**Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais, Vol. 2, No. 1, pp. 14-21, 2004**


# Spatiotemporal Patterns Estimation Using a Multilayer Perceptron Neural Network in a Solar Physics Application

**Maria Conceição de Andrade[1], Atair Rios Neto[2], Reinaldo R. Rosa[2], Hanumant S. Sawant[1], Francisco C. R. Fernandes[1]**

[1]Astrophysics Division DAS-INPE C.P. 515 12201-970 S.J. Campos, SP, Brazil

[2] Computing and Applied Mathematics Laboratory LAC-INPE C.P. 515 12201-970 S.J. Campos, SP, Brazil

E-mails: con@das.inpe.br, atairrn@uol.com.br, reinaldo@lac.inpe.br, sawant@das.inpe.br, guga@das.inpe.br

**Abstract** - In this paper we intend to evaluate the use of Multilayer Perceptron neural networks for a spatiotemporal patterns estimation problem and to compare the performance of the Kalman Filtering with the Backpropagation and the Levenberg-Marquardt training algorithms. The study consists of applying Multilayer Perceptron for estimation of the solar active regions evolution using sequential soft X-ray images observed by the YOHKOH solar satellite telescope. In this application, the performance test is done by using the mean squared error, image visualization and the Gradient Pattern Analysis (GPA) techniques. The last one is based on the operator for characterization of Amplitude Asymmetric Fragmentation (AAF). The AAF operator is being used for the first time for a performance test with an Artificial Neural Network (ANN) applied in spatiotemporal patterns estimation. The results confirm the efficiency and efficacy of ANN as a tool to estimate spatiotemporal patterns in this kind of application. The tests indicate that although the Kalman Filtering showed an efficacy to learn the patterns comparable to those of the Backpropagation and the Levenberg-Marquardt algorithms, it is inefficient from the computational viewpoint in the sense that it takes a longer processing time. In addition, the ANN performance validation tests confirm the utility of the AAF operator for the performance characterization of spatiotemporal patterns estimation algorithms.

**Keywords:** artificial neural networks, supervised training, spatiotemporal series estimation, gradient pattern analysis, non-linear systems, solar physics

## 1. INTRODUCTION

The implementation and performance of Artificial Neural Networks (ANN) for the solar corona spatiotemporal patterns estimation are investigated. The patterns are visualized by satellite images of the solar actives regions (http://www.solar.isas.ac.jp). To perform the patterns estimation, the Backpropagation [1], the Levenberg-Marquardt [2] and the Kalman Filtering [3] training algorithms are used and compared with each other. Based on these images, solar events forecast studies can allow us to predict the possible subsequent effects in the earth nearest regions (magnetic storms and ionospheric disturbances). The technique has been tested as a phenomenological tool to help the prediction of solar activity into the Latin-American Space Weather Forecasting Program (PLAICE) [4] and in the Brazilian Decimetric Array Project (BDA) at DAS-INPE [5,6]. The technique will also be one of the tools in the Phenomenological Analysis System (SAF), which has been developed by Nucleus for Simulation and Analysis of Complex Systems (NUSASC) at LAC-INPE. The neural network output will be mainly evaluated using the Amplitude Asymmetric Fragmentation (AAF) matrix computational operator [7]. This operator is applied to both images, namely to the neural network answers (estimate) and to the real images.

The expectation in getting good results with ANN is based on the fact that in the last few years the ANN have been established to be an effective tool in temporal series behavior forecast. They have also been applied for some solar-terrestrial temporal series forecast (e.g. [8]). In this work we are extending this application for the spatiotemporal domain (images sequence analysis). The use of ANN in image prediction is attractive because they are capable of learning patterns, which due to their complexity are hard to analyze by other technique. They also have the ability of integrating information from samples and incorporating new characteristics without degrading the previously acquired knowledge.

## 2. TRAINING ALGORITHMS

The Backpropagation is the most common learning algorithm, however a neural network implementation based on it could take too many steps and thus a long training time. Therefore there are many studies based on heuristics or on numeric optimization techniques to accelerate the algorithm convergence in order to improve its efficiency related to the training time. In order to do this, one of the chosen heuristics to this work is a definition of the momentum term. This heuristic has been demonstrated to have a good performance. We also choose the Levenberg-Marquardt [9] and the Kalman Filtering [3,10,11] as numeric optimization techniques. Although they present a greater complexity they have been demonstrated to be efficient when compared to the gradient descent technique.

### 2.1 Backpropagation Algorithm

The Backpropagation is the most popular algorithm to train Multilayer neural networks. It was developed by Rumelhart *et al.* in 1986. It implements the gradient descent technique, in order to minimize the mean square error between the desired and network output (with respect to the weights). When the network is properly trained with the Backpropagation algorithm it tends to give reasonable answers when presented to new inputs that it has never seen.

### 2.2 Levenberg-Marquardt Algorithm

The Levenberg Marquardt is an approximation to Newton's method, an optimization technique more powerful than the descent gradient. The Levenberg-Marquardt weight actualization rule is:

$$\Delta w = (J^T J + \mu I)^{-1} J^T e, \qquad (1)$$

where $J$ is the Jacobian matrix of derivatives of errors with respect to the weights, $\mu$ is a scalar, and $e$ is the error vector. If the scalar $\mu$ is large, the above expression represents the steepest descent (with step $1/\mu$), while for small $\mu$ it reduces to the Gauss-Newton method, which is more rapid and more precise, near the minimum error. Thus the goal is to transfer the learning to the Gauss-Newton method. So $\mu$ is increased or decreased after each step depending upon the case [2].

### 2.3 Extended Kalman Filtering Algorithm

The Kalman Filtering is an algorithm used to estimate dynamic system states from noisy measurements. Thus it is possible to use it in neural network for adjusting (estimating) weights, to find a computational model for a given data set mapping such as:

$$\{(x(t), y(t)) : y(t) = f(x(t), t = 1,2,...,m)\}. \qquad (2)$$

The neural network weights are the states to be estimated and the network outputs are the measurements from which the Kalman filter does the estimation:

$$\hat{y}(t) = \hat{f}(x(t), \hat{w}) \qquad (3)$$

where: $\hat{w}$ is the estimated neuron synaptic weight vector and $x(t)$ are the network inputs.

In the proposed solution [3], a linear perturbation is done in an iteration $i$, changing the training into the following estimation problem:

$$\varpi = w(i) + \bar{e}, \qquad (4)$$

$$\alpha(i)[y(t) - \bar{y}(t,i)] \cong \hat{f}_w(x(t), \overline{w}(i))[w(i) - \overline{w}(i)] + v(t) \qquad (5)$$

where: $i = 1,2,...I$, $\varpi(i)$ is the a priori estimate of $w$ coming from the previous iteration, starting with $\varpi(1) = \varpi$, $\bar{y}(t,i) = \hat{f}(x(t), \varpi(i))$. Here $\hat{f}_w(x(t), \overline{w}(i))$ is the matrix of partial derivatives with respect to $w$; $0<\alpha(i)\leq 1$ is a parameter to be adjusted in order to guarantee the hypothesis of linear perturbation, $\bar{e}$ is the a priori error distribution, and $v(t)$ are the Gaussian random variables, the error in the network output approximation.

## 3. VALIDATION

### 3.1 Mean Square Error (MSE)

The square error $E$ is the answer that comes from an input vector $x$ presented to the network, producing an output signal $y$. Based on this error, the learning performance index is obtained using the MSE criteria:

$$E_{med} = \frac{1}{N}\left[\sum_{n=1}^{N} E(n)\right]. \qquad (6)$$

This is the most commonly used technique to validate the neural network results. However the MSE value by itself is not sufficient to represent the visual quality of an image, i.e. the MSE is not always capable of determining the necessary precision to validate the neural network answers [12]. Therefore, to complete the neural network answers validation, a new technique is applied in addition to the estimated images visualization. This new technique is explained next.

### 3.2 Gradient Pattern Analysis

The Gradient Pattern Analysis (GPA) was developed in order to characterize complex regimes in the spatiotemporal domain [7,13,14,15]. It was first applied to the identification of weak turbulence patterns in the solar coronal plasma [14]. Nowadays this technique is being applied not only to the characterization of non linear phenomena observed in the spatiotemporal domain (intermittence, spatiotemporal chaos) but also to the characterization of complex structures (convex asymmetry, labyrinth and non-homogeneous roughness) in a strictly spatial domain, such as the study of porosity in semiconductors [16,17]. This technique involves the application of two computational operators, which act over the gradient field of a given matrix. One of these operators is the Amplitude Asymmetric Fragmentation (AAF), which computes the broken symmetry in the gradient field of a matrix. This operator working over the spatiotemporal domain is efficient to detect small non-linear fluctuations that are not visualized with accuracy during the spatiotemporal patterns evolution. In this work the AAF operator was used, for the first time, to test the neural network estimation performance working in the spatiotemporal domain.

The AAF operator is capable of locally quantifying the symmetry breaking in the image field gradient. It turns a matrix $M$ into a triangulation field with $L$ points and $I$ lines, where the $L$ points correspond to the asymmetric vector numbers of the matrix gradient field and the $I$ lines correspond to the number of triangulation lines between the $L$ points. The measure of the asymmetric fragmentation degree is given by the value $F_A = (I - L)/L$. This parameter shows high sensitivity for characterization of small alterations in a matrix gradient field, reflecting the possible symmetry breaking that can occur in the spatiotemporal domain [7]. A set of output examples from the GPA application is shown in figures 7, 8, 9 and 10. The asymmetries in the gradient field are given by the $F_A$ parameter calculated from the asymmetric triangulation field.

## 4. DATA, PATTERN SET AND TRAINING METHODOLOGY

The data used to form the pattern set are 10 soft X-ray images of a solar corona active region, obtained from the YOHKOH satellite telescopes (http://www.solar.isas.ac.jp), in a temporal sequence. An image series with a little pattern alteration was used. These images observed in 04/21/92 from 11h00’ to 13h41’ UT, are available in the ASCII format, as 128×128 lattice (16384 pixels). The image sequence has a different temporal resolution, since the images are not taken at regular time intervals. Therefore it was necessary to interpolate the images in order to make the intervals equal, for getting the training and test sets. A linear interpolation with a time interval of 3 minutes between the images was used.

Two approaches were taken for the neural network architecture. In the first one it was defined an architecture whose training patterns were formed by image temporal series of a certain spatial region where the whole image was considered in time $t$ and $t+1$ for the input and output patterns, respectively. The number of hidden layers and of perceptron neurons in each layer were adjusted empirically. Figure 1 illustrates this kind of architecture.

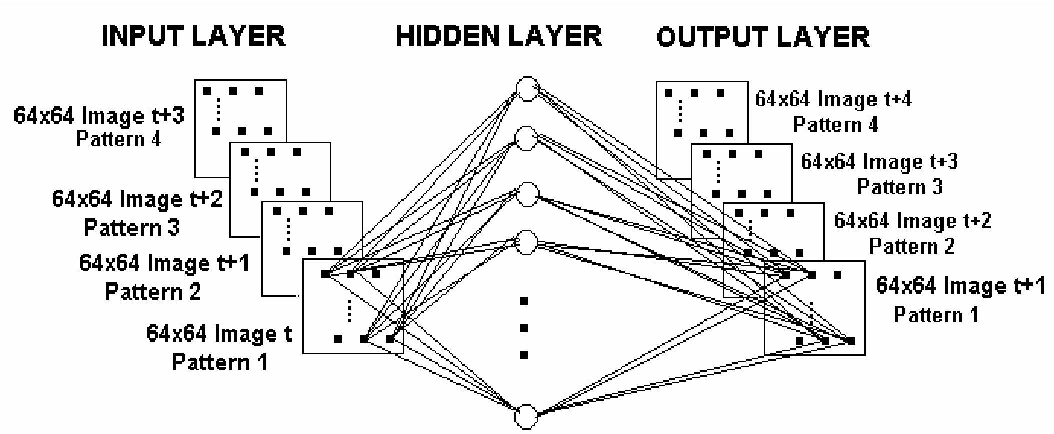


Figure 1: Illustration of first approach for ANN architecture

The large volume of data in the input and output of the ANN was a problem for the training algorithms when this architecture was used. To deal with this difficulty the images were resized. The computational memory limitations constrained the use of Backpropagation with images of 64×64 pixels and of Levenberg-Marquardt and Kalman Filtering with images of 16x16 pixels. However 16x16 pixels image was found to be not enough for the neural network to learn the spatiotemporal patterns behavior.

The second approach was constructed in the spatial domain. Starting from a given spatial position of the image, one defines as the ANN input pattern the pixels in the same spatial position in the delayed images of a temporal series, and as the output the pixel in the same position of the image one step of time ahead. Since each of the images have 128×128 pixels and since in the training they were organized in a series of five images, 16384 neural network patterns were formed with four inputs and one output each one. Figure 2 illustrates this kind of architecture.

The simulations were done in a microcomputer (architecture CISC compatible with IBM-PC standard) Pentium III, 1.1GHz and 512 Mb of RAM memory.

Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais, Vol. 2, No. 1, pp. 14-21, 2004


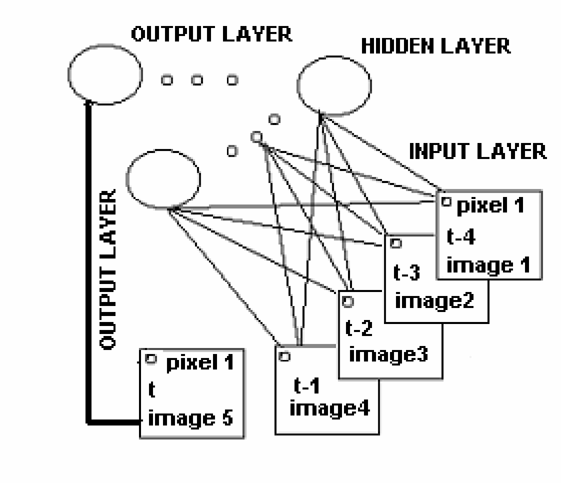


Figure 2: Second approach for ANN architecture

## 5. RESULTS

### 5.1- First Approach

The Levenberg-Marquardt and the Kalman Filtering algorithms did not show a good performance when used with this neural network architecture, since they need a lot of memory in the weight correction computation The learning algorithm with better training performance with this approach was Backpropagation with momentum. The best training, with an architecture of 64-15-64 nodes in each layer, took about 1000 epochs in 40 minutes approximately. The ANN parameters were: learning rate 0.5, the momentum constant 0.5, the threshold $10^{-3}$ and the activation functions sigmoid and the identity were used on the hidden and the output layer respectively. The values of MSE were $1.25x10^{-3}$ and 0.5 (training and test respectively). The values of $F_A$ were 1.98926 to the real and 1.98865 to the estimated image. Figure 3 shows the temporal series (64×64 pixels images) used as input-output pattern in all training during this approach and Figure 4 shows the image used for the test, the image that the neural network should estimate and the neural network answer.

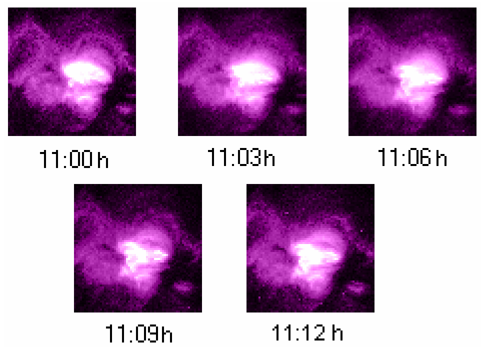


Figure 3: Training patterns used for training with the first approach

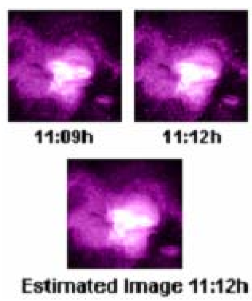


Figure 4: Training results with the first approach: image used in the test (11:09h), image to be estimated (11:12h) and estimated image with Backpropagation algorithm (11:12h below)

## 5.2- Second Approach and Kalman Filtering Algorithm

In the second approach the architecture of the neural network was defined with 18 neurons in the hidden layer, 4 nodes in the input layer, one node in the output layer and 16384 patterns were used in the training. The activation functions sigmoid and the identity were used on the hidden and the output layer respectively. Figure 5 shows the temporal series (128 ×128 pixels images) used as input-output patterns in the second approach training.

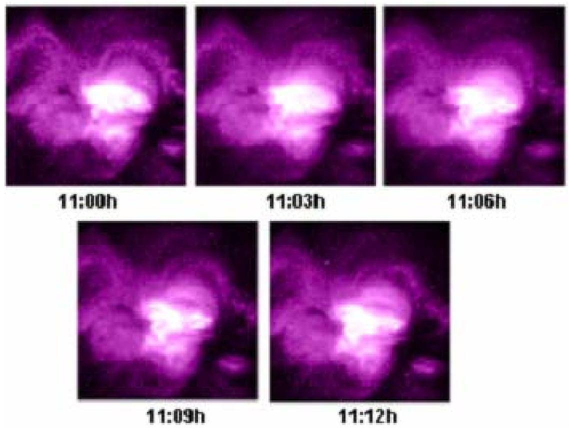


Figure 5: Training patterns with the second approach

The best training of the Kalman Filtering algorithm with the architecture defined above was made with a learning rate 0.1, noise value 0.1 (variances of uncorrelated errors in output patterns) and threshold $10^{-6}$. The training took 20 epochs in 20 minutes approximately, with the training MSE of $2.1\times10^{-3}$ and the test MSE of $1.9\times10^{-5}$. The values of $F_A$ are 1.99202, for the real image, and 1.99183 for the estimated image (Figures 7 and 8 respectively). Figure 6 shows the temporal series patterns used for the test (13h27' to 13h36'), the image that the neural network should estimate (13h39') and the neural network answer with second approach and Kalman Filtering algorithm.

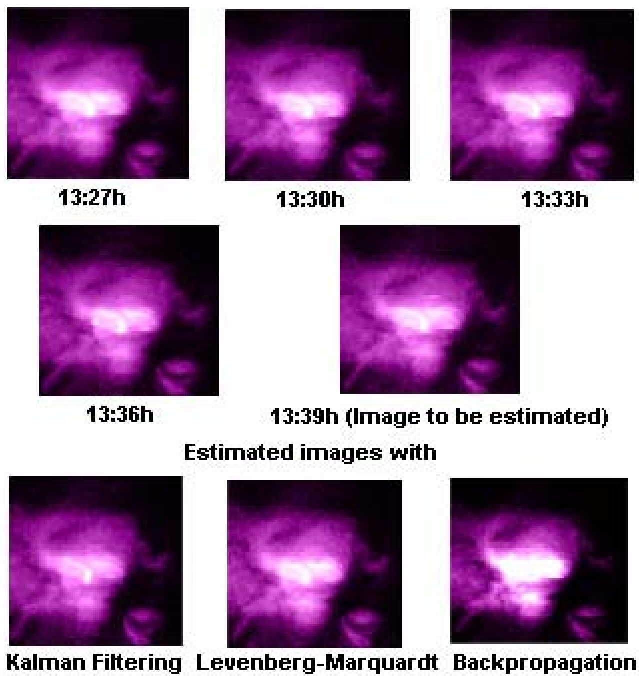


Figure 6: Training with the second approach: ANN test patterns (13h27'-13h36'); the image to be estimated for the ANN (13h39'); the estimated image with Kalman Filtering, Levenberg-Marquardt and Backpropagation algorithms respectively.

## 5.3- Second Approach and Levenberg-Marquardt Algorithm

The architecture of the neural network and the activation functions for this training were the same as used in the Kalman Filtering. The best training of the Levenberg-Marquardt algorithm was made with learning rate 0.1 and threshold $10^{-6}$. The training took 15 epochs in 10 minutes approximately, with the training MSE of $1.89\times10^{-3}$ and the test MSE of $1.2\times10^{-3}$. The values of $F_A$ are 1.99202, for the real image, and 1.99315 for the estimated image (Figures 7 and 9, respectively). Figure 6 shows the neural network answer with second approach and Levenberg-Marquardt algorithm.

## 5.4- Second Approach and Backpropagation Algorithm

The neural network architecture and the activation functions for this training were the same as used in both the above cases. The best training of the Backpropagation algorithm was made with learning rate 0.5, threshold $10^{-6}$ and momentum constant 0.9. The training took about 100 epochs in 10 minutes approximately, with the training MSE of $4.91\times10^{-3}$ and the test MSE of $4.00\times10^{-3}$. The values of $F_A$ are 1.99202, for the real image, and 1.99396 for the estimated image (Figures 7 and 10 respectively). Figure 6 shows the neural network answer with second approach and Backpropagation algorithm.

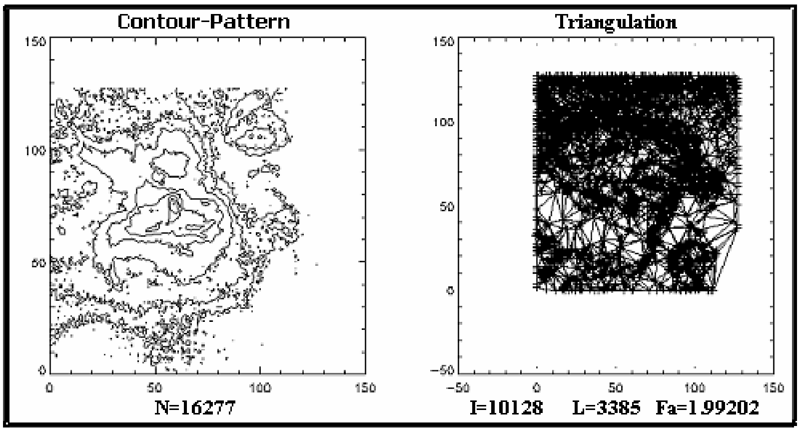


Figure 7: $F_A$ to be estimated (Yohkoh image 13h39')

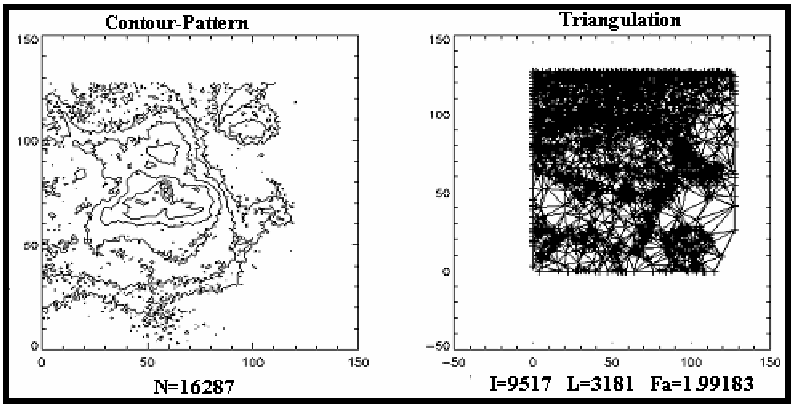


Figure 8: $F_A$ estimated by Kalman Filtering Algorithm (Yohkoh image 13h39')

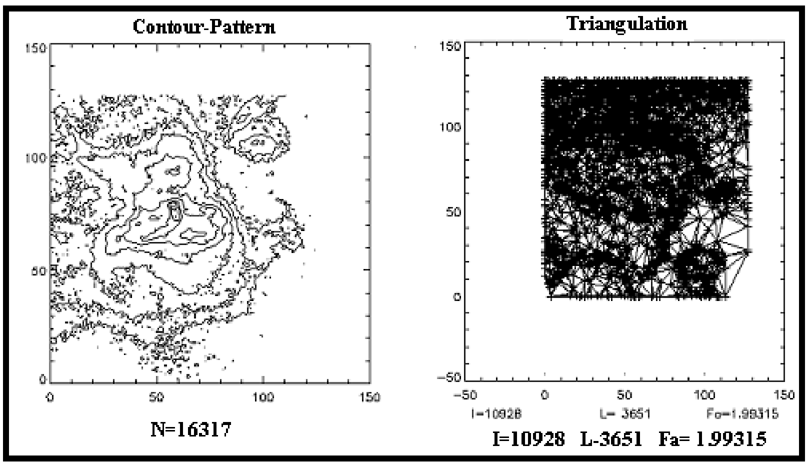


Figure 9: $F_A$ estimated by Levenberg-Marquardt Algorithm (Yohkoh image 13h39')

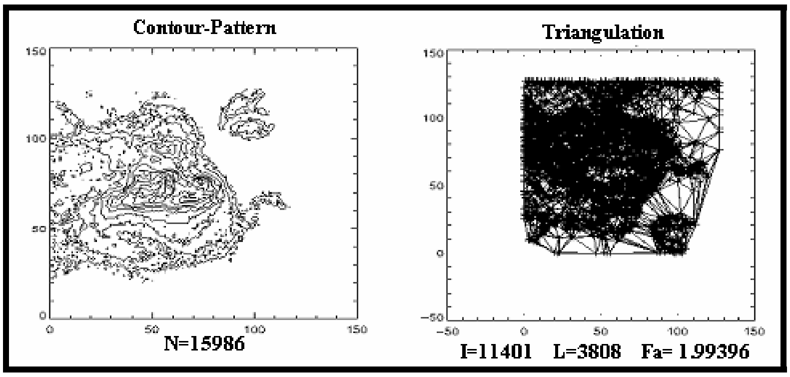


Figure 10: $F_A$ estimated by Backpropagation Algorithm (Yohkoh image 13h39')

## 6. CONCLUDING REMARKS

We presented in this paper an evaluation of the use of Multilayer Perceptron neural networks in a spatiotemporal patterns estimation problem. To do this we considered the use of solar images of active regions (local scale) using the Kalman Filtering learning algorithm comparing its results with Levenberg-Marquardt and Backpropagation with momentum.

Performance tests were done by using the mean squared error (MSE), image visualization and the Gradient Pattern Analysis (GPA) techniques. Visual inspections of images indicated that the MSE test was not effective in the evaluation and validation of ANN learning capacity and of training algorithms performance. The Gradient Pattern Analysis technique based on the operator for characterization of Amplitude Asymmetric Fragmentation (AAF) was used for the first time, and the validation tests confirm its effectiveness in the performance characterization of spatiotemporal patterns estimation algorithms.

With the first approach, throughout the training period, it was noticed that although Kalman Filtering and Levenberg-Marquardt training algorithms are robust they turned out to be inefficient from the computational memory viewpoint. These algorithms use matrices in the numeric calculations to estimate the neural network weights, with sizes increasing with the number of inputs, hidden nodes and outputs, leading to an excessive computational demand. The Backpropagation with momentum with its numerical simplicity made it feasible to train an ANN with this architecture but only at the cost of a significant reduction in the resolution of the images.

The computational memory problem was solved with the second architecture. With this architecture the Kalman Filtering presented slightly better results (MSE, $F_A$ and image visualization) with an efficacy to learn the patterns comparable to that of the Levenberg-Marquardt and the Backpropagation algorithms. However Kalman Filtering still needs to be optimized in its processing time to improve its efficiency.

From the preliminary results one can conclude that the artificial neural network methodology is promising for the solar coronal pattern estimation. The three algorithms have been shown to be effective tools to train neural network for short-term forecast (three minutes) for spatiotemporal patterns produced by non-linear systems. However their performance should be further evaluated for other phenomenological forecast applications.

## 7. REFERENCES


[1] D. E. Rumelhart, G. E. Hinton and R. J. Willims. "Learning Representations of Backpropagation Errors." *Nature* 323:533-536, 1986.

[2] M. T. Hagan and M. B. Menhaj. "Training Feedforward Networks with The Marquardt Algorithm." *IEEE Transactions on Neural Networks* 5(6), 1994.

[3] A. Rios Neto. "Stochastic optimal linear parameter estimation and neural nets training in systems modeling." *Journal of the Braz. Soc. Mechan. Science*s, 19, (2): 138-146, 1997.

[4] R. R. Rosa, H. S. Sawant, J. R. Cecatto, C. Rodrigues Neto, V. A. Lopes, K. R. Subramanian, F. R. Fernandes, J. H. Saito, C. E. Moron, M. L. Mucheroni, N. Furuya and N. Mascarenhas. "Phenomenological Dynamics of Coronal Loops Using a Neural Network Approach." *Adv. Space Res*., 25(9):1917-1921, 2000.

[5] H. S. Sawant, E. Ludke, K. R. Subramanian, F. R. Fernandes, J. R. Cecatto and R. R. Rosa. "A high resolution decimetric solar radioheliograph." *Astronomical Soc. of the Pacific Conference Serie*s, bf206:341-346, Edited by R. Ramaty and N. Mandzhavidze, 2000a.

[6] H. S. Sawant, K. R. Subramanian, E. Ludke, J. H. Sobral, G. Swarup, F. C. Fernandes, R. R. Rosa and J. R. Cecatto. "Brazilian Decimetric Array." *Advances in Space Researc*h, 25 (9):1809-1812, 2000b.

[7] R. R. Rosa, A. S. Sharma and J. A. Valdivia. "Characterization of Asymetric Fragmentation Patterns in Spatially Extended Systems." *Intern. Journal of Modern Physics C*10(1):147-163, 1999.

[8] K. P. Macpherson, A. J. Conway and J. C. Brown. "Prediction of Solar and Geomagnetic Activity Data Using Neural Networks." *Journal of Geophysical Research* 100(A11):21735-21744, 1995.

[9] D. Marquardt. "An algorithm for least squares estimation of non-linear parameters." *J. Soc. Ind. Appl. Math.* 431-441, 1963.

[10] S. Singhal and L. Wu. "Training feedforward networks with the extended Kalman filter." *Proceedings of IEEE Intern. Conference on Acoustics Speech and Signal Proc.* 1187-1190, Glasgow, Scotland: IEEE Press, 1989.

[11] R. S. Scalero and N. Tepedelenlioglu. "A fast new algorithm for training feedforward neural network." *IEEE Transactions on Signal Proc.* 40(1):202-210, 1992.

[12] F. Russo. "Hybrid neuro-fuzzy filter for impulse noise removal." *Pattern Recognition* 32:1843-1855, 1999.

[13] R. R. Rosa, C. Rodrigues Neto, F. Ramos, A. S. Sharma and J. A. Valdivia. "Computational operators for dynamical complex pattern recognition." In EPS-Modelling Collective Phenom. in Complex Systems, v. 22F, Pub. By European Phys. Soc., p. 304-305, 1998b.

[14] R. R. Rosa, A. S. Sharma and J. A. Valdivia. "Characterization of localized turbulence in plasma extended systems." *Physica A* 257(1-4):509-514, 1998a.

[15] R. R. Rosa, M. R Campos, F. M. Ramos and N. L. VijayKumar. "Gradient Pattern Analysis of Structural Dynamics: Application to Molecular System Relaxation." Brazilian Journal of Physics, vol. 33, no. 3, September, 2003.

[16] A. Ferreira da Silva, R. R. Rosa, L.S. Roman and E. Veje. "Characterization of asymmetric fragmentation patterns in SFM images of porous silicon." *Solid State Comm.* 113(12): 703-708, 2000a.

[17] A. Ferreira da Silva, R. R. Rosa, P. W. Machado, F.M. Ramos, C. Rodrigues Neto, L.S. Roman and E. Veje. "Asymmetric nonlinear amplitude in patterns of porous silicon." *Physica A* 283:223-227, 2000b.185.

[18]

## Acknowledgements

The authors are grateful to the Astrophysics Division and Computing and Applied Mathematics Laboratory of INPE for the support received in order to develop this work experiments. This work was partially supported by FAPESP (Grant, Proc. 97/13374-1). We are indebted to Laerte Sodré Jr. for useful discussion.