Neural Processing Letters **19:** 175–189, 2004.

Neural Network Based Recognition of Smoke Signatures from Lidar Signals

ARMANDO M. FERNANDES, ANDREI B. UTKIN, ALEXANDER V. LAVROV[†] and RUI M. VILAR
Departamento de Engenharia de Materiais, Instituto Superior Técnico, Av. Rovisco Pais 1, Lisbon 1049-001, Portugal. e-mail: {afernandes;rui.vilar}@ist.utl.pt

Abstract. The automatic recognition of smoke signatures in lidar signals collected during very small-scale experimental forest fires using neural-network algorithms was studied. An algorithm for pre-processing of raw lidar signals is proposed, which selects suspicious backscattering peaks and makes them unbiased and scale independent. The resulting patterns can be successfully classified as corresponding to alarm or no-alarm conditions by a neural-network algorithm based on a simple one-neuron structure (perceptron). In the case of an alarm, the pre-processing algorithm provides the location of the smoke plume. Five algorithms selected from the literature, and one that was specially developed, all with learning rate adaptation, were used for training the perceptron. Their efficiencies and statistical properties were compared. The best perceptron classifier presented an efficiency of 97% in the classification of smoke-signature patterns and a false alarm rate of 0.9%.

Key words. backpropagation, fire, generalisation, laser, lidar, misdetections, perceptron, smoke

Abbreviations. BP – backpropagation; BPSDLR – backpropagation with adaptation of self-determined learning rate; NPRCG – Newton-like method with periodically restarted conjugate gradient; PA – polynomial approximation; PNR – peak-to-noise ratio; PPRCG – polynomial approximation with periodically restarted conjugate gradient; RPROP – resilient propagation; SDAS – steepest descent with adaptive step size

1. Introduction

The lidar (LIght detection and ranging) technique has been widely used in atmospheric and environmental research [4, 11, 14]. A lidar consists of a radiation emitter, receiver optics (usually a Newtonian or Cassegrainian telescope), a photo-detector, and data processing hardware and software. The radiation emitter is a pulsed laser, which emits short intense radiation pulses through the atmosphere. Part of the radiation that is backscattered by molecules, particles, and objects is collected by the receiver and measured as a function of time. The distance from the backscattering centre to the lidar may be calculated from the time delay between the laser pulse emission and the reception of the backscattered signal, τ, and the velocity of light, c, by the equation $R = c\tau/2$. The raw lidar signal is represented by the dependence of

[†]On leave from Russian Science Center 'Applied Chemistry', St. Petersburg 197198, Russia

the backscattered radiation intensity on the distance at which backscattering occurred. To improve the signal-to-noise ratio, returns from several laser pulses are usually accumulated in one signal.

Remote detection of forest fires using lidar was recently demonstrated by the authors [20, 21]. This technique presents considerable advantages when compared to passive surveillance methods, because it yields higher sensitivity and does not require line-of-sight observation of the flames. The high scattering efficiency of the particles in the smoke which results from burning wood and grass, enables smoke plumes from small forest fires to be easily detected up to distances of 6.5 km using a 100 mJ Q-switched Nd:YAG laser based lidar [20].

For successful application of lidar in automatic forest fire surveillance the smoke signatures in lidar signals must be promptly recognised and false alarms rejected. False alarms resulting from atmospheric phenomena lead to spurious peaks in the lidar signal. Lack of knowledge of aerosol distribution in the atmosphere and its dependence on weather conditions, as well as random changes in atmospheric refraction index, make it difficult to use parametric models of the lidar sensing process. Neural-network techniques have been successfully applied to automatic forest fire detection based on images obtained using ground-based, airborne, or spaceborne infrared or video cameras [1, 3, 17, 19]. They have also been exhaustively applied to the classification of radar [8], sonar [7], and sodar [16] signals. The present application and previously reported applications are however different because in image analysis two-dimensional scenes are classified, while in the present case the classification deals with a one-dimensional signal representing the distribution of backscattered radiation intensity along the laser beam propagation direction [2].

The present paper reports the successful application of neural networks to the recognition of smoke signatures in raw lidar signals. Five learning algorithms previously reported in the literature and one specifically designed were tested and compared, with the aim of assessing which algorithm provides the best performance for the present application. The neural networks were used to classify signals as a smoke plume (alarm) or absence of smoke (no alarm) and, in an alarm situation, the pre-processing algorithm provides the distance to the smoke plume. The training and validation sets were constructed from experimental curves containing peaks corresponding to smoke plumes and atmospheric noise. The work described in the present paper is part of a research programme that aims at applying the best learning algorithm to the training of neural networks that are associated in committee machine structures [6].

2. Network Structure and Signal Pre-processing

As classification implies a one-directional processing of information, the natural option for the network topology is a feedforward structure, whose simplest implementation is a single-layer perceptron [9]. While a multilayer perceptron is composed of several layers of neurons and one-directional synaptic links between neighbouring

layers of neurons, a simple perceptron is composed of only one layer of input nodes connected by synaptic links to a single output neuron. The performance of the simplest neural network represented by a perceptron was investigated in the present study. The activation function used in the output neuron is the hyperbolic tangent, which maps the outputs into the bounded interval (−1, 1) and, at the same time, provides non-zero and continuous gradient values necessary for learning algorithms based on error backpropagation (BP).

Information processing with a single neuron is linear, even though a non-linear function is used in its output [9]. The non-linear activation function can only help reducing oscillations around the minima of the error function during the minimisation procedure, by decreasing the gradient value when the weights are close to those that minimise the error function. Therefore the application of single-layer perceptrons is confined to the classification of linearly separable patterns. However, the simple structure of the perceptron results in shorter calculation times for both learning and classification, a very important feature for real-time applications.

The amplitude of the smoke-plume peak in a raw lidar signal varies due to changes in the density of particles in the smoke plume, distance, and wind. The background in the lidar signal curve, as shown in Figure 1, is composed of electronics noise and atmospheric noise [5]. Since this signal background can be represented as a linear combination of non-coincident peaks, the distinction of smoke signatures from atmospheric noise using a perceptron is in principle impossible. However, preliminary tests have demonstrated that smoke signatures and atmospheric noise peaks can be efficiently distinguished by perceptron-based algorithms if the segments used for classification are centred around the peak maximum. Although the signal-to-noise ratio decreases with distance, the smoke signature holds its shape. Thus, the classification problem is distance-independent in the sense that the recognition conditions for a tenuous smoke plume observed at some distance are equivalent to

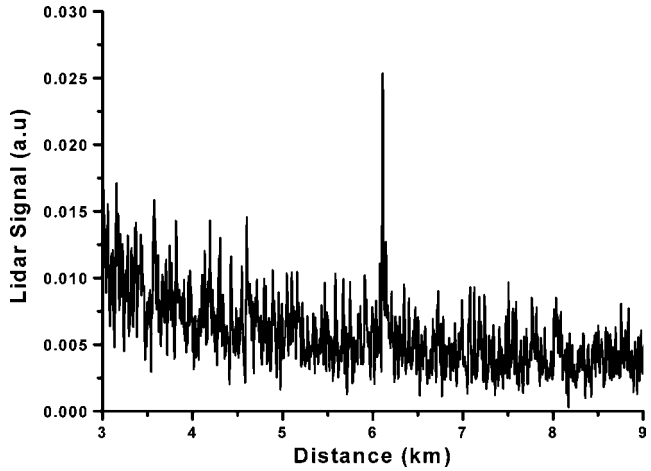

Figure 1. Lidar curve with smoke signature at 6.1 km.

those for a denser plume at a greater distance, provided that the signal-to-noise ratio is similar. This fact enables a lidar signal of one to two thousand points (a range of about 10 km, measured with six-metre resolution) to be viewed by the neural network through a narrow window of several tens of points, which moves along the signal curve, reducing signal pre-processing to the window motion algorithm. Motion stops only if a local maximum coincides with the window centre. When this situation occurs, the signal curve in the region of interest is scaled so that the minimum and maximum values are −0.9 and 0.9, in order to eliminate the signal background and avoid saturation of the neurons. This technique considerably simplifies all data processing because: (i) the number of neural inputs and links is reduced to the window width, decreasing calculation complexity; (ii) recognition becomes independent of scale and of signal background; (iii) linear separation methods, in particular simple perceptron-based algorithms, become applicable; and (iv) the neural network is only required for classification, while the problem of calculating the distance to the target is eliminated, because this distance is given by the co-ordinate of the window centre.

3. Pattern Description

Two sets of patterns were used for the supervised learning: a training set consisting of the lidar patterns and corresponding outputs to be learnt by the network, and the validation set, used to test the ability of the trained network to generalise. The patterns in the training and validation sets contain peaks due to atmospheric noise and smoke-plume signatures of experimental campfires (burning rate of about 0.02 kg/s) resulting from the accumulation of medium (32) and large (more than 128) numbers of lidar returns in one raw lidar signal. The lidar signals, from which the patterns were gathered, were obtained using laser radiation with wavelengths of 532 and 1064 nm and laser pulse energies ranging from 2 to 17 mJ. However, variations in these parameters did not produce any substantial differences in the characteristics of smoke-signature or atmospheric noise patterns.

To capture the smoke-plume structure, the signals were recorded with a 6-m distance between sampling points, resulting in approximately 7-points-wide smoke signatures. After some trial runs of the recognition algorithms, a width of the sliding window and, correspondingly, a pattern length of 21 points, three times the smoke signature width, was chosen. In fact this number of points was sufficient to provide statistically meaningful information about the central peak and the noise in its vicinity and so to supply enough information to allow a correct classification of the pattern on the basis of the peaks shape and width, and the peak amplitude in relation to the background. In accordance with the pre-processing procedure, all patterns had centred maximums and were normalised to fit in the interval [−0.9, 0.9], as presented in Figure 2.

In order to quantitatively characterise the pattern noisiness, a parameter called peak-to-noise ratio (PNR) was introduced. The PNR is defined as the ratio between

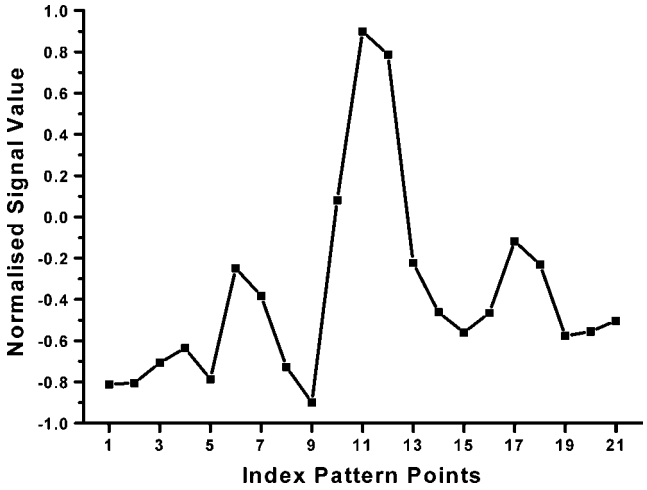

Figure 2. Smoke signature extracted from the lidar curve presented in Figure 1.

the maximum amplitude and the standard deviation of the pattern points in the vicinity of the peak (here PNR appears to be a more appropriate term than signal-to-noise ratio, because the central maximum itself may correspond to noise). The maximum and standard deviation values are calculated with reference to a linear background obtained by the root-mean-square interpolation of all pattern points except seven points in the centre, which presumably form the peak to be analysed.

The distribution of the PNR for the smoke-signature patterns is shown in Figure 3. The smoke-signature patterns present PNRs over a wide range from 7 to 92, although the highest concentration of patterns occurs for $7 \leq \text{PNR} \leq 16$. The histograms of PNR distribution for atmospheric noise and the smoke signatures are

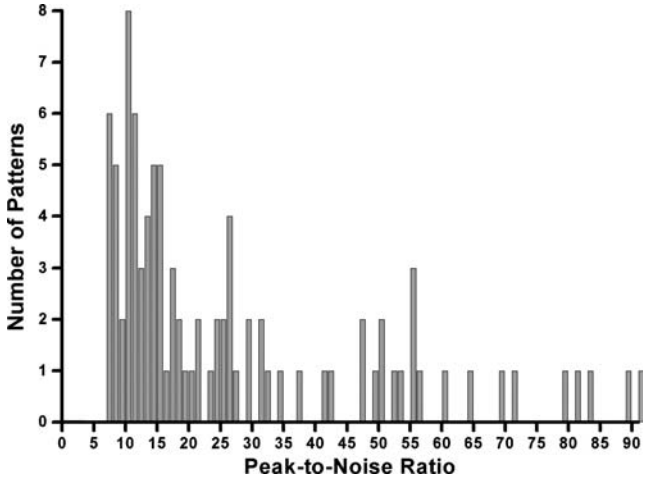

Figure 3. Distribution of smoke-signature patterns over PNR.

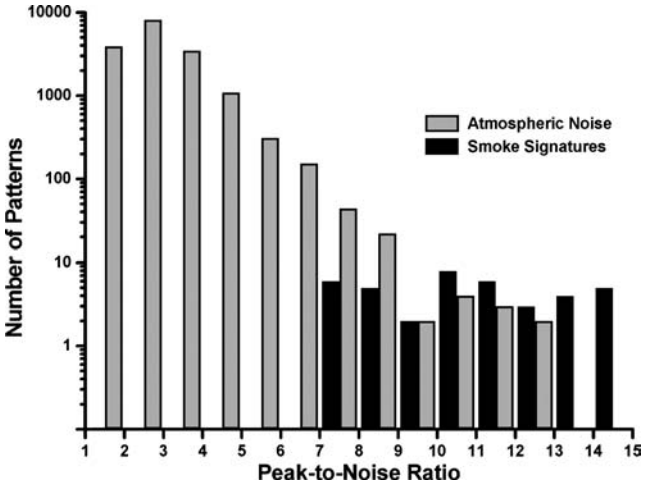

Figure 4. Distribution of atmospheric noise and smoke-signature patterns over PNR.

compared in Figure 4. The histogram shows that atmospheric noise has a pronounced non-Gaussian distribution [12], with a slow decay towards larger PNR values. Despite the noise reduction resulting from the accumulation of at least 32 lidar returns in each signal, the two distributions overlap considerably. For this reason, methods based on a signal-to-noise criterion, which have been traditionally used in automatic signal recognition, are not the most appropriate. Forest-fire detection with a low false-alarm rate can only be achieved with derivation of the decision from several parameters characterising the signal peak. This fact provides additional support for the application of neural networks that form the decision from as many signal parameters as they have input nodes.

4. Learning Algorithms

BP is an efficient algorithm for supervised learning of multilayer and simple perceptrons. This method minimises output errors by gradient descent over the error surface $E(\mathbf{w})$ in the multidimensional space of the interconnection weights $\mathbf{w}$ (transmission coefficients of the synaptic links). The weights are updated according to the rule

$$w_i \rightarrow w_i - \mu \nabla_i E \tag{1}$$

where w_i is the weight of the link to the ith input node, μ is the learning rate, and $\nabla_i E = \partial E / \partial w_i$ is the component of the error gradient corresponding to the direction w_i. In the simplest implementation of the BP algorithm, the learning rate remains fixed. This results in slow convergence, because the error surface usually presents gentle slopes intersected by narrow steep regions. In steep regions the learning rate has to be low to prevent over-adjustment, while in flat regions it has to be high in order to compensate the small local gradient values. Taking this fact into consid-

eration, all the learning algorithms tested in the present work use dynamic adaptation of the learning rate in function of the changes in the error gradient value. Six algorithms were used for training the perceptrons, five of them selected from the literature (back propagation with adaptation of self-determined learning rate, BPSDLR; steepest descent with adaptive step size, SDAS; resilient propagation, RPROP; polynomial approximation, PA and Newton-like method with periodically restarted conjugate gradient, NPRCG, please see below), and the sixth (polynomial approximation with periodically restarted conjugated gradient, PPRCG) specially developed by the authors for the present classification task.

In the calculation of the optimal learning rate, the BP method with adaptation of a self-determined learning rate, BPSDLR [13] takes error and error gradient values into consideration. Accelerating the descent, this option leads to poor stability, so a special convergence criterion tests each value of the learning rate in order to ensure algorithm convergence. If convergence cannot be guaranteed, learning-rate tuning is performed by reducing its value to half.

For the steepest descent with adaptive step size, SDAS [22], a Lipschitz continuity condition is imposed on the error gradient, and the optimal learning rate is calculated on the basis of a local estimation of the Lipschitz constant.

In the resilient propagation method, RPROP [18], the calculation of the value of the weight update does not take into account the magnitude of the error gradient. The update is based on changes of the error gradient sign because they indicate a jump over error surface minima. If the sign does not change, the weight update value increases; if the sign changes, the weight update value decreases, enabling the weight vector to step closer to the minimum of the error surface.

The following three BP selected methods perform dynamic learning rate adaptation through the computation of both the first and second derivatives of the error. The computational complexity is higher than for standard BP methods, but the gain in convergence speed fully compensates for it. The first two methods are taken from the literature, while the third one, derived from the previous two, is novel.

The objective of polynomial approximation, PA [23] is to find the smallest learning rate, $\mu = \mu^*$, for which the error function has a minimum. Since ∇E indicates the local descent direction, the value of the first derivative of the error with respect to the learning rate is less than zero for $\mu = 0$. Starting with zero, μ is increased stepwise until the first derivative becomes positive. With two values of the learning rate, one corresponding to positive and the other to negative values of the first derivative of the error, it is possible to estimate μ^* using a third-order polynomial interpolation.

In the Newton-like method with periodically restarted conjugate gradient, NPRCG [23] it is assumed that a convex parabola can approximate the dependence of the error function on the learning rate. The optimal learning rate is calculated from the first and second derivatives of the error function. The momentum factor, which defines the influence of the previous weight update values on the current ones, leading to more stable and faster convergence, is simultaneously calculated. At the points of the weight space where the second derivative of the error for $\mu = 0$ is equal

to or less than zero the calculations do not converge. At these points polynomial approximation is used instead.

Combining polynomial interpolation with the periodically restarted conjugate gradient it is possible to obtain a new algorithm, polynomial approximation with periodically restarted conjugate gradient, PPRCG, which exhibits very fast convergence while maintaining good generalisation characteristics. In this algorithm the learning rate is obtained using the third-order polynomial interpolation, but the descent is performed through the conjugate-gradient method. To the best of the authors' knowledge, this algorithm was not reported previously.

5. Numerical Results

Numerical computations were performed using a perceptron consisting of 21 signal input nodes and an additional node that always receives a signal $+1$ (to represent the neural activation bias in the same form as the signal components), connected by synaptic links with a single output neuron. To find the conditions that provide the best generalisation capability for the perceptron, the following learning parameters were varied:

- the range for randomly generating the initial weights of the synaptic links (the intervals $[-0.2, 0.2]$, $[-0.5, 0.5]$ and $[-0.8, 0.8]$ were tested);
- the number of patterns in the training set;
- the number of updates to the interconnection weights (number of training epochs).

Random generation of the initial weights introduces a stochastic character to the algorithms and makes it necessary to assess the statistical properties of the results by repeating each combination of the above parameters 10 times.

5.1. OPEN-SKY LIDAR OBSERVATIONS: RECOGNITION OF SMOKE SIGNATURES AGAINST ATMOSPHERIC NOISE

From the thousands of patterns gathered, 282 representative patterns were chosen for the training and validation sets. The atmospheric-noise patterns were collected from lidar signals of the smoke-free atmosphere resulting from the accumulation of 32 or more lidar returns. From lidar signals containing peaks due to smoke plumes, it was possible to gather 95 patterns with smoke signatures. All smoke signatures have PNR > 7. Due to the smaller number of smoke signature patterns in comparison with atmospheric noise patterns, all the smoke signature patterns were used in the composition of the training and validation sets. Several training sets containing 37, 62, 78 and 118 patterns were built, and the rest of the patterns were used in the validation set. For all tested algorithms, the 78-pattern training set produced the classifiers showing the least number of misclassifications. This training set was composed of 41 smoke signatures and 37 atmospheric-noise patterns. Together with this training set, a validation set composed of 54 smoke signatures and 150

atmospheric noise patterns was used. At the same time, changing the weight initialisation range did not produce significant variations in the percentage of misclassifications.

Bearing in mind the application, the algorithm efficiency was assessed by the number of misclassifications (misdetections and false alarms), although training was performed using the Euclidean error. For a classic step-function activation of the neuron, each misclassification adds one unity to the Euclidean classification error and these two assessments would be equal, but not in the present situation. As a result, the best algorithm is not the one leading to the lowest error for a given training set because, after reaching a certain value, further reductions in the error imply a degradation of the generalisation capability of the classifier [10]. The important point is how fast an algorithm can attain the classification error level corresponding to a minimum number of misclassifications for both the training and validation sets. To find the algorithm capable of producing classifiers yielding the lowest number of misclassifications, 360 perceptrons were constructed for each of the six algorithms with the 78-pattern training set and subjected to testing with 204 patterns of the validation set. A histogram showing the number of classifiers that achieved a particular number of misclassifications in both sets is presented in Figure 5. No algorithm led to a classifier achieving less than 5 misclassifications and only PPRCG and NPRCG produced classifiers leading to less than 10 misclassifications. PPRCG predominates in the group of the best classifiers (5–9 misdetections) but produces its maximum number of classifiers in the next group (10–14).

In Figure 6 the total number of misclassifications for each algorithm is plotted as a function of the number of training epochs (as previously, the results presented were

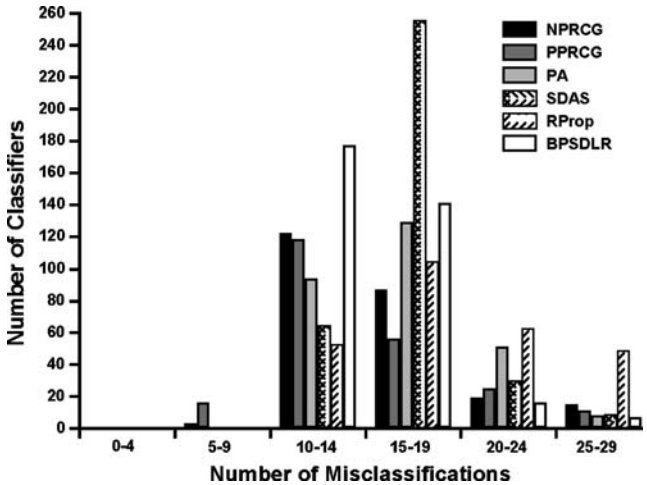

Figure 5. Distribution of classifiers over total number of misclassifications with respect to 78 patterns of the training set and 204 patterns of the validation set.

Figure 6. Dynamics of the learning process for the six tested algorithms.

obtained with the 78-pattern training set). Ten trials were performed for each set of parameters and the results averaged. Each algorithm presents a minimum of misclassifications for a particular number of training epochs. If the number of training epochs is too small, the perceptron does not capture the characteristic features of the patterns in the training set; if the number of epochs is too large, the perceptron is excessively affected by irrelevant features in the training patterns, resulting in poor generalisation capability [10].

Table 1. Values of N_{Epoch}, N_{Error}, N_{Grad} and TC for the six algorithms studied.

	N_{Epoch}	N_{Error}	N_{Grad}	TC
PPRCG	11	2	55.77 ± 25.63	1862
NPRCG	18	1	1	72
BPSDLR	120	7.378 ± 1.643	1	1245
SDAS	100	1	1	400
RPROP	140	1	1	560
Polynomial	300	2	9.088 ± 0.8893	8779

PPRCG and NPRCG are the only algorithms capable of attaining a percentage of misclassifications lower than 4.5%. The minimum percentage of misclassifications is 3.83 ± 0.297% and 4.22 ± 0.414% for PPRCG and NPRCG, respectively. Thus for the problem in question, PPRCG appears to be the best learning algorithm because it exhibits the lowest percentage of misclassifications.

In Table 1 are presented the average number of evaluations per epoch of gradient and error function (N_{Grad} and N_{Error}, respectively), the average number of epochs required by each algorithm to attain the lowest percentage of misclassifications (N_{Epoch}) and the corresponding total computational cost (TC). The average and standard deviation values indicated in Table 1 were calculated by averaging the results of 10 runs. The other values are constant for each algorithm. The computational cost of each gradient evaluation (CC_{Grad}) was estimated according to the formula suggested by [15]:

$$CC_{\text{Grad}} = 3 \times CC_{\text{Error}} \tag{2}$$

where CC_{Error} is the computational cost of each error function evaluation. The total computational cost (TC) was calculated using the expression:

$$TC = N_{\text{Epoch}}(3N_{\text{Grad}} + N_{\text{Error}}) \tag{3}$$

Different learning algorithms present the minimum percentage of misclassifications at different values of the training epochs. PPRCG and NPRCG attain their best performance, on average, in 11 and 18 training epochs respectively, while the other learning algorithms need 100 or more training epochs. Since the second order derivatives computational cost for 18 epochs is negligible as compared to other TC values, NPRCG is the learning algorithm requiring the lowest total computational cost to achieve its best performance. PPRCG allows the minimum misclassification percentage to be achieved in only 11 epochs, but at the expense of a larger number of gradient function evaluations per epoch as compared to other algorithms. Nevertheless using a computer equipped with a 1.7 GHz Pentium IV processor and 256 MB of memory PPRCG takes approximately 5 s to calculate 11 epochs for 78 training patterns.

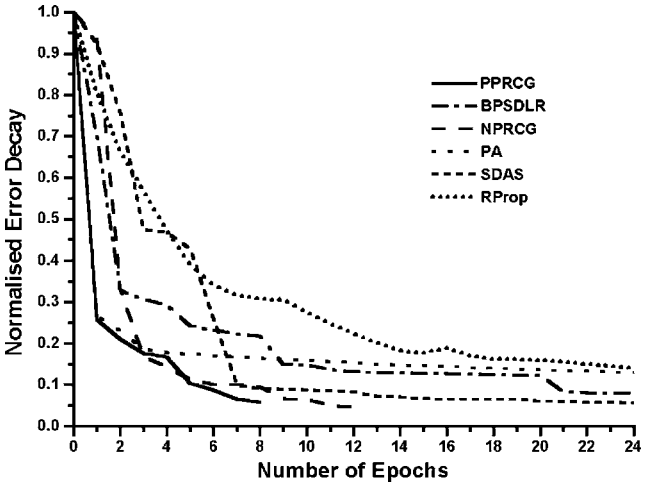

Figure 7. Normalised decay of the classification error.

The decay of the classification error, measured as the sum of the squared Euclidean distances between the desired and the actual output for all patterns in the training set [9], is plotted in Figure 7 as a function of the number of training epochs. For each algorithm, the curves show the error decay for the classifier that presents the best results.

PPRCG presents the fastest error decay, because it benefits from the best characteristics of each of the algorithms that compose it. For the first three epochs, the PPRCG error decay is similar to the PA algorithm, which presents the fastest error decrease in these epochs. After that, PPRCG demonstrates a slightly faster convergence than NPRCG. The PPRCG and NPRCG methods achieve a 5% level of the initial error after 8 and 12 epochs, respectively. The other algorithms need more than 25 epochs to achieve the same result.

The best classifier was constructed with the PPRCG learning algorithm and the training set of 78 patterns, 41 containing smoke signatures and 37 containing atmospheric noise peaks. The validation set was composed of 204 patterns, 54 with smoke signatures and 150 with atmospheric noise. The efficiency of the best classifier was 99% for the validation set (no misdetections and three false alarms), and 98% (three false alarms and three misdetections) with respect to both the training and validation sets. This means 97% efficiency in the classification of smoke-signature patterns. When evaluating the efficiency of this classifier with respect to 17,174 atmospheric noise patterns, it was able to correctly classify 99.1% of the patterns. The histogram of Figure 8 shows the PNR distribution for the atmospheric noise patterns causing false alarms. For given PNR groups, the percentage of patterns that cause false alarms increases with PNR. The presence of false alarms in almost all groups of PNR for atmospheric noise patterns illustrates the fact that perceptrons base their decisions on features other than PNR.

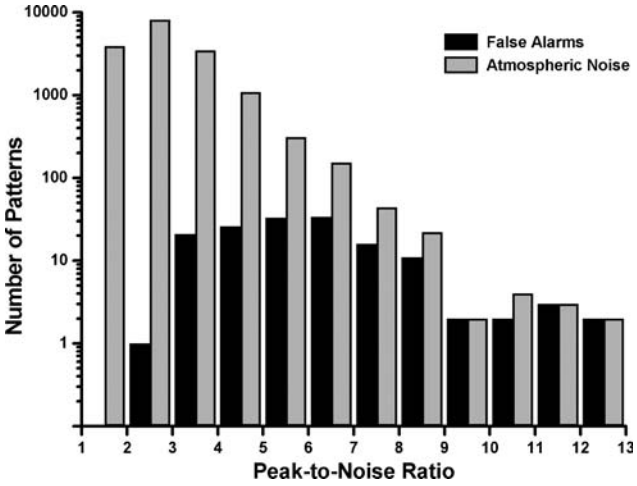

Figure 8. Distribution of atmospheric noise and false alarm patterns over PNR.

6. Conclusions

The possibility of applying neural networks to smoke signature recognition was demonstrated by constructing a sliding-window pre-processing algorithm coupled with a simple single-neuron classifier. The pre-processing algorithm provides the distance to the smoke plume. The smoke-signature patterns are characterised by PNRs ranging from 7 to 92, testifying of the high variability in plume and fire parameters as well as of the lidar signal acquisition conditions in general.

The six learning algorithms under study presented different learning characteristics. PPRCG and NPRCG were the only ones able to generate classifiers presenting less than 10 misclassifications for training and validation sets. PPRCG and NPRCG obtained their best performance, on average, in less than 20 training epochs; all the other learning algorithms needed at least 100 training epochs. Among the algorithms studied, NPRCG presents the lowest computational cost and is the second best in terms of misclassification percentage. The BP procedure of polynomial approximation with periodically restarted conjugate gradient (PPRCG) developed in the present work turned out to be the best learning algorithm for neural-network classifiers of raw lidar signals even though it does not present the lowest computational cost. PPRCG can produce the classifiers leading to the lowest number of misclassifications obtained in this study, while maintaining optimum generalisation. These results indicate that for the problem in question the periodically restarted conjugate gradient calculation drastically affects the convergence rate while improving classification efficiency.

With PPRCG it was possible to obtain a perceptron showing 97% efficiency in the classification of smoke-signatures with peak-to-noise ratio higher than 7, and 99.1% efficiency in the classification of the total of 17,174 noise patterns resulting from the accumulation of 32 or more lidar returns. The atmospheric noise patterns presenting higher values of PNR have a higher probability of resulting in false alarms.

Acknowledgements

A. Fernandes gratefully acknowledges Ph.D. grant SFRH/BD/2943/2000 from Fundação para a Ciência e a Tecnologia. A. B. Utkin gratefully acknowledges financial support from Instituto de Ciência e Engenharia de Materiais e Superfícies. This research was partially supported by INTAS grant N. 99-1634. The authors wish to thank to the Portuguese Air Force for providing the location where the experiments took place.

References

1. Arrue, B. C., Ollero, A. and Martinez de Dios, J. R.: An intelligent system for false alarm reduction in infrared forest-fire detection, *IEEE Intell. Syst.* May/June (2000), 64–72.
2. Bhattacharya, D., Pillai, S. R. and Antoniou, A.: Waveform classification and information extraction from LIDAR data by neural networks, *IEEE Trans. Geosci. Remote Sensing*, **35**(3) (1997), 699–707.
3. Benediktsson, J. A., Swain, P. H. and Ersoy, O. K.: Neural network approaches versus statistical methods in classification of multisource remote sensing data, *IEEE Trans. Geosci. Remote Sensing*, **28**(4) (1990), 540–552.
4. Bosenberg, J., Brassington, D. and Simon, P. C. (eds.), *Instrument Development for Atmospheric Research and Monitoring*, Springer-Verlag: Berlin, 1997.
5. Durieux, E. and Fiorani, L.: Data processing. In: Bosenberg, J., Brassington, D. and Simon, P. C. (eds.), *Instrument Development for Atmospheric Research and Monitoring*, pp. 89–93, Springer: Berlin, 1997.
6. Fernandes, A., Utkin, A., Lavrov, A. and Vilar, R.: Classification of Lidar Signals by Committee Machines Applied to Automatic Forest Fire Detection. In: Wang, P. (ed.), *Proc. Seventh Joint Conf. Inf. Sci.*, pp. 1585–1588, Cary, USA, Association for Intelligent Machinery, 2003.
7. Gorman, R. P. and Sejnowsky, T. J.: Learned classification of sonar targets using a massively parallel network, *IEEE Trans. Acous. Speech Signal Process.*, **36**(7) (1988), 1135–1140.
8. Haykin, S. and Deng, C.: Classification of radar clutter using neural networks, *IEEE Trans. Neural Networks*, **2**(6) (1991), 589–600.
9. Haykin, S.: *Neural Networks*, pp. 117–155, Prentice Hall: London, 1999.
10. Haykin, S.: *Neural Networks*, p. 192 and 216, Prentice Hall: London, 1999a.
11. Jelalian, A. V.: *Laser Radar Systems*, Artech House: Boston, 1992.
12. Kuruoglu, E. E., Rayner, P. J. W. and Fitzgerald, W. J.: Least l_p-norm impulsive noise cancellation with polynomial filters, *Signal Process.*, **69** (1998), 1–14.
13. Magoulas, G. D., Vrahatis, M. N. and Androulakis, G. S.: Improving the convergence of the backpropagation algorithm using learning rate adaptation methods, *Neural Comput.*, **11** (1999), 1769–1796.
14. Measures, R. M.: *Laser Remote Sensing*, John Wiley & Sons: New York, 1984.
15. Moller, M. F.: A scaled conjugate gradient algorithm for fast supervised learning, *Neural Networks*, **6** (1993), 525–533.
16. Pal, P., Mukherjee, A., Acharya, S. and Das, J.: Continuous detection of atmospheric patterns from SODAR signals, *Signal Process.*, **74** (1999), 153–168.
17. Rauste, Y., Herland, E., Frelander, H., Soini, K., Kuoremaki, T. and Ruokari, A.: Satellite-based forest fire detection for fire control in boreal forests, *Int. J. Remote Sensing*, **18**(12) (1997), 2641–2656.

18. Riedmiller, M. and Braun, H.: A direct adaptive method for faster backpropagation learning: The RPROP algorithm, In: *Proc. IEEE Int. Conf. Neural Networks* (*ICNN*), San Francisco, pp. 586–591, 1993.
19. Ugarte, M. F., de Castro, A. J., Briz, S., Aranda, J. M. and Lópes, F.: Optimized geometry in infrared arrays for remote sensing of forest fires, *Infrared Phys. Technol.* **41** (2000), 35–39.
20. Utkin, A. B., Lavrov, A. V., Costa, L., Simões, F. and Vilar, R.: Detection of small forest fires by lidar, *Appl. Phys. B*, **74**(1) (2002), 77–83.
21. Utkin, A., Fernandes, A., Lavro, A., Simões, F. and Vilar, R.: Feasibility of forest-fire smoke detection using lidar, *Int. J. Wildland Fire* **12**(2) (2003), 159–166.
22. Vrahatis, M. N., Androulakis, G. S., Lambrinos, J. N. and Magoulas, G. D.: A class of gradient unconstrained minimisation algorithms with adaptive stepsize, *J. Comput. Appl. Math.* **114** (2000), 367–386.
23. Yu, X.-H., Chen, G.-A. and Cheng, S.-X.: Dynamic Learning Rate Optimization of the Backpropagation Algorithm, *IEEE Trans. Neural Networks*, **6**(3) (1995), 669–677.