ABCM Symposium Series in Mechatronics - Vol. 4 - pp.21-30
Copyright © 2010 by ABCM

AN INTELLIGENT COMPUTER VISION SYSTEM TO ROCK CLASSIFICATION IN OIL AND GAS INDUSTRY

Laercio Brito Gonçalves, laercio@lmdc.uff.br
Computer Department – CEFET-RJ[1]
PGMEC-UFF[2]
[1]RJ, [2]Niterói, Rio de Janeiro, Brazil

Fabiana Rodrigues Leta, fabiana@ic.uff.br
Mechanical Engineering Department
LMDC – PGMEC
UFF – Universidade Federal Fluminense
Niterói, Rio de Janeiro, Brazil

Abstract. *This paper explores the use of a hierarchical neuro-fuzzy model for image classification of macroscopic rock texture. The relevance of this study is to help Geologists in diagnosing and planning the oil reservoir exploitation. The same methodology can be also applied to metals, in order to classify the different types of materials based on their grain texture. We present an image classification for macroscopic rocks, based on these texture descriptors and on a neuro-fuzzy approach. Then, a neural network is used, and the results obtained by both approaches are compared. To evaluate the system performance we used 50 RGB images, for each rock class and subclass, thus producing a total of 800 images. The classes of igneous rocks that make up the image database are: gneiss (two classes), basalt (four classes), diabase (five classes), and rhyolite (five classes). For each image were extracted: Hurst coefficient for gray and color images (a coefficient for each RGB channel); spatial variation coefficient (gray and color); entropy and co-occurrence matrix. Tests converged to optimum solution for the classification taking into account the fuzzy rules extraction which had a good performance.*

Keywords*: image classification, computer vision, texture, neuro-fuzzy systems, neural network, rock.*

1. INTRODUCTION

Oil is an essential energy resource for industrial production. It can be found in a variety of geological environments. The exploitation of oil is a large scale activity, in which the acquisition, distribution and use of expert knowledge are critical to decision making. Two sets of information are of fundamental importance in the exploitation of a new oilfield, which were: the oil reservoir geometry and the description of the type of porous rock that holds the oil.

Analyzing the oil reservoir geometry it is possible to identify the amount of oil in the reservoir. The second set of information consists in describing the porous rock that holds the oil, called reservoir rock. The quality of a reservoir is determined by the unique characteristics of its rock, such as the minerals that form it, the volume and shape of pores (spaces that preserve fluids within the rock), the connections between the pores and the physical-chemical processes that may have modified these characteristics.

The study of reservoir rocks is made based on a systematic description of rock samples collected from oil exploration boreholes, this is called petrography. The petrography is an activity performed in the laboratory, which incorporates the results of rock macroscopic and microscopic analysis. In macroscopic analysis, rock samples are cylindrical cleaved by drill bit, this is called witnesses. Using these samples, slices are withdrawn and there are prepared thin sections of 0.03 mm thickness, which will, in turn, analyzed with the use of optical microscopes of polarized light. It is a second important description, i.e. petrography under microscopic scale.

In macroscopic analyses there are described several physical characteristics such as color, structure, texture, size and orientation of grains and fossil when available. For instance, there are many ways to classify rocks; they can be described as igneous, sedimentary or metamorphic. The three classes of rocks above can also be subdivided into subclasses. The existence of a large number of classes and subclasses makes difficult the task of rock classification, requiring long time of training, since it depends on the identification of features based on images.

This paper will present the Neuro-Fuzzy Hierarchical model for automatic classification of rock texture images, called NFHB-Class and neural networks for classification problems where each class is composed of various subclasses. The system uses the following features: spatial variation coefficient, Hurst coefficient, entropy and co-occurrence matrix.

2. NFHB-CLASS MODEL

The NFHB-Class (Gonçalves *et al.* 2009) model is an extension of the NFHB Inverted model (Gonçalves 2001) and (Gonçalves *et al.* 2006) used in data classification. The main difference between the NFHB Inverted model to NFHB-Class model is the way the structure of the system is obtained. In NFHB Inverted model, the structure of the neuro-fuzzy model is created by NFHB (Souza 1999); afterward the system gets the pattern classification system. On the other side the NFHB-Class is capable of generating its own structure, thus avoiding the need to use the model NFHB.

2.1. Basic NFHB-Class Cell

A basic NFHB-Class cell (Gonçalves *et al.* 2009) is a mini neuro-fuzzy system that performs a fuzzy binary partitioning in a given area, according to the relevance of functions described by Eq. (1). The cell NFHB-Class generates two precise outputs (crisp) after a defuzzification process.

$$\begin{aligned} \mu &= sig[a(x_i - b)] \\ \rho &= 1 - \mu \end{aligned} \tag{1}$$

Figure 1 shows the basic representation of the cell-Class NFHB and Fig. 2 illustrates its details.

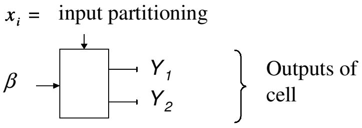


Figure 1. NFHB-Class cell.

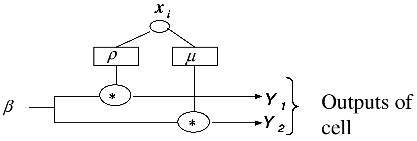


Figure 2. NFHB-Class cell schematic symbol.

The outputs (crisp) in a NFHB-Class cell are given by Eq. (2) and Eq. (3).

$$y_1 = \frac{\beta * \rho(x)}{\rho(x) + \mu(x)} \tag{2}$$

$$y_2 = \frac{\beta * \mu(x)}{\rho(x) + \mu(x)} \tag{3}$$

β can be one of the two scenarios below:

- The first cell input: in this case β = 1. The value '1' in the first cell input represents the entire input space, i.e. the entire universe of discussion of the variable x_i that is being used as an input cell.
- The output of a previous level: in this case β = y_i, where y_i represents one of the two outputs of a generic cell 'j', whose value is also calculated by the Eq. (2) and Eq. (3).

In NFHB-Class basic cell, the high pertinence function (μ) was implemented by a sigmoid function and the low pertinence function (ρ) was implemented as its complement [1- μ(x)]. The complement use leads to a simplification of the defuzzification procedure performed by Eq. (2) and Eq. (3), because the sum given by $\rho(x) + \mu(x)$ is equal to 1 for any values of 'x'. More details about the learning process of this process can be found in (Gonçalves 2001).

2.2. NFHB-Class Architecture

Figure 3 shows an example of an architecture NFHB-Class, obtained during the training system to a database that has three distinct classes, and its respective partition is illustrated in Fig. 4. The structure was created automatically without the need of using the model NFHB to the stage of training, as was the case of the model NFHB Inverted (Gonçalves *et al.* 2006).

As the model NFHB Inverted, in NFHB-Class architecture, Fig. 3, the system has several outputs that are connected to cells T-conorms that defining the classes. The output of the system (class1, class2 or class3) with higher value defines the class to which the pattern belongs.

The outputs of the leaves cells of the system, in Fig. 3, are listed below:

$$y1 = \rho_0 . \rho_1 \tag{4}$$

$$y2 = \rho_0 . \mu_1 . \rho_{12} \tag{5}$$

$$y3 = \rho_0 . \mu_1 . \mu_{12} \tag{6}$$

$$y4 = \mu_0 . \rho_2 \tag{7}$$

$$y5 = \mu_0 . \mu_2 \tag{8}$$

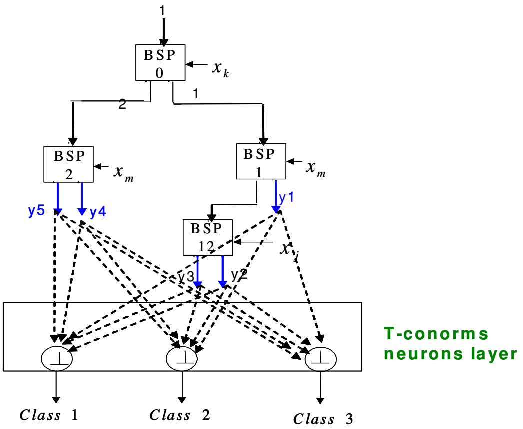


Figure 3. NFHB-Class architecture.

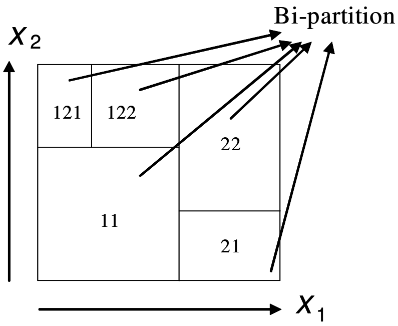


Figure 4. Input space partitioning of NFHB-Class.

After calculating the output of each leaf cell, is done the connection of these leaves cells with the T-conorms neurons, which can be obtained the final output of the classifier of patterns. Each T-conorm neuron is associated with a specific class, as can be seen in the example described in Fig. 3, where there are three distinct classes, and consequently, three T-conorms neurons.

The connections of the leaves cells with the T-conorms neurons are made, initially, connecting all the leaves cells with all the T-conorms neurons, as the number of classes that are organized the database. After this connection, it is necessary to establish weights for these connections (arcs). For the allocation of weights, we used the method of least squares with the use of the interactive method of Gauss-Seidel (Barret *et al.* 1994).

Having defined the strategy of how the leaves cells were connected to T-conorms neurons and theirs weights of these links, it is necessary to define which operators T-conorms will be used to obtain the final output of the model NFHB-Class.

All the outputs of the leaves cells were connected to all neurons T-conorms, as shown in Fig. 3. Each of these outputs is multiplied by the weight of its connection with the T-conorm neurons. The next step is to define how it should be the treatment of all these inputs of the neurons T-conorms that are derived from the leaves cells.

In the example of Fig. 3, the outputs of three T-conorm neurons are calculated according Eq. (9), Eq. (10) and Eq. (11).

$$y1 * w_{11} \oplus y2 * w_{21} \oplus y3 * w_{31} \oplus y4 * w_{41} \oplus y5 * w_{51} \tag{9}$$

$$y1 * w_{12} \oplus y2 * w_{22} \oplus y3 * w_{32} \oplus y4 * w_{42} \oplus y5 * w_{52} \tag{10}$$

$$y1 * w_{13} \oplus y2 * w_{23} \oplus y3 * w_{33} \oplus y4 * w_{43} \oplus y5 * w_{53} \tag{11}$$

Where: y1, y2, y3, y4, e y5 are the outputs of the leaf cells; W_{11}, W_{12}, W_{13}, W_{21}, W_{22}, W_{23}, W_{31}, W_{32}, W_{33}, W_{41}, W_{42}, W_{43}, W_{51}, W_{52} e W_{53} are the weight of the link between the leaf cell and the T-conorms neuron; $\oplus$ is the T-conorm operation used for processing the output of the neuron.

In this paper, the limited -sum T-conorm operator (Yager and Filev 1994) has been used. This operator is the most appropriated in this case, since it considers all inputs in the output calculation. Another T-conorm operator that is very popular in the literature, the *max*, only takes the maximum membership value into account, ignoring the membership values of the inputs.

The final output obtained of the NFHB-Class system is specified by the highest output obtained among all the T-conorm neurons, determining the class to which the input pattern belongs.

2.3. Learning Algorithm

In the neuro-fuzzy literature the learning process is generally divided in two parts: 1) the identification of the structure and 2) the adjustments of parameters. The NFHB-Class model follows the same process. However, only one algorithm carries out both learning task simultaneously. The learning algorithm of the NFHB-Class is performed in nine steps, as illustrated in the flowchart of Fig. 5.

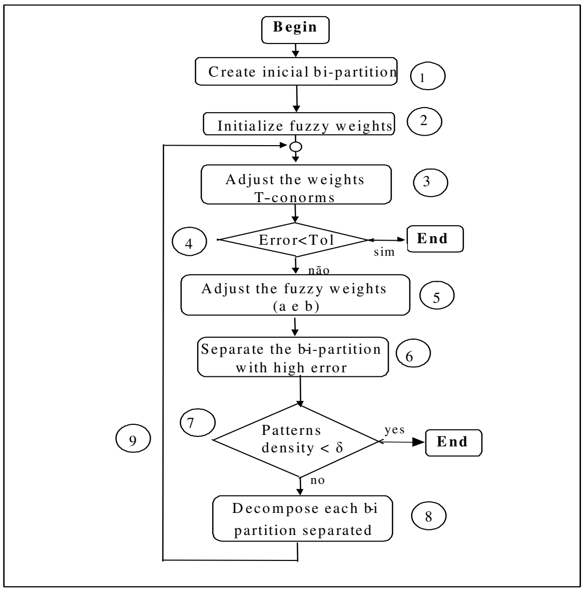

Begin
Create inicial bi-partition
1
Initialize fuzzy weights
2
Adjust the weights T-conorms
3
4
Error<Tol
sim
End
não
Adjust the fuzzy weights (a e b)
5
Separate the bi-partition with high error
6
7
Patterns density < δ
yes
End
9
no
Decompose each bi partition separated
8

Figure 5. Learning algorithm of NFHB-Class model.

2.4. Strategies for Selection of Attributes for the NFHB-Class System

In problems of pattern classification is important to define the goals and select, in the available database, which characteristics will be used. These features are attributes deemed as relevant for obtaining significant goals.

In the case of rock classification there are a large number of interested variables available, such as the Hurst coefficient for grayscale and RGB channels, grayscale and RGB channels spatial variation coefficient, in addition to the descriptors used to define texture characteristics obtained in image co-occurrence matrices. In general, we choose the most representative variables collection, which are called features.

The correct selection of the attributes avoids unnecessary partitioning, resulting more compact BSP tree structures, resulting in a better generalization, fewer rules and a greater interpretation degree (Gonçalves 2001).

Two methods for characteristics selection were tested: the Jang (Barret *et al.* 1994) algorithm which showed better performance, and the entropy method (Yager and Filev 1994). Besides those methods, there are several studies using other techniques of features selection, such as Principal Components Analysis (Jang 1994), (Lanas 2000), (Roffel *et al.* 1989), (Santen *et al.* 1997), machine learning (Dong and McAvoy 1996), (Aoyama, and Walsh 1997); hierarchical clustering (Blum and Langley 1997), (Langley 1994) and genetic algorithms (Talavera, 1999), (Dash and Lui 1997), (Yang and Honavar 1998).

Two strategies for selection, fixed and adaptive, have been proposed to deal with the selecting problem, i.e. which input variables should be applied to partitioning inputs of each NFHB-Class cell.

In the fixed selection sets strategy, as the order of the attributes is determined by Jang algorithm, during the model NFHB-Class learning process and architecture construction, each of these features is chosen and used as input for each level of BSP tree. The same input (attribute) is used for all nodes of the same level. This strategy generates unnecessary partitioning due to the fact that all nodes at the same level are forced to use the same fixed input, which is not always the best characteristic for this node. One of the advantages of this strategy is that the computational cost is very small, since the selection of features is performed once, before the learning process.

Contrasting the methodology described above, the adaptive selection strategy chooses the best feature of input (attribute) for each node of the tree, regardless the level where the node is. For each node is chosen the best input using only the subset associated with this node. This strategy generates more compact neuro-fuzzy BSP structures according to the specialization of each node, resulting in better generalization performance. However, the computational cost is higher, since the selection algorithm must be run for each new node of NFHB-Class model.

2.5. Fuzzy Rule Extraction

As the NFHB^{-1} model (Gonçalves *et al.* 2006), the NFHB-Class model (Gonçalves *et al.* 2009) is capable of generating interpretable rules for purpose of extracting information from a specific database. The rules extracted in this model are of the following type: If *x* is high and *y* is small… and *w* is hot, then class *k*.

Figure 9 show an example of NFHB-Class model without the connections of leaves cells with the T-conorm neurons. In this approach, each partition of the input space (leaf node) will have an associated rule. The elements of each partitioning are associated to all existing k classes, with different membership levels.

To evaluate rules were used the fuzzy accuracy and coverage (Gonçalves *et al.* 2006).

2.5.1. Fuzzy Accuracy

The accuracy of a rule measures how well it is applied to the data (Gonçalves *et al.* 2006). In order to determine how suitable a particular fuzzy rule describes a specific class k, the fuzzy accuracy measurement was described in Eq. (12).

$$Fuzzy\_Acurracy_k^i = \frac{\sum_{j=1}^{P_k} \alpha_{k,j}^i}{\sum_{j=1}^{P_i} \alpha_j^i} \tag{12}$$

Where $Fuzzy\_Acurracy_k^i$ is the accuracy of the rule for class *k* in partition *i*; $\alpha_{k,j}^i$ is the membership level of pattern *j* of class *k* in partition *i*; α_j^i is the membership level of pattern *j* in partition *i* (regardless of the class); P_k is the total number of patterns of class *k*; and P_i is the total number of patterns in partition *i*. More details can be found in Gonçalves *et al.* (2006).

2.5.2. Fuzzy Coverage

Fuzzy coverage supplies a measure of how comprehensive a rule is in relation to the total number of patterns in the rule base, i.e., it measures “how many” patterns are affected by the rule at hand. The definition of fuzzy coverage is given by Eq. (13).

$$Fuzzy^i\_Coverage = \frac{\sum_{j=1}^{P_i} \alpha_j^i}{P} \tag{13}$$

where $Fuzzy^i\_Coverage$ = fuzzy coverage of partition *i*; *P* = total number of patterns in the database; α_j^i is the membership level of the pattern *j* in the partition *i*; and P_i is the number of patterns in partition *i*. More details can be found in Gonçalves *et al.* (2006).

3. NEURAL NETWORKS

The model of an artificial neuron is shown in Fig. 6. The artificial neuron consists of *m* inputs weighted by weights w_{kl} , ..., w_{km} then added a junction additive. These weights are known as synaptic weights and are responsible for the ability to model and learn of a neural network. An activation function is placed below the sum; this function is known as restrictive (Haykin 2001), which in general means that it limits the range of permissive output neuron. The model also provides a setting called bias that has the effect of increasing or decreasing the value applied to the activation function.

In mathematical terms we can define the operation of the neuron *k*, Fig. 6, which has the entries *j (j=1,2,....m)* by Eq. (14), Eq. (15) and Eq. (16).

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{14}$$

$$v_k = u_k + b_k \tag{15}$$

$$y_k = \varphi(v_k) \tag{16}$$

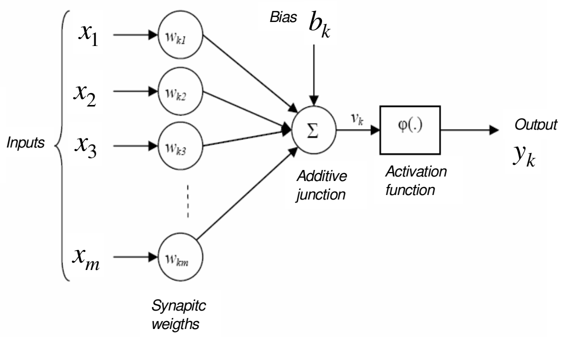


Figure 6. Nonlinear neuron model.

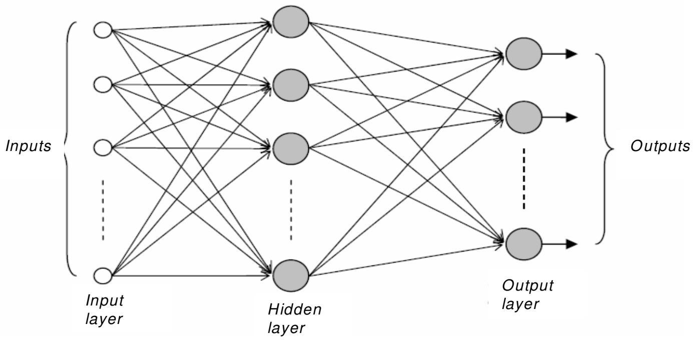


Figure 7. Architecture of an MLP-type neural network with one hidden layer.

The activation function $\varphi(.)$ is responsible for defining the output of a neuron in terms of an input v. These functions can be represented in many ways. More commonly, and in this work, we use sigmoid functions (Haykin 2001).

Once defined the character of a neuron, can be defined the architecture of a neural network linking each other neurons through synaptic weights. The configuration presented in this work is that of a neural network with multiple layers known as MLP (multilayer perceptrons). This configuration is illustrated in Fig. 7.

In general, in the task of pattern classification, the neural network architecture receives attributes predictive as input of the first layer of the neural network. The objective attributes (classes) are modeled by the output layer of the neural network. Thus the algorithm can estimate how much the desired output is far from the actual output. More details can be found in Freeman and Skapura (1992).

4. TEXTURE

According to Turceyan and Jain (1993), image texture is a combination of similar patterns with a regular frequency. It is an attribute that represents a spatial arrangement of pixels in a region (IEEE Standard 610.4, 1990). Jain (1998) defines texture as a repetition of basic patterns in space. Conci *et al.* (2008) refer to the texture as a visual pattern that has some properties of homogeneity that does not result simply in a color or intensity; it can be defined as the visual appearance of a surface. Although there is no precise definition for texture, it is easily perceived by human vision due to its range of visual patterns composed of sub-patterns, which have underlined properties, such as uniformity, density, roughness, regularity, intensity (Haralick *et a*l. 1973).

Textures can be defined as patterns of repetition of certain formed elements, called texels. Texel can be defined as the smallest area of digital image that make up a distinct texture. The size of the region of an image seen can not be very small compared to the basic element. Thus the texture can be characterized by a repeating pattern (texton or Texel) on a region. This model can be repeated accurately or with variations (random or not random). Size, shape, color and orientation of the elements of the model may vary over the regions, characterizing a pattern diversity of a texture (Conci *et al.* 2008).

4.1. Spacial Variation Coefficient - SVC

This coefficient quantifies the spatial characteristic of texture by means of statistical measures that describe the spatial variations of intensity or color. Two measures of the pixels belonging to the region of the image are used: a measure of position (average) and a measure of dispersion (standard deviation) (Conci *et al.* 2008).

The SVC takes into account not only the distribution of intensities, but also their spatial distribution through the use of classes of distances. To better describe the data dispersion in terms of its value, we can use the coefficient of variation given by Eq. (17):

$$CV = \frac{\sigma}{x}.100 \tag{17}$$

where σ is the standard deviation.

After obtaining the mean and coefficient of variation for each class of distance measures for the position and dispersion are combined using Eq. (18), whose unique value (SCV Class) preserves the information obtained on both measures.

$$CVE = \frac{arctg\left(\frac{\bar{x}}{CV}\right)\pi}{180}\sqrt{\bar{x} + CV^2} \tag{18}$$

For color images, we must obtain the mean and coefficient of variation for the SCV of each class of distances on. Finally, the mean and coefficient of variation of SCV for each class are combined again by Eq. (18), for each band (R, G, B), resulting in the SCV of the color texture region. More details can be found in Conci *et al.* 2008.

4.1. Hurst Coefficient

The Hurst coefficient, Eq. (19), is described by some authors as an approximation of the fractal dimension for images in gray levels (Parker 1997).

$$D = \frac{\ln N}{\ln\left(\frac{1}{r}\right)} \tag{19}$$

It is used in the calculation of the Eq. (19), the intensity of a set of pixels of an image I divided into N shares coincident not identical and staggered by scale factor r. Major details can be found in Conci *et al.* 2008.

4.2. Entropy

The entropy of an image can be defined as a characteristic of texture which measures its randomness, that is, the greater this number is, more irregular is the image analyzed. The texture entropy is given by Eq. (20).

$$E = \sum_{i=0}^{M-1}\left(p_i \log_2\left(\frac{1}{p_i}\right)\right) \tag{20}$$

where M is the total number of different textures in the image and p_i is the probability that the *i-th* stored texture is used again.

4.2. Co-occurrence Matrix

The second-order statistic is calculated by the occurrence probability of a given pair of gray levels *i* and *j* at a certain distance δ and θ direction, i.e. the co-occurrence matrix may be described as a two-dimensional histogram that provides the occurrence frequency of *P (i, j, δ, θ).*

The relationship between the pair of levels of gray *i* and *j* is performed in four directions: 0°, 45°, 90° and 135° from the central pixel X, considering its eight neighbors (Fig. 8).The distances are chosen according to the image granularity of the manipulated images.

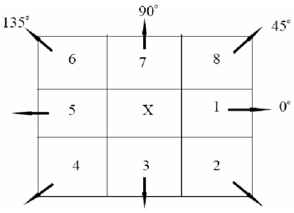


Figure 8. The four directions of θ: 0, 45, 90 and 135 degrees from the central pixel X.

The co-occurrence matrix has the dimension (Nc x Nc), where Nc is the number of gray levels. The matrix is ordered from the lowest gray level to the highest, considering line and column. Thus each position of the matrix stores the probability of occurrence *P (i, j, δ, θ)* of the color of the line *i* with the color of column *j*.

For each direction and each distance, it generates an array of co-occurrence. Haralick has shown that these four directions 0°, 45°, 90° and 135°, produce four different co-occurrence matrices that should be combined to form the final co-occurrence matrix (Haralick *et a*l. 1973).

The textural information is characterized by the matrix of relative frequency *P (i, j, δ, θ)*, which indicates the probability of two pixels occurrence (*i* and *j*), a separated by distance δ and angle θ (Conci *et al.* 2008).

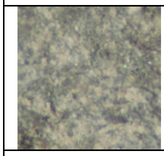
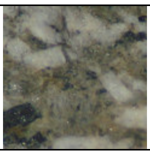
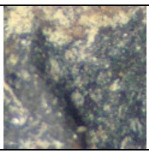
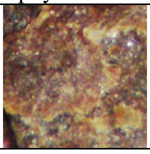
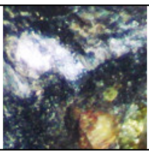
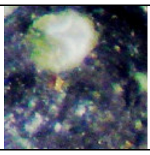
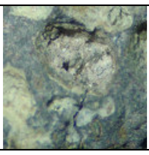
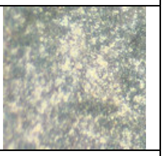
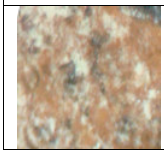
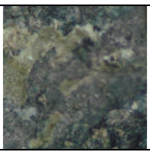
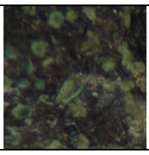
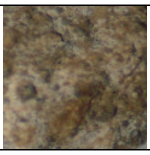
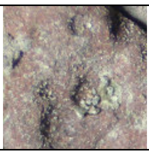
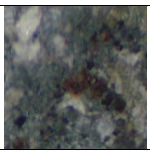
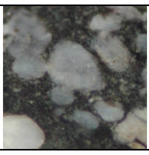
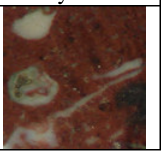
In general, the problem is to analyze a set of co-occurrence matrices that characterize the texture, using one or more descriptors. Haralick *et al.* (1973) proposed a set of 14 texture descriptors.

5. CASE STUDY

To evaluate the system performance we used 50 RGB images (401x401), for each rock classes and subclasses, thus producing a total of 800 images. The classes and the subclass of igneous rocks that make up the image database are: gneiss (granite gneiss and leucocratic gneiss), basalt (aphanitic aphyric basalt, oxidized aphanitic aphyric basalt, porfiritic basalt and amygdaloid porfiritic basalt), diabase (altered amygdaloid diabase, equigranular diabase, porfiritic diabase, altered porfiritic diabase and amygdaloid porfiritic diabase), rhyolite (altered rhyolite, amygdaloid rhyolite, porfiritic rhyolite, amygdaloid porfiritic rhyolite and venulated amygdaloid porfiritic rhyolite).

For each image were extracted: Hurst coefficient for gray and color images (a coefficient for each RGB channel); spatial variation coefficient (gray and color); entropy and co-occurrence matrix. From the 160 co-occurrence matrix, we computed the matrices average in the directions 0^o, 45^o, 90^o and 135^o for each distance, resulting in 40 matrices for the 40 distances. Analyzing these 40 matrices, we used the following descriptors: contrast, homogeneity, energy, entropy and correlation. We construct 5 curves for each image, and the higher value and the area were used as attributes to determine the texture of the images. A sample rock example can be seen in Tab. 1.

Table 1. Rocks samples.

granite gneiss	leucocratic gneiss	aphanitic aphyric basalt	oxidized aphanitic aphyric basalt	porfiritic basalt	amygdaloid porfiritic basalt	altered amygdaloid diabase	equigranular diabase
							
porfiritic diabase	altered porfiritic diabase	amygdaloid porfiritic diabase	altered rhyolite	amygdaloid rhyolite	porfiritic rhyolite	amygdaloid porfiritic rhyolite	venulated amygdaloid porfiritic rhyolite
							

In all tests were used 50% of the database to train the NFHB-Class model and the neural network and 50% of the database to validate them. Table 2 summarizes the classification results for the NFHB-Class models and for the neural networks.

To illustrate the tree structure found by the model NFHB-Class was chosen the test performed with the gneiss rock, using adaptive strategy selecting of features, where the success in the training set was 100% and hit the set of validation was 98%, as can be seen in Fig. 9. Figure 9 shows the structure founded by the NFHB-Class model. The T-conorms connections are not shown, because they hinder the understanding process of the rules extracting.

In the Fig. 9 the attributes are encoded by: X2 – Hurst coefficient for Red channel, X3 – Hurst Coefficient for Green channel, X4 – Hurst coefficient for Blue channel, X6 – CVE for Red channel, X7 – CVE for Green channel, X8 – CVE for Blue channel, X10 – Entropy of the Image for Red Channel, X11 – Entropy of the Image for Green Channel, X12 – Entropy of Image for Blue channel.

Through the path in the tree (Fig. 9) is possible to extract rules that describe the database of the gneiss rock. Below is listed some rules extracted from Fig. 9.

Rule1: If X8 is low and if X12 is low and if X6 is low then Class = 1. [Accuracy: 0.6813 / Coverage: 0.1105]
Rule2: If X8 is low and if X12 is low and if X6 is high and if X3 is low then Class = 1. [Accuracy: 0.548 / Coverage: 0.06137]
Rule3: If X8 is low and if X12 is low and if X6 is high and if X3 is high then Class = 1. [Accuracy: 0.5152 / Coverage: 0.05707]

Table 2. Classification results with the NFHB-Class model and neural network.

Rock	Model	Training set	Validation set	Number of generated rules
Gneiss	*NFHB-Class*[1]	100 %	98 %	52
	NFHB-Class[2]	100 %	98 %	12
	Neural network	100 %	96 %	
Basalt	*NFHB-Class*[1]	95%	87%	81
	NFHB-Class[2]	88%	84%	25
	Neural network	100%	86%	
Diabase	*NFHB-Class*[1]	81.6%	71.2%	110
	NFHB-Class[2]	92%	73.6%	83
	Neural network	93.6%	69.6%	
Rhyolite	*NFHB-Class*[1]	93.6%	78.4 %	225
	NFHB-Class[2]	96%	78.4 %	56
	Neural network	97.6%	75.2%	

[1] Strategy for fixed selection of feature sets.

[2] Strategy for adaptive characteristics selection.

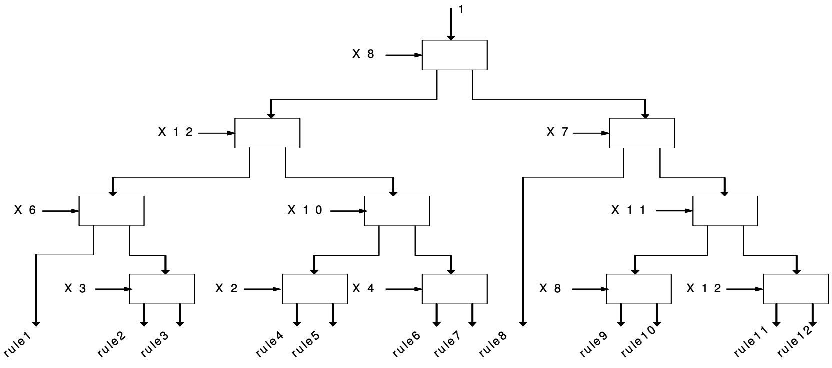


Figure 9. Tree structure of NFHB-Class model, without T-conorm connections, with adaptive selection strategy for the test performed with the rock gneiss.

6. CONCLUSIONS

This paper presents the use of NFHB-Class model to classify macroscopic image of rocks. This approach creates its own architecture, thus creating automatically their rule base. Two strategies for characteristic selection in the database were adopted; fixed and adaptive. Using the algorithm embedded in the model of Jang, it is not necessary to use principal component analysis to determine the best combination of attributes which is more representative for the rock textures.

The results with the NFHB-Class model, to the classification task, obtained more than 73% of accuracy in the validation set for all classes of rocks, which indicates the great potential of the model to this aim.

For the gneiss rock class, the greatest hit for the four databases tested was 98% using the validation set. For the basalt rock class the result was 87%. For the diabase rock class the best result for all databases tested was 73.6% in the validation set. For the rhyolites rock class we obtained 78.4%.

All tests performed with the NFHB-Class model were repeated with neural networks for comparison purposes. In all cases, the model NFHB-Class obtained better validation results than the neural networks.

One of the advantages of NFHB-Class model is the fact that it generates fuzzy rules that explain the extraction of knowledge, i.e., it is possible to have a very satisfactory rating. Therefore it presents a classification explanation, which does not happen if using a neural network or a committee of neural networks for a classification task.

7. ACKNOWLEDGEMENTS

The authors would like to acknowledge FAPERJ (E-26/171.362/2001, E-26/110.992/2008) for the financial support.

8. REFERENCES

Aoyama, A. and Walsh, S.P.K., 1997, "Nonlinear Modelling of LPG %C5 content of Catalytic Reformer Debutanizer Column", Computers chem. Eng. Vol. 21, Suppl., pp. S1155-S1160.

Barret, M.B., Chan, T. and Demmel, J., 1994, "Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods", <http://www.netlib.org>.

Blum, A.L. and Langley, P., 1997, "Selection of Relevant Features and Examples in Machine Learning", Artificial Intelligence, 97:245-271.

Conci, A., Azevedo, E. and Leta, F.R., 2008, "Computação Gráfica – Teoria e Prática", Volume 2. Editora Campus.

Dash, M. and Lui, H., 1997, "Dimensionality Reduction for Unsupervised Data", Nineth IEEE International Conference on Tools with Artificial Intelligence, pp. 532-539.

Dong, D. and McAvoy, T.J., 1996, "Nonlinear Principal Component Analysis Based on Principal Curves and Neural Networks", Comp. Chem. Engng. 20, pp. 65-78.

Freeman, J.A. and Skapura, D.M., 1992, "Neural Networks: Algorithms, Applications and Programing Techiniques", Addison-Wesley, Reading, MA.

Gonçalves, L.B., Leta, F.R. and Valente, S.C., 2009, "Macroscopic Rock Texture Image Classification using an Hierarchical Neuro-Fuzzy System", 16th International Conference on Systems, Signals and Image Processing, IWSSIP, Chalkida, Greece, June, accepted.

Gonçalves, L.B., 2001, "Modelos Neuro-Fuzzy Hierárquicos BSP para Classificação de Padrões e Exração de Regras Fuzzy em Banco de Dados", Dissertação de Mestrado. Departamento de Engenharia Elétrica. PUC-Rio, Brazil.

Gonçalves, L.B., Vellasco, M.M.B.R., Pacheco M.A.C. and Souza, F.J., 2006, "Inverted hierarchical neuro-fuzzy BSP system: a novel neuro-fuzzy model for pattern classification and rule extraction in databases", IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Volume: 36, Issue: 2, pp. 236-248.

Haralick, R.M., Shanmugan, K. and Dinstein, I., 1973, "Textural Features for Image Classification", IEEE Transactions on Systems, Man and Cybernetics, SMC-3(6): p. 610-621.

Haykin, S., 2001, "Redes Neurais, princípios e prática". ARTMED Editora LTDA, Porto Alegre, RS, Brazil.

IEEE Standard 610.4, 1990. IEEE "Standard Glossary of Image Processing and Pattern Recognition Terminology," IEEE Press, New York.

Jain, A.K., 1998, "Fundamentals of Image Processing," Prentice-Hall, New York.

Jang, J.S.R., 1994, "Structure Determination in Fuzzy Modeling: A Fuzzy Cart Approach," Proceedings of IEEE International Conference on Fuzzy Systems. Orlando.

Lanas, A.I., 2000, "Sistemas Neuro-Fuzzy Hierárquicos BSP para Previsão e Extração de Regras Fuzzy em Aplicações de Mineração de Dados," Tese de Mestrado. DEE-Puc-Rio,Brazil.

Langley, P., 1994, "Selection of Relevant Features in Machine Learning," In Proccedings of the AAAI Fall Symposium on Relevance. AIII Pres.

Parker, J.R., 1997, "Algorithms fo Image Processing and Computer Vision", John Wiley & Sons, Toronto, 432 p.

Roffel, J.J., MacGregor, J.F. and Hoffman, T.W., 1989, "The Design and Implementation of a Multivariable Internal Model Controller for a Continuous Polybutadiene Polymerization Train", IFAC Dynamics and Control Chemical Reactors, Maastricht.

Santen, A., Koot, G.L.M. and Zullo, L.C., 1997, "Statistical Data Analysis of a Chemical Plant", Computers Chem. Engng, Vol 21, Suppl., pp.S1123-S1129.

Souza , F.J., 1999, "Modelos Neuro-Fuzzy Hierárquicos", Tese de Doutorado. DEE. PUC-Rio, Brazil.

Talavera, L., 1999, "Feature Selection as Retrospective Pruning in Hierarchical Clustering" , In Third International Symposium on Intelligent Data Analysis. Amsterdam, The Netherlands: Springer Verlag.

Turceyan, M. and Jain, A.K., 1993, "Handbook of Pattern Recognition and Computer Vision", chapter Texture Analysis, pages 235–276. World Scientific Publishing Company.

Yager, R.R. and Filev, D.P., 1994, "Template-based Fuzzy Systems Modeling" , Journal Intell. and Fuzzy Syst., vol. 2, pp. 39-54.

Yang, J. and Honavar, V., 1998, "Feature Subset Selection Using a Genetic Algorithm". IEEE Intelligent Systems.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.