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. *In this paper we present 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 oil reservoir exploitation. The same approach can be also applied to metals, in order to classify different materials types based on their grain texture. We present an image classification for macroscopic rocks, based on these texture descriptors and on a neuro-fuzzy methodology. We also performed a neural network to classify these rocks, and the results obtained by both approaches were compared. To evaluate the system performance we used 50 RGB images, for each rock classes and subclasses, thus producing a total of 800 images. The rock classes tested are: gneiss (two classes), basalt (four classes), diabase (five classes), and rhyolite (five classes), they are identified as igneous rock. For each image were extracted the following texture features: Hurst coefficient for gray and color images (one coefficient for each RGB channel); spatial variation coefficient (considering gray and color images); entropy and co-occurrence matrix. The tests performed converged to optimum solution taking into account the fuzzy extraction rules good performance.*

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

1. INTRODUCTION

Nowadays, oil and gas are an essential energy resource for industry development. They can be found in a variety of geological environments. Their exploitation is a large scale activity and the use of expert knowledge is primary to decision making. Two sets of information are important in the exploitation of a new oilfield: the reservoir geometry and the porous rock identification.

Analyzing the oil reservoir geometry it is possible to identify the oil quantity inside the reservoir. The second information set consists in describing the porous rock that holds the oil, that is named reservoir rock. The reservoir quality is affected by the rock particular characteristics, such as: the rock formation minerals; the pores volume and shape (spaces that preserve fluids within the rock); the connections between the pores and the physical-chemical processes, that may have modified its characteristics.

Reservoir rocks study is based on a systematic description of rock samples that are collected from oil exploration boreholes, this is called petrography. The petrography is an activity performed in the laboratory, which incorporates 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 0.03 mm thin sections, which are analyzed using optical microscopes with polarized light.

In macroscopic analyses a rock is described based on a several physical characteristics such as color, structure, texture, grain size and orientation and existing fossil. There is a large number of rock classes and subclasses. The classification task is not simple, requiring long training time. It strongly depends on features identification based on image analysis.

This paper presents a Neuro-Fuzzy Hierarchical model for rock texture image automatic classification, called NFHB-Class. We also present a neural networks to solve the same problem to establish a comparison between both methodologies. The systems use 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 and NFHB-Class model is the system structure obtaining. 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 to generate its own structure, thus avoiding the NFHB model use.

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 NFHB-Class cell generates two precise outputs (crisp) after a defuzzification process.

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

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

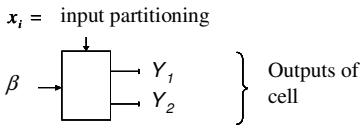


Figure 1. NFHB-Class cell.

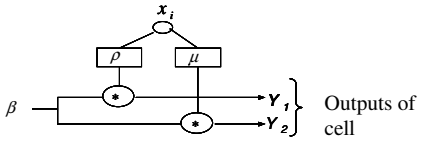


Figure 2. NFHB-Class cell schematic symbol.

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 input cell: in this case β = 1. The value '1' in the first input cell represents the entire input space, i.e. the entire discussion universe of the variable x_i that is being used as an input cell.
- The output of a previous level: in this case $\beta = y_i$, 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 defuzzification procedure simplification 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 can be read in (Gonçalves 2001).

2.2. NFHB-Class Architecture

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

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

The outputs of the leaf cells of the system are listed below:

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

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

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

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

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

After calculating each leaf cell output, the leaf cells connection is performed considering the T-conorms neurons. Each T-conorm neuron is associated to a specific class, as can be seen in the example described in Fig. 3, where there are three distinct classes, and consequently, there are three T-conorms neurons.

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

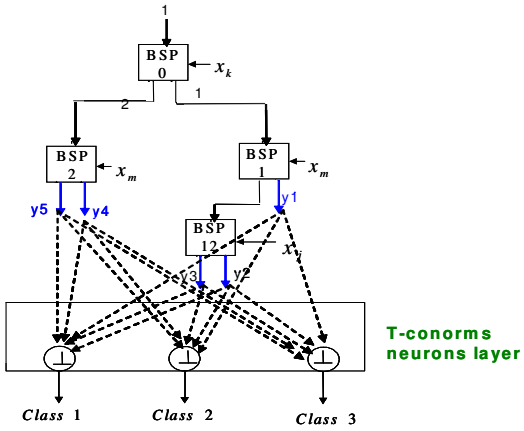


Figure 3. NFHB-Class architecture.

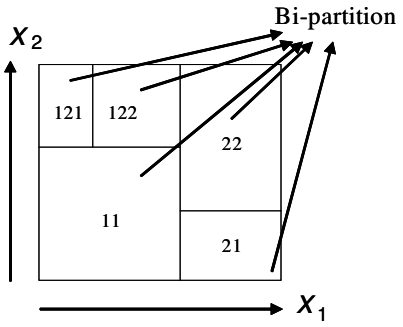


Figure 4. Input space partitioning of NFHB-Class.

After defining the strategy of how leaf cells were connected to T-conorms neurons and theirs weights of these links, it is necessary to define which T-conorms operators will be used to obtain the final output. All leaf cells outputs are connected to all T-conorms neurons, as shown in Fig. 3. Each output is multiplied by the weight of its connection with the T-conorm neurons. The next step is to define the treatment of all T-conorms neurons inputs that are derived from the leaf cells.

In the example, the three T-conorm neurons outputs are calculated according to 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 leaf cells outputs; 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 neuron output.

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 consists in the *max* operator that only takes the maximum membership value into account, ignoring the inputs membership values.

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

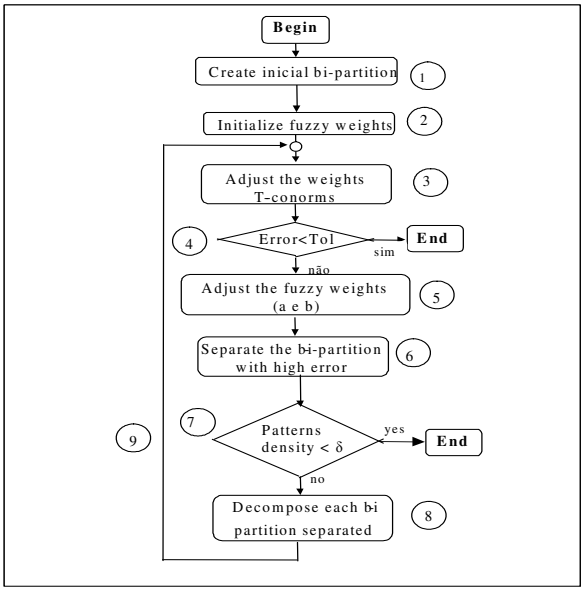


Figure 5. Learning algorithm of NFHB-Class model.

2.4. Strategies for Attributes Selection - NFHB-Class System

In pattern classification problems it is important to define the goals and selection method, i.e. in the available database it is necessary to identify which characteristics should 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. This approach results in a better generalization, a small number of rules and a better interpretation degree. Two methods for characteristics selection were tested: the Jang algorithm (Barret *et al.* 1994) which presented better performance, and the entropy method (Yager and Filev, 1994). Besides those methods, there are several studies using other features selection techniques, 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 at 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. The strategy advantage is the small computational cost, since the feature selection is performed one time, before the learning process.

Contrasting the methodology described above, the adaptive selection strategy chooses the best input feature (attribute) for each tree node, 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. It results in a better generalization performance. However, the computational cost is higher, since the selection algorithm must be run for each new NFHB-Class model node.

2.5. Fuzzy Rule Extraction

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

Figure 9 shows a NFHB-Class model example without the leaf cells connections with T-conorm neurons. In this approach, each input space partition (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 are used the fuzzy accuracy and coverage (Gonçalves *et al*. 2006).

2.5.1. Fuzzy Accuracy

Rule accuracy measures how well it is applied to the data (Gonçalves *et al*. 2006). It determines how suitable a particular fuzzy rule describes a specific k class.

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

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

2.5.2. Fuzzy Coverage

Fuzzy coverage means 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 available rule. The fuzzy coverage definition is given by Eq. (13).

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

where $Fuzzy^i\_Coverage$ is *i* partition fuzzy coverage; *P* is the total patterns number in the database; α_j^i is the pattern *j* membership level in *i* partition; and P_i is the number of patterns in *i* partition. 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 in *m* inputs weighted by w_{kl} , ..., w_{km} then added a junction additive. They are known as synaptic weights and are responsible for the neural network modeling and learning ability. 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 set called bias that increases or decreases the value applied to the activation function.

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

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

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

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

The $\varphi(.)$ activation function is responsible for defining the neuron output in terms of an input *v*. These functions can be represented in many ways. The most common manner is the sigmoid functions use (Haykin, 2001).

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

In general, in pattern classification task, the neural network architecture receives attributes predictive as the neural network first layer input. The objective attributes (classes) are modeled by the neural network output layer. 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).

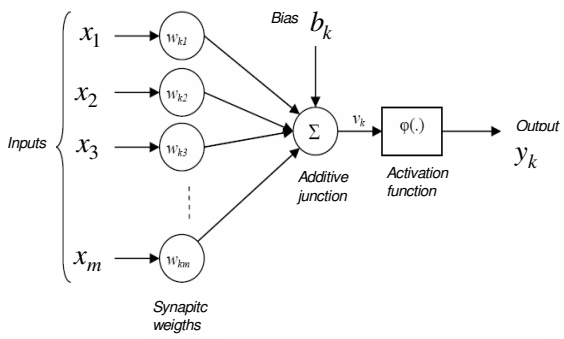


Figure 6. Nonlinear neuron model.

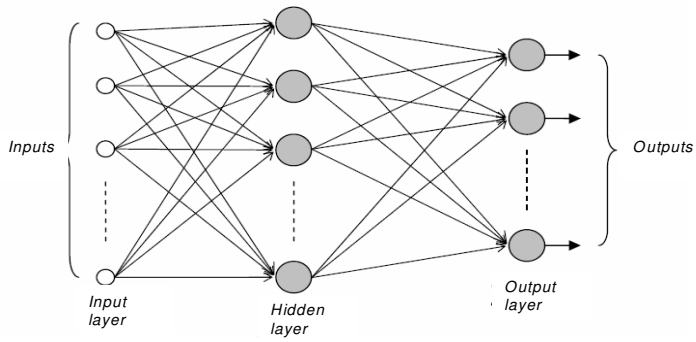


Figure 7. Architecture of an MLP-type neural network with one hidden layer.

4. TEXTURE

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

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

4.1. Rock texture – related works

Lepistö *et al.* (2003) presented a classification method based on rocks structural and spectral characteristics. Launeau *et al*. (1994) extract features that could identify the texture considering as spectral feature some color parameters. To define the structural aspect they used co-occurrence matrices. For rock classification task the non-homogeneous textures were divided into blocks. Lepistö *et al*. (2005) used Gabor filter in rock color images for classification. Autio *et al*. (1999) used co-occurrence matrices and Hough coefficients for rock classification. Starkey and Samantary (1993) used morphology and color parameters to distinguish rock images. Genetic programming and edge detector techniques were also used by Ross *et al*. (2000) and Starkey and Samantary (1993) in petrography. Genetic programming with decision trees was used for grain segmentation by Ross *et al*. (2001). Thompson *et al*. (2001) used three layers of neural networks for minerals classification. Fueten and Manson (2007) used neural networks for detecting edges in petrography color images to threshold the grains.

4.2. Spacial Variation Coefficient - SVC

This coefficient quantifies the texture spatial characteristic taking into account statistical measures that describe the intensity or color spatial variations. There are two measures obtained from the pixels that belong to a region: position measure (average) and dispersion measure (standard deviation) (Conci *et al.* 2008).

The SVC considerers not only intensity distribution, but also their spatial distribution. To describe the data dispersion in terms of its value, we can use the variation coefficient given by Eq. (17):

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

where σ is the standard deviation.

After obtaining the mean and variation coefficient for each distance measures class, position and dispersion are combined using Eq. (18), whose unique value (SVCClass) 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 variation coefficient for each distance class. Finally, SVC mean and variation coefficient for each class are combined again using Eq. (18), for each band (R, G, B), resulting in the color texture region SVC. More details can be found in Conci *et al.* 2008.

4.3. Hurst Coefficient

Hurst coefficient, Eq. (19), is described as a fractal dimension approximation for images in gray levels (Parker, 1997).

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

The intensity of an image set of pixels (I) is divided into N coincident not identical shares and staggered by scale factor r. Major details can be found in Conci *et al.* (2008).

4.4. Entropy

An image entropy can be defined as a texture characteristic which measures its randomness, i.e. the greater this number is, more irregular is the analyzed image. 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 different textures total number in the image and p_i is the probability that the *i-th* stored texture is used again.

4.5. Co-occurrence Matrix

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

The relationship between the gray level pair is performed in four directions: 0^o, 45^o, 90^o and 135^o from the central pixel X, considering its eight neighbors (Fig. 8).The distances are chosen according to the image granularity.

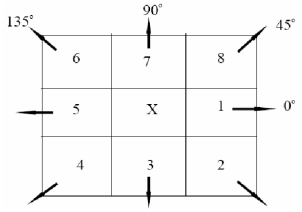


Figure 8. The four directions of θ: 0, 45, 90 and 135 degrees from the central pixel X.

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^o, 45^o, 90^o and 135^o, 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).

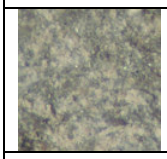
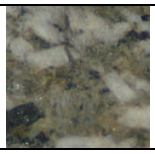
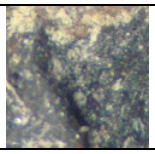
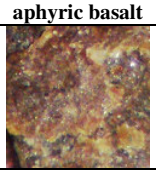
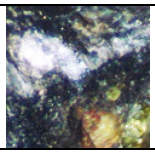
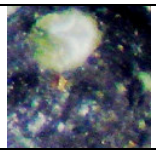
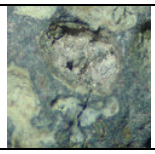
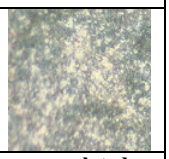
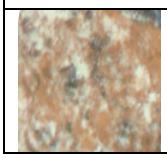
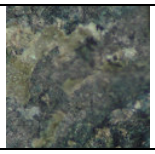
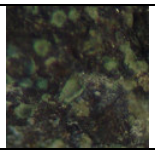
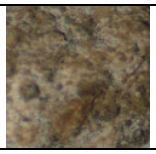
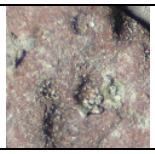
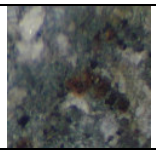
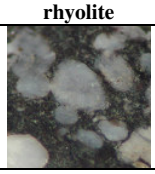
In general, the problem is to analyze a set of co-occurrence matrices that characterize the texture, using one or more descriptors. Haralick *et al.* (1973) proposed a set of 14 texture descriptors.

5. CASE STUDY

To evaluate the system performance we used 50 RGB images (401x401), for each rock classes and subclasses, thus producing 800 images. The igneous rocks classes and subclass 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 create 5 curves for each image, and the highest value and the area were used as attributes to determine the image texture. Sample rocks examples 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.

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.

To illustrate the tree structure found by the model NFHB-Class we choose the gneiss rock performed test, using adaptive strategy features selection. In this case the success in the training set was 100% and the validation set hit was 98%. Figure 9 shows the structure created by the NFHB-Class model. The T-conorms connections are not shown, because they hinder the rules extracting understanding process.

In 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 (A): 0.6813 / Coverage (C): 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. [A: 0.548 / C: 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. [A: 0.5152 / C: 0.05707]

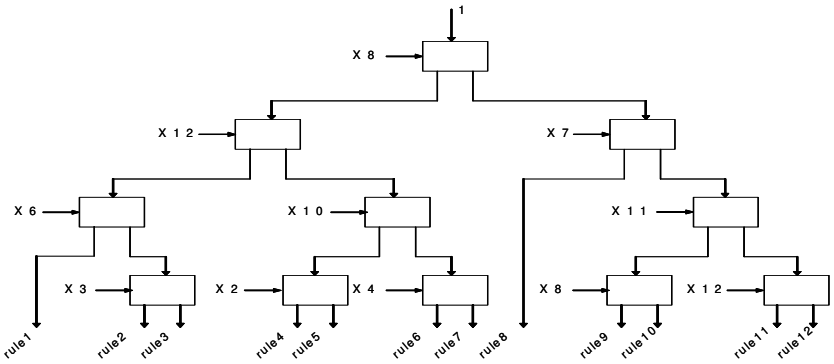


Figure 9. Tree structure of NFHB-Class model, without T-conorm connections, with adaptive selection strategy for the test performed with the rock gneiss.

6. CONCLUSIONS

This paper presents some approaches to solve macroscopic rock texture recognition problem. We used Artificial Neural Network and NFHB-Class model to classify macroscopic rock images. The NFHB-Class approach creates its own architecture, thus automatically generating their base rule. 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 attributes combination which is more representative for rock textures.

The results using the NFHB-Class model were more than 73% of accuracy in the validation set for all rock classes, which indicates the great model potential 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 got better validation results than artificial neural networks method.

One of the NFHB-Class model advantages is the fact that it generates fuzzy rules that explain the knowledge extraction, i.e., it is possible to have a very suitable rating. Therefore it presents a classification explanation, which does not happen when we use an artificial neural network or a neural network committee 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.

Autio, J., Luukkanen, S., Rantanen, L., Visa, A., 1999. “The Classification and Characterisation of Rock Using Texture Analysis by Co-occurrence Matrices and the Hough Transform,” International Symposium on Imaging Applications in Geology, pp. 5-8, Belgium.

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, p. 245-271.

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

Dash, M. and Lui, H., 1997, "Dimensionality Reduction for Unsupervised Data". In 9th IEEE International Conference on Tools with AI.

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

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

Fueten, F. and Mason, J., 2007, "An artificial neural net assisted approach to editing edges in petrographic images collected with the rotating polarizer stage". Computers & Geosciences Vol. 33 , Issue 9, pp. 1176-1188.

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, Vol. 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): pp. 610-621.

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

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.

Launeau, P., Cruden, C. A., Bouchez, J.L., 1994, "Mineral recognition in digital images of rocks: a new approach using multichannel classification", Can Mineral, Vol. 32, pp. 919–933.

Lepistö, L., Kunttu, I., Autio, J., Visa, A., 2003, "Rock Image Classification Using Non-Homogenous Textures and Spectral Imaging", Spectral Imaging, WSCG SHORT PAPERS proceedings, WSCG'2003, Plzen, Czech Republic.

Lepistö, L., Kunttu, I., Visa, A., 2005, "Rock image classification using color features in Gabor space". Journal of Electronic Imaging, - Vol. 14, Issue 4, 040503.

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.

Ross, B. J., Fueten, F., Yashkir, D. Y., 2000. "Edge Detection of Petrographic Images Using Genetic Programming". Proceedings of the Genetic and Evolutionary Computation Conference, GECCO.

Ross, B. J., Fueten, F., Yashkir, D. Y., 2001, "Automatic Mineral Identification Using Genetic Programming", Machine Vision and Applications.

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.

Starkey, J., Samantary, A. K., 1993, "Edge detection in petrographic images". J. Microsc., Vol. 172, pp. 263–266.

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.

Thompson, S., Fueten, F., Bockus, D., 2001, "Mineral identification using artificial neural networks and the rotating polarizer stage". Computers & Geosciences Vol. 27, Issue 9, pp. 1081-1089.

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

Yager, R.R. and Filev, D.P., 1994, "Template-based Fuzzy Systems Modeling". J. Int. and Fuzzy Sys., Vol.2, pp. 39-54.

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

9. RESPONSIBILITY NOTICE

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