graph 1. Recognition rates and standard deviation.

The obtained results are in accordance with the standard evaluation criteria for the recognition system that require: (i) an increase of the classification confidence when matching a class description with data belonging to this class, (ii) a decrease of the classification confidence when matching data with other class description, and (iii) perform on the similar confidence level for all classes when data is matched with their class descriptions. We express these criteria by an average recognition rate computed through testing all twelve texture classes (we require the highest averaged recognition rate), a standard deviation measuring the distribution of recognition rates from their mean value (this system stability criteria prefers the minimum value of standard deviation), and a recognition rate for the worst performing concept description (we seek the improvement of the minimum recognition rate, searched through all classes of texture considered in the learning process).

7 Conclusion

Learning mechanism in the TEXTRAL system is based on the following elements:

(i) extraction of information form raw textural images by applying convolution operators.

(ii) extraction of symbolic information from the transformed representation of initial data.

(iii) generation of class description by applying the AQ inductive learning methodology.

(iv) regeneration of symbolic representation by flexibly matching extracted information to learned rule descriptions from the previous iteration of the algorithm.

Experiments presented in this paper show the capabilities and effectiveness of inductive learning techniques in a low-level vision domain. The most important conclusion drawn from these experiments is that the presented method is unsusceptible to the attributes extraction process (see results from the twelve textures experiment). For a given extraction process, there might always be some texture classes learned that cannot be used for the recognition because of incorrect classification results. The relevant/discriminatory information derived from these classes is not captured by the extraction process. Choosing other extraction process may help to generate better descriptions for these classes, but there might still be a different subset of an unknown class set that cannot be used for recognition. To alleviate this problem in the TEXTRAL system, correct/incorrect classification results are used as the essential class dependent information that help to discriminate between different classes by learning the next set of rules. This approach differs from traditional approach which tries to improve the effectiveness of recognition by designing more sophisticated extraction methods and which applies classifiers in such a way that they are adapted on the feature set to take optimal advantage of the extracted information. Such an approach belongs to the class of feature extraction oriented methods, where an extraction of relevant feature plays a very important role. The main problem with traditional approaches is the lack of a universal extraction method that works effectively with noisy data.

ACKNOWLEDGEMENTS

This research was supported in part by the Office of Naval Research under grants No. N00014-88-K-0397 and No. N00014-88-K-0226, and in part by the Defense Advanced Research Projects Agency under the grant administered by the Office of Naval Research No. N00014-K-85-0878.
The authors wish to thank Dr Gheorghe Tecuci, Dr Bradley Kjell, Dr Arun Sood, and Dr Peter Pachowicz for valuable comments and discussion, and Janet Holmes for editing suggestions.

REFERENCES

[1] Brodatz, P., "A Photographic Album for Arts and Design", Toronto, Dover Publishing Co., 1966.

[2] Bala, J. "Combining Structural and Statistical Features in a Machine Learning Technique for Texture Classification", The Third International Conference on Industrial and Applications of Artificial Intelligence and Experts Systems, Charleston SC, July 1990.

[3] Bala, J. and K. De Jong,"Generation of Feature Detectors for Texture Discrimination by Genetic Search", The Second International Conference - IEEE Tools for AI, Washington D.C., November 1990.

[4] Bala, J.W. and Pachowicz, P.W., "Application of Symbolic Machine Learning to the Recognition of Texture Concepts" The Seventh IEEE Conference on Artificial Intelligence Applications, Miami Beach FL, February 1991.

[5] Bala, J.W. and Pachowicz, P.W., "Recognizing Noisy Patterns of Texture via Iterative Optimization and Matching of Their Rule Descriptions" Report of Machine Learning and Inference Laboratory, MLI-90-12, Center for Artificial Intelligence, George Mason University, 1990.

[6] Channic, T., "TEXPERT : An Application of Machine Learning to Texture Recognition", A publication of the Machine Learning and Inference Laboratory; MLI 89-17, George Mason University, Fairfax, Virginia.

[7] Haralick, R. M., K. Shanumugam, and I. Dinstein, "Textural features for image classification", IEEE Trans. Syst., Man, Cybern., vol. SMC-3, pp 610-621, Nov. 1973.

[8] Julesz, B., "Experiments in a visual perception of texture", Sci. Amer., vol. 232, Apr. 1975, pp 34-43.

[9] Laws, K.I., "Textured image segmentation", Ph.D. dissertation, Dept. of Engineering, Univ. of Southern California.

[10] Michalski, R. S., "A Variable-Valued Logic System as Applied to Picture Description and Recognition", in Graphic Languages, Nake, F. and Rosenfield, A. [Eds.], North Holland, 1972.

[11] Michalski R. S. "AQVAL/1--Computer Implementation of a Variable-Valued Logic System VL1 and Examples of Its Application to Pattern Recognition", First International Joint Conference on Pattern Recognition, October 30, 1973, Washington D.C.

[12] Michalski, R. S., "A Theory and Methodology of Inductive Learning", in *Machine Learning: An Artificial Intelligence Approach*, TIOGA Publishing, Palo Alto, CA, pp 83-134.

[13] Michalski, R.S., Mozetic I., Hong J.R., Lavrac N., "The AQ15 Inductive Learning System", Report No. UIUCDCS-R-86-1260, Department of Computer Science, University of Illinois at Urbane-Champaign, July, 1986.

[14] Pachowicz, P.W., "Low-level Numerical and Inductive Learning Methodology in Texture Recognition", IEEE International Workshop on Tools for AI, Washington, D.C. October 1989.

[15] Reinke R. E., "Knowledge Acquisition and Refinement Tools for the Advice Meta-Expert System", ISG 84-4, Department of Computer Science, University of Illinois at Urbana-Champaign, July 1984.

[16] Rosenfeld, A., and Lipkin, B.S. "*Texture analysis*", in Picture processing and psychopictorics, Academic Press, 1970, pp 300-322.