A MODULAR NEURAL ARCHITECTURE FOR IMAGE CLASSIFICATION USING KOHONEN FEATURE EXTRACTION

Márcio L. Gonçalves[1]
Márcio L. de Andrade Netto[2]
Jurandir Zullo Júnior[3]
[1] DCA/FEE/UNICAMP - Cidade Universitária "Zeferino Vaz" C.P. 6101 13081-970 Campinas - SP - Brasil
Email: mleandro@dca.fee.unicamp.br
[2] DCA/FEE/UNICAMP Email: marcio@dca.fee.unicamp.br
[3] CEPAGRI/UNICAMP Email: jurandir@cpa.unicamp.br

Commission II, Working Group 3

KEY WORDS: Classification, Extraction, Algorithms, Neural, Performance, Image, Multispectral

ABSTRACT

This work presents an architecture for Remote Sensing (RS) multispectral image classification based on Artificial Neural Networks (ANN), aiming at two objectives, namely: searching of techniques for improving the performance in the classification task and to exploit the advantages of unsupervised learning for feature extraction. The architecture is divided in two modules: *feature extraction* by the Kohonen Self -Organizing Map (SOM) and *classification* by a Multilayer Perceptron (MLP) network, trained by a learning algorithm which uses 2nd-order information exactly calculated. To evaluate the efficiency of this classification scheme, a comparative analysis with the maximum likelihood algorithm, conventionally used for RS multispectral images classification, is realized.

KURZFASSUNG

Diese Arbeit legt ein Schema für die Klassifikation von mehrspektralen Bilder aus Fernerkundung vor, auf der Basis der künstlichen Neuronalen Netze. Die Ziele waren folgende: Untersuchung von Methoden zur Leistungserhöhung der Klassifizierung; und die Ausnutzung der Vorteile des unüberwachten Lernen für die Merkmalextraktion. Das Schema ist in zwei Phasen unterteilt: Merkmalextraktion durch die selbstorganisierende Karte Kohonens und Klassifikation durch das *Multilayer Perceptron* Netz mit einem Lernverfahren, das exakt berechnete Information 2.Ordnungs nutzt. Um die Effizienz des vorgeschlagenen Klassifikationschema zu bewerten, wurde eine Vergleichung mit dem statistischen *Maximum Likelihood* Algorithmus durchgeführt.

1. INTRODUCTION

Since the resurgence of interest in the middle eighties, Artificial Neural Networks (ANN) have shown its efficacy and versatility in a wide range of applications. In particular, successful applications in Remote Sensing (RS) image classification have already been reported (Benediktsson et al. 1990, Hepner et al. 1990, Kanellopoulos et al. 1992, Schlünzen 1993), showing superior results to those of conventional statistical classification approaches.

Among the advantages of ANN over conventional statistical methods one can point out the nonnecessity of *a priori* knowledge of the probabilistic data model, since ANN have the ability to learn the data distribution properties during the training phase (Benediktsson et al. 1990), as well as the ability to generalize and to incorporate nonstatistical information and knowledge that may be potentially valuable.

The most relevant restrictions on a broader utilization of ANN refer basically to its present performance limitations, due to the slow convergence of standard *backpropagation* training algorithm (Rumelhart et al. 1986), that is normally used, and to the amount of adjustments to the training parameters (Key et al. 1989, Benediktsson et al. 1990, Hepner et al. 1990, Liu et al. 1991, Kanellopoulos et al. 1992, Schlünzen 1993).

To improve the performance of ANN techniques this work investigates the following approaches: parallel implementation of the training algorithms in a multiprocessing environment and utilization of an advanced training algorithm.

Besides improving the performance in the classification task by these approaches, another objective to be searched in this work is to exploit the ANN potential for the task of unsupervised feature extraction.

In this way a modular neural architecture (fig.1) was proposed, dividing the classification problem in two phases: a module for feature extraction from the RS image by Kohonen's Self-Organizing Map (SOM) and a module for classification, using a Multi-Layer Perceptron (MLP) network, where the above ideas were tested.

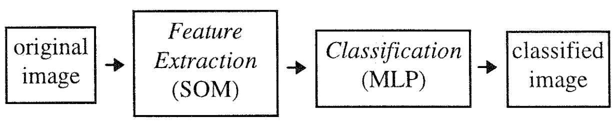

Figure 1: Neural architecture for classification.

2. FEATURE EXTRACTION BY KOHONEN MAP

The objective of the feature extraction phase is to identify the spectral classes present in the image and to define the set of correspondent samples to be used in the classification phase afterwards.

There is no well-developed theory for feature extraction, mostly features are application-oriented and often found by heuristic methods and interactive data analysis.

An important basic principle is that the features must be independent of class membership because, by definition, at the feature extraction phase the membership to the classes is not yet known. This implies that any learning methods used for feature extraction should be *unsupervised* in the sense that the target class for each object is unknown (Oja et al. 1994).

One of the approaches is the use of competitive learning resulting in data clustering. An example is Kohonen's Self-Organizing Map (SOM) (Kohonen 1988).

It's well known the SOM property of dividing the input space into convex regions, where a set of reference vectors associates vector codes with the input space. The classification of an image may then be based on the cluster codes found to the image by the SOM.

In our approach we generated an auxiliary visual tool from the SOM, denominated *Kohonen Clusters Map* (KCM), which enables to identify the spectral classes present in the image through the visualization of the clusters generated by SOM.

2.1. SOM Description

The SOM belongs to the class of unsupervised neural netwoks based on competitive learning, in which only one output neuron , or one per local group of neurons at a time gives the active response to the current input signal. The level of activity indicates the similarity between the input signal vector and its respective weight vector. A standard way of expressing similarity is through the Euclidian distance between these vectors.

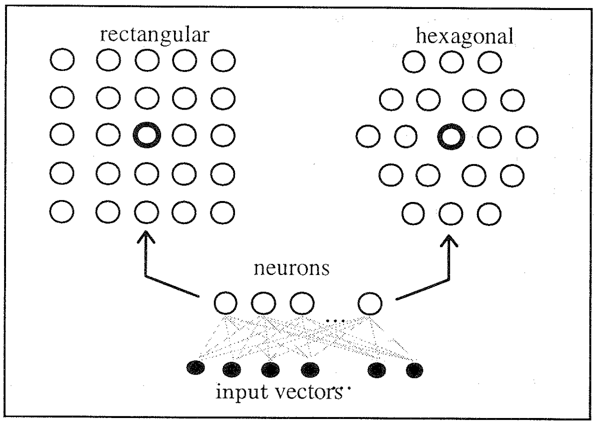

Figure 2: Geometrical representations of neurons for SOM .

Since the distance between the weight vector of a given neuron and the input data vector is minimal to all neurons in the network, this neuron together with a predefined set of neighbour neurons will have their weights automatically updated by the learning algorithm. The neighbourhood for each neuron may be defined accordingly to the geometrical form, over which the neurons are arranged. Figure 2 depicts two examples of representation proposed by (Kohonen 1989): a rectangular grid and an hexagonal grid.

A short description of the learning algorithm of SOM is given bellow:

Step 1: Select a training pattern $X = (x_1, x_2, \ldots, x_N)$ and present it as an input to the network.

Step 2: Compute distances d_i between the input vector, and each j neuron's weight vector, acording to:

$$d_i = \sum_{j=1}^{N} (x_j(t) - w_{i,j}(t))^2 \qquad (1)$$

where $x_j(t)$ is the j-th input in a given iteraction and $w_{i,j}(t)$ is the weight of neuron j from the input layer connected to neuron i from the output layer.

Step 3: Select neuron i^* with the smallest distance among all other neurons, and update the weight vector of i^* and its neighbours using the following expression:

$$w_{i,j}(t+1) = w_{i,j}(t) + \alpha(t) * (x_j(t) - w_{i,j}(t))$$
$$\text{for } i \in N_{i^*}, j = 1,2,\ldots,N \qquad (2)$$

where N_{i^*} is a set that contains i^* and its neighbours, and $\alpha(t)$ is the learning rate, usually smaller than 1. This procedure repeats until the the weight update is no longer significant.

By the end of the learning process each neuron or group of neighbour neurons will represent a distinct pattern among the set of patterns presented as input to the network.

2.2. Kohonen Clusters Map (KCM)

In this approach, 3x3 pixel windows taken from the original image were used as training patterns for the SOM. These patterns were randomly and uniformly obtained from all over the image and presented as input vectors to the SOM. Since the SOM has the property of arranging its weight vectors in rectangular or hexagonal *grids* and considering that both input data vectors and weight vectors have the same dimension, this enables to generate an *image of the weight grid of the SOM.*

The resulting grid image, after the unsupervised learning by the SOM, was denominated *Kohonen Clusters Map* (KCM). Figure 5 shows an example of a rectangular KCM generated from the test image (Fig. 8).

The KCM produces a visual auxiliary tool for the task of identifying and selecting the spectral classes present in the image and their correspondent training samples, which will be used afterwards in the module of neural classification. The

KCM has some important characteristics that enable these tasks:

• As the SOM performs a clustering on the training patterns, it's possible to visualize the spectral classes present in the original image through the clusters obtained in the KCM.

• The SOM property of preserving the topological relationships among the input data vectors is reflected in the KCM property of preserving these relationships among the clusters so obtained, in terms of distances among them. Clusters that are close to each other in KCM represent land cover classes which posses similar spectral features;

• The SOM property of preserving the probability distributions found in the input data can be verified in the KCM, where higher frequency spectral classes in the input data will be mapped onto bigger regions in the KCM.

Therefore, the spectral classes and their samples, which will be used in the classification phase, are selected from the KCM and not directly from the original image as is usually done.

2.3 Parallel Implementation of SOM

The inherent parallelism of ANN is well known. Efficient parallel implementation of neural networks both in hardware and in software is an active research field.

In this module for feature extraction the parallel implementation of SOM was realized by software, aiming at improving the performance in terms of the training time of SOM, using a tool developed by the Oak Ridge National Laboratory, the Parallel Virtual Machine (PVM).

PVM is a software system that enables a collection of heterogeneous computers to be used as a coherent and flexible concurrent computational resource. The individual computers may be shared- or local-memory multiprocessors, vector supercomputers, specialized graphics engines, or scalar workstations, that may be interconnected by a variety of networks, such as Ethernet, FDDI.

User programs written in C, C++ or Fortran access PVM through library routines. Daemon programs provide communication and process control between computers.

For SOM, the basic idea for paralleling the training algorithm was to allocate sets of neurons from the SOM to the processors, distributing the training patterns among them, so as to reduce the global computational time.

A comparison between the performance of the sequential and the parallel training algorithm of SOM is shown in section 4.

3. MLP FOR CLASSIFICATION

Having selected the desired classes and their correspondent samples from KCM, the objective of the second phase in our proposed system is to perform the final classification of the image using a Multilayer Perceptron (MLP) network.

MLP belongs to the class of feedforward neural networks, consisting of a number of neurons which are connected by weighted links. The units are organized in several layers, namely an input layer, one or more hidden layers, and an output layer. The input layer receives an external external activation vector, and passes it via weighted connections to the units in the first hidden layer. These compute their activations and pass them to neurons in succeeding layers.

The training of the MLP network in performed in a supervised way, where the objective is to tune the weights in the network such that the network performs a desired mapping of input to output activations.

The MLP network in our system has one hidden layer (fig.3). The number of neurons per layer varies according to the number of classes and to the size of selected samples to perform the training.

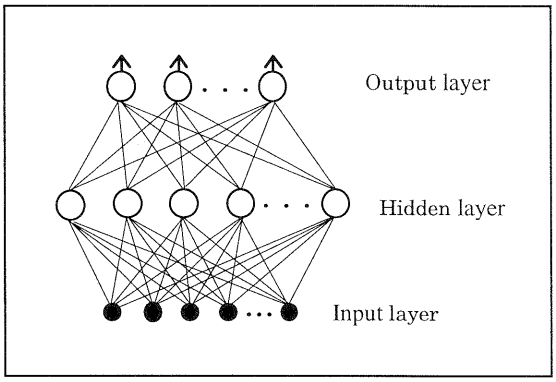

Figure 3: MLP network.

3.1 Training algorithm for MLP

Several adaptive learning algorithms for MLP neural networks have recently been discovered. Many of these algorithms are based on the gradient descent algorithm well know in optimization theory. They usually have poor convergence rate and depend on parameters which have to be specified by the user, as no theoretical basis for choosing them exists. The values of these parameters are often crucial for the success of the algorithm. An example is the standard backpropagation algorithm (Rumelhart et al 1986), which often behaves very badly on large-scale problems and whose success depends on user dependent parameters like learning rate and momentum constant (Moller 1993), which is often the case with RS applications, that normally handle large and full of details images.

In this search for alternatives to the poor performance of standard backpropagation, normally pointed out as the main drawback to a broader utilization in RS image classification, this work used an advanced training algorithm for the MLP network.

The MLP learning algorithm used here is an improved version of the Scaled Conjugate Gradient (SCG) algorithm presented in (Moller 1993).

The SCG is a learning algorithm for ANN developed from the general conjugate gradient methods for optimization problems, which are well suited to handle large-scale problems in an effective way (Fletcher 1975). The SCG method makes use of both first- and second-order information to determine the parameters in the learning process, but Moeller uses an approximation for the 2nd-order information.

(Pearlmutter 1994) solved the problem concerning the computation and storage of the Hessian matrix, which is responsible for the 2nd-order information, in a surprisingly efficient way, resulting in a SCG learning algorithm that presents a final convergence much quicker than the standard backpropagation.

4. EXPERIMENTAL RESULTS

A TM/LANDSAT image composed by bands 7, 4 and 5, respectively associated to the colors red, green and blue, was used to test the efficiency of the proposed neural architecture (figure 7). The image has 512x512 pixels, and it corresponds to the crossing region of rivers Pardo and Mogi (São Paulo state, Brazil, longitude 48^{0} west and 21^{0} south) and was taken on August 8th of 1990 .

The SOM used in the first phase was composed of 600 neurons, geometrically distributed over a rectangular grid of 20 rows by 30 columns. The KCM obtained after the SOM training is shown in fig.8.

The table bellow shows the training time (in seconds) for the sequential and for the PVM parallel implementation for three grid sizes of SOM. The machines used for the training were IBM POWERstation 360 workstations.

SOM dimension	sequential	parallel (6 machines)
20x30	6110	2458
40x60	27300	10707
80x100	75632	26280

Table 4: SOM training time performance.

After the SOM training, 5 land cover classes were easily identified from the generated KCM with the assistance of an expert: nude soil, water, humid soil, growing crop and vegetation. The samples correspondent to these classes were selected near the corners and in the center of the KCM. Each sample for the neural classification was composed by a 3x3 pixel window.

error	backpropagation	advanced SCG
1e-2	989	169
1e-3	1606	239
1e-4	8029	343
1e-5	60625	364
1e-6	542720	439

Table 5: MLP training time performance.

These samples were presented in the training of the MLP network. The network has 12 neurons in the hidden layer and 5 neurons in the output layer, correspondent to the land cover classes chosen for this test classification. The table 5 shows the performance in terms of number of epochs in the training process between the standard backpropagation learning algorithm and the SCG algorithm with the exact computation of second order information.

To evaluate now the classification performance of the proposed architecture its results were compared to those obtained by the Maximum Likelihood algorithm, a statistical classification method widely used. The samples used for classification by the Maximum Likelihood algorithm were 5x5 pixel windows taken directly from the original image as is conventionally done.

The performance classification results between both methods are shown on table 6.

	neural network	maximum likelihood
total of pixels from training samples	81	425
nonclassified pixels	37641	48862
relative variance	0.5214	0.3747
kappa coefficient	59	

Table 6: Classification performance.

First it's shown the total of pixels from the samples that were needed for both methods. Then two measures for evaluating the classification performance for each method were given: the number of nonclassified pixels and the Relative Variance (Johnson et al. 1982). The Relative Variance describes the percentage of variability from the original data which are explained by the classification. In this way the bigger the Relative Variance the better the classification.

Finally the table shows the kappa coefficient, which is a measure of the concordance between two data sets, it's another commonly used parameter to evaluate the classification accuracy of sattelite images (Rocha 1992). In this case the kappa coefficient is used to compare two classified images, where one image's pixels are used as the set of reference patterns while the other image's pixels are the set of testing patterns.

Figures 10 and 11 show the classified test image by the neural architecture and by the Maximum Likelihood algorithm respectively with the 5 land cover classes chosen.

5. CONCLUSIONS AND FUTURE WORK

The advantages of the proposed system may be better understood taking into account that ANN classification is commonly done by a single MLP network, where the feature extraction task, i.e., the selection of classes and samples is done directly from the original image by an expert. With the modularization of the architecture, the feature extraction task was performed by SOM, generating an auxiliary visual tool, the Kohonen Clusters Map (KCM), which provides useful information regarding the representativity, the distribution and the similarity of spectral classes. It makes possible the identification and selection by visual inspection of the spectral classes and its respective training samples, which will be used in the classification phase. These advantages are emphasized when the image presents a highly complex variability of spectral classes, which may difficult the task of selecting the classes directly from the original image

Figure 9 shows the classified KCM itself, where one can verify the separation of classes, building *decision regions* for the classification. The non-classified regions, in black, may indicate the presence of distinct classes, which may be interactively incorporated in finer and more comprehensive selections of classes. This may lead to increasingly better and/or more specific classifications.

In this work we also proposed alternatives for improving the performance of Artificial Neural Networks in terms of training time, with the parallel implementation of the SOM and an advanced learning algorithm for the MLP network, a version of the Scale Conjugate Gradient algorithm instead of the standard backpropagation.

The positive results achieved show the feasibility to search for new techniques either in software or in hardware and their combination for reaching increasingly better performances as shown in tables 4 and 5. As for the classification performance itself, the proposed neural architecture showed superior performance over a well known statistical classification method (table 6). The results obtained for neural classification associated with its performance improvement in terms of time motivate therefore to continue and to expand research efforts in this area, since still only a limited part of the power of the neural nets is actually being utilized.

As an example, another research direction is to further investigate the applicability and flexibility of dynamic structural determination techniques for the SOM and MLP networks, as proposed by (Fritzke 1995) and (Zuben 1996) respectively. These techniques consist basically in enabling the network structure to grow up until specific performance criteria to the application be achieved.

6. ACKNOWLEDGMENTS

The first author thanks to Jesus Christ, the Source of Life and Hope. The first author also thanks to Nelson Hi Man Pak for his comments and suggestions. This research was developed in part with help of CENAPAD (National Center for High Performance Processing - São Paulo - Brazil) project UNICAMP/FINEP - MCT.

7. REFERENCES

Benediktsson, J. A., Swain, P. H., Ersoy, O. K., 1990. Neural Network Approaches Versus Statistical Methods in Classification of Multisource Remote Sensing Data. IEEE Transactions on Geoscience and Remote Sensing, 28(4), pp. 540-552.

Fritzke, B., 1995. Growing Grid - a self-organizing network with constant neighborhood range and adaptation strength. Neural Processing Letters, 2(5), pp. 1-5.

Fletcher, R., 1975. Pratical Methods of Optimization. New York: John Wiley & Sons.

Hepner, G. F., Logan, T., Ritter, N., Bryant, N., 1990. Artificial Neural Networks Classification using a Minimal Training Set: Comparison to Conventional Supervised Classification. Photogrammetric Engineering and Remote Sensing, 56(4), pp. 469-473.

Jacobs, R. A., Jordan, M. I., Barto, A. G., 1991. Task Decomposition through Competition in a Modular Connectionist Architecture: The what and where vision tasks. Cognitive Science, 15, pp. 219-250.

Johnson, R.A., Wichern, D.W., 1982. Applied Multivariate Statistical Analysis, Prentice-Hall, New Jersey.

Key, J., Maslanik, J. A., Schweiger, A. J., 1989. Classification of Merged AVHRR and SMMR Artic Data with Neural Networks. Photogrammetric Engineering and Remote Sensing, 55(9), pp. 1331-1338.

Kanellopoulos, I., Varfis, A., Wilkinson, G. G. and Mégier, J., 1992. Land-cover Discrimination in SPOT HRV Imagery using an Artificial Neural Network - a 20-class experiment. International Journal of Remote Sensing, 13(5), pp. 917-924.

Kohonen, T., 1988. Self-Organization and Associative Memory. Springer-Verlag, New York.

Liu, Z. K., Xiao, J. Y., 1991. Classification of Remotely-sensed Image Data using Artificial Neural Networks. International Journal of Remote Sensing, 12(11), pp. 2433-2438.

Moller, M. F., 1993. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. Neural Networks, 6(4), pp. 525-533.

Oja, E., Lampinen, J., 1994. Unsupervised Learning for Feature Extraction. Computational Intelligence: Imitating Life, IEEE Press.

Pearlmutter, B. A., 1994. Fast Exact Multiplication by the Hessian. Neural Computation, 6(1), pp. 147-160.

Rocha, J.V., 1992. The Influence of Ground Survey Size on Accuracy of Area Estimates from Satellite Images. Ph.D. Thesis, Silsoe College.

Rumelhart, D. E. and McClelland, J. L., 1986. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. MIT Press, Cambridge, MA.

Schlünzen, E. T. M., 1993. Perspectives on the Application of Neural Networks in the Classification of Remote Sensing Multispectral Data. M.Sc. Thesis, Departament of Computer Engineering and Automation (DCA), School of Electrical Engineering (FEE), State University of Campinas (UNICAMP), Campinas, Brazil. (in portuguese)

Zuben, F. J. V., 1996. Parametric and Nonparametric Models of Artificial Neural Networks and Applications. Ph.D. Thesis, Departament of Computer Engineering and Automation (DCA), School of Electrical Engineering (FEE), State University of Campinas (UNICAMP), Campinas, Brazil. (in portuguese)

Clustering of input data by **Self-Organizing Map (SOM)**

Figure 7: Landsat TM image (composition of bands 7, 4 and 5).

Classification by **Maximum Likelihood Algorithm**

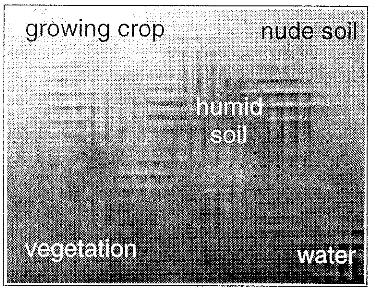

Figure 8: Kohonen Clusters Map (KCM).

Classification by **Multilayer Perceptron (MLP)**

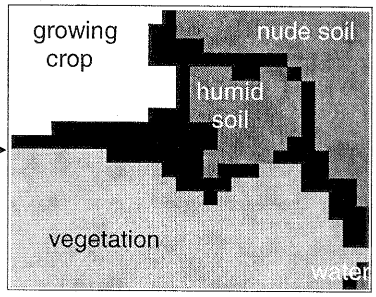

Figure 9: Kohonen Clusters Map classified.

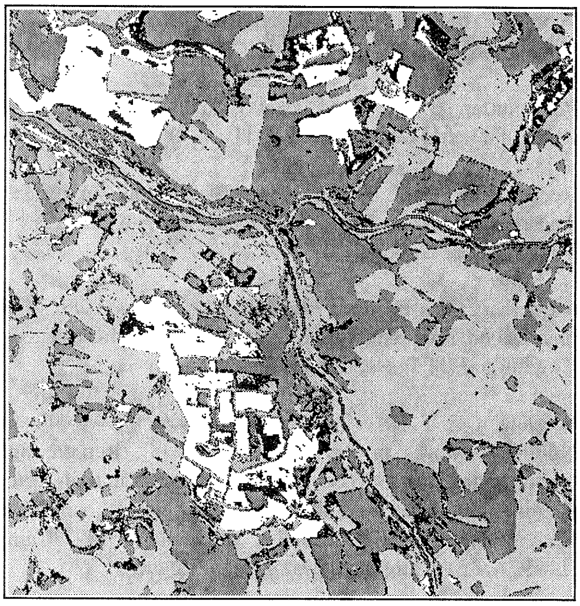

Figure 10: ANN classification.

Figure 11: Maximum Likelihood Algorithm classification.