Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 10 (13): 1331-1335, 2010
ISSN 1812-5654

Rotated Kernel Neural Networks for Radar Target Detection in Background Noise

A. Lotfi, K. Mezzoug and A. Benyettou
Department of Computer Science, Faculty of Sciences,
University of Sciences and Technology Mohamed Boudiaf USTO, P.O. Box 1505, El Menaour, Oran, Algeria

Abstract: This study presents the principle of operation of the Rotated Kernel Neural Network (RKNN) for radar target detection in non-Gaussian noise. This classifier is based on adopting the architecture of standard probabilistic neural networks and using different kernel functions to approximate density functions. The training algorithm for this classifier is more complicated than the original PNN training algorithm but allow better generalization. Performance curves of the Rotated Kernel Neural Network are compared to those of probabilistic neural networks (original), Radial Basis Neural Networks with an expectation maximization training algorithm and Back propagation neural networks for Radar target detection in background noise in terms of probability of detection versus signal-to-noise ratio (SNR). For most cases, the Rotated Kernel Neural Network classifier outperforms other conventional Radar target detection techniques and presents the advantage of resistance to background noise for values of SNR greater than 5 dB.

Key words: Probabilistic neural networks, signal processing, classification, generalization, weibull noise, lognormal noise

INTRODUCTION

During the last few years, neural networks have received the attention of many scientists. Due to their potential of solving difficult problems with no pre-knowledge of the methodology of the resolution (Ghwanmeh *et al*., 2006), they represent a powerful tool (Misra and Dehuri, 2007). Signal processing is one of the fields where the solution is not all the time evident (Zaknich, 2003) and becomes even more difficult to find when the background noise is taken into account. In signal processing, neural networks present the advantage of real time parallel processing (Samet and Miri, 2009), training with real data and adaptability. Probabilistic Neural Networks (PNN) perform well in signal processing (Specht, 1988), they have been used effectively to solve many problems like image classification, hand digit recognition and alphabet classification (Burcu and Tulay, 2006), but they suffer from one major drawback: the generalization of the network is poor due to the similar kernel function for all units (Galleske and Castellanos, 2002). Rotated Kernel Neural Networks can overcome this limitation by adopting the same architecture as probabilistic neural networks and using different kernels (instead of one) according to the shape of the different classes presented to the network. In this work, we applied RKNN to the problem of Radar target detection in non-Gaussian noise. Experiments show that RKNN outperforms the standard PNN in detecting targets hidden in a background noise.

PROBABILISTIC NEURAL NETWORKS

The probabilistic neural network was developed by Donald Specht. His network architecture was first presented in two papers (Specht, 1988). This network provides a general solution to pattern classification problems by following an approach developed in statistics, called Bayesian classifiers. Bayes theory, developed in the 1950's, takes into account the relative likelihood of events and uses a priori information to improve prediction. The network paradigm also uses Parzen Estimators, which were developed to construct the probability density functions required by Bayes theory (Shen and Yan, 2008).

The architecture of a probabilistic neural network as defined by Specht (1988) is shown in Fig. 1. This network is a special case of a multi-layer perceptron that has three layers:

- An input layer which has as many elements as there are separable parameters needed to describe the objects to be classified
- A pattern layer, which organizes the training set such that each input vector is represented by an individual processing element

Corresponding Author: Abdelhadi Lotfi, Rue Miloud Ben Mohamed, Gambetta, 31000 Oran, Algeria Tel: 00213772948902

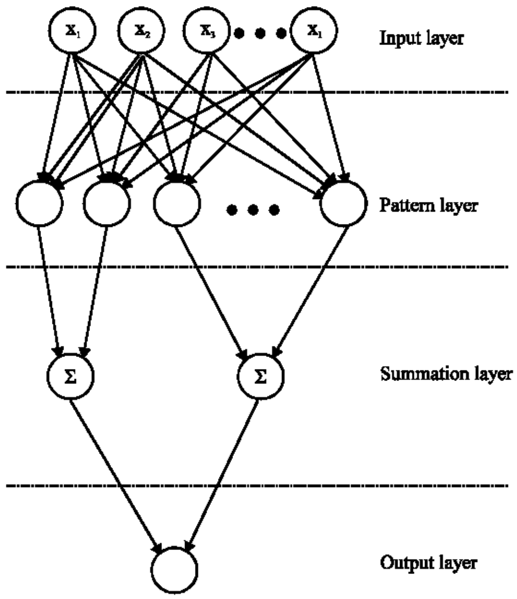

Fig. 1: Probabilistic neural network general architecture

- An output layer, called the summation layer, which has as many processing elements as there are classes to be recognized. Each element in this layer combines via processing elements within the pattern layer which relate to the same class and prepares that category for output

Sometimes a fourth layer is added to normalize the input vector, if the inputs are not already normalized before they enter the network.

The radial units are copied directly from the training data, one per case. Each one models a Gaussian function entered at the training case (Bolat and Yildirim, 2003). There is one output unit per class (Wahab *et al*., 2007). Each unit is connected to all the radial units belonging to its class, with zero connections from all other radial units. Hence, the output units simply add up the responses of the units belonging to their own class. The outputs are each proportional to the kernel-based estimates of the pdfs of the various classes and by normalizing these to sum to 1. 0 estimates of class probability are produced.

The pattern units use the following Gaussian activation function:

$$f(x) = e^{-\frac{(w_i - x_i)^t (w_i - x_i)}{2\sigma^2}} \quad (1)$$

where, w_i s represent the weights, x_i s represent the input vector elements and σ is a smoothing parameter (Fig. 2).

The training function may include a global smoothing factor to better generalize classification results

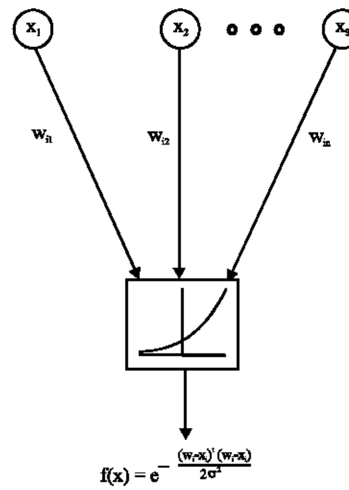

Fig. 2: The pattern unit in a PNN

(Zhong *et al*., 2002). In any case, the training vectors do not have to be in any special order in the training set, since the category of a particular vector is specified by the desired output of the input. The learning function simply selects the first untrained processing element in the correct output class and modifies its weights to match the training vector.

Rotated kernel probabilistic neural networks: The major problem in standard probabilistic neural networks is generalization which is poor because of Gaussian functions similar for all units. This gives classes with the same shape which is not the case in many real world data (as for Radar target data). The Rotated Kernel Probabilistic Neural Network (RKPNN) keeps the architecture of the original PNN and uses different Gaussian kernel functions for each unit and with different kernel parameters (Galleske and Castellanos, 2002). In this case, the classes represented by the network have different shapes and can easily fit the shape of the input data classes.

The general idea of this method is to divide the covariance matrix Σ_i into two other matrices S_i and S_i to obtain the parameters of the ith training pattern with the formula:

$$\sum_i = R_i^T \cdot S_i \cdot S_i \cdot R_i \quad (2)$$

where, R_i is the rotation matrix and S_i is a diagonal matrix.

The kernel parameters of the network are estimated automatically in the training process and have not to be chosen to suit a specific classification problem like original probabilistic neural networks do.

RADAR DETECTION

The Radar target detection problem can be seen as testing two hypotheses H_0 and H_1:

$$\begin{aligned} &H_1 : \\ &H_1 : x = s_i + n_i \\ &H_0 : x = n_i, \qquad i = 1, 2, \ldots, n \end{aligned} \tag{3}$$

where, x's are the samples of the waveform, s_i's are samples of the signal and n_i's are samples of the background noise (McDonough and Whalen, 1995).

We can then calculate the maximum likelihood ratio L(r) and compare it to a threshold as follows:

$$L(n) = \frac{f(x/H_1)}{f(x/H_0)} \underset{>}{<} th \tag{4}$$

Where:
$f(x/H_1)$ = The conditional density function of x given H_1 is true
$f(x/H_0)$ = The conditional density function of x given H_0 is true

To simplify, we take a threshold = 1 in this document. So, the decision rule becomes:

- H_1 is true $\Rightarrow f(x/H_1)>f(x/H_0)$
- H_0 is true $\Rightarrow f(x/H_0)>f(x/H_1)$

The probability of detection P_d is:

$$P_d = \int_{th}^{\infty} f(x/H_1)\,dx \tag{5}$$

The probability of false alarm is given by:

$$P_{fa} = \int_{th}^{\infty} f(x/H_0)\,dx \tag{6}$$

From Eq. 4, we can say that if we want to make a decision, we have to calculate $f(x/H_1)$ and $f(x/H_0)$. Specht has proposed a probabilistic neural network based Baysian classifier for Radar detection (Specht, 1988).

RESULTS

The project was conducted in SIMPA laboratory at the University of Science and Technology of Oran during the last semester of 2009.

Performance of the RKNN for radar target detection in background noise (unwanted clutter) (Vassileva, 2006) has been evaluated in terms of probability of detection versus Signal to Noise Ratio (SNR) in different environments (noises). The classifier is also compared to MLP, PNN and RBF neural networks for the same conditions. The RBF neural network is used with and EM (Expectation Maximization) training algorithm for better performance. This network is widely used for classification and approximation problems with different alternatives (Lu and Ye, 2007; Alippi *et al*., 2001) and it can be used even with high-dimensional problems (Joo *et al*., 2002). For all networks we use a window size of 15 points. The training samples are generated from -10 to 20 dB SNR (signal to noise ratio) with 100 samples for each 1 dB. The probability of false alarm is set to 10^{-5}; we can find this value in many radar references (McDonough and Whalen, 1995). For all these results we used Matlab for the training, testing and simulation of results. This latter presents the advantage of incorporated functions for signal and neural network processing. Hence, it is easier to test our technique in comparison with other techniques without wasting a long time in programming every single method.

DISCUSSION

In log-normal noise

- **For σ = 0.1:** The RKNN detector outperforms the other detectors for values of SNR>8 dB and reaches a probability of detection of 1 (certainty) when the SNR is greater than 16 dB (Fig. 3). However, this is not the case for SNR<8 dB where the RKNN have poor performance in front of the other detectors

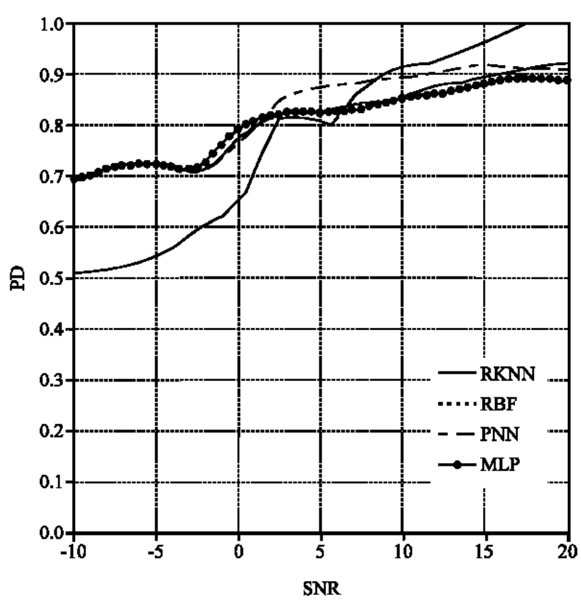

Fig. 3: Performance curves in log-normal noise σ = 0.1

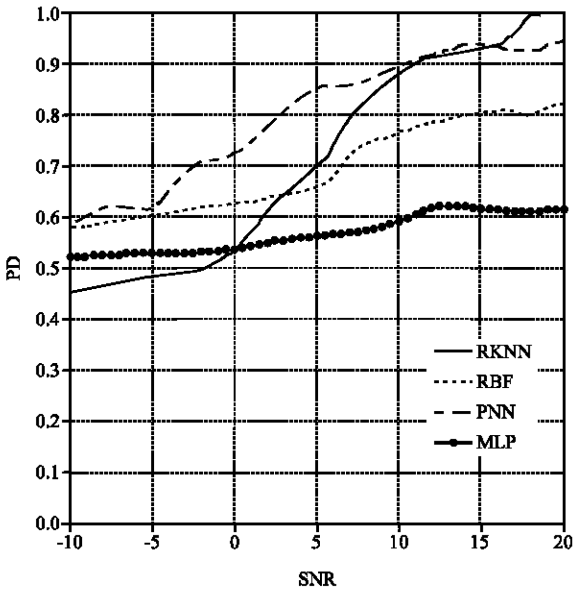

Fig. 4: Performance curves in log-normal noise σ = 0.5

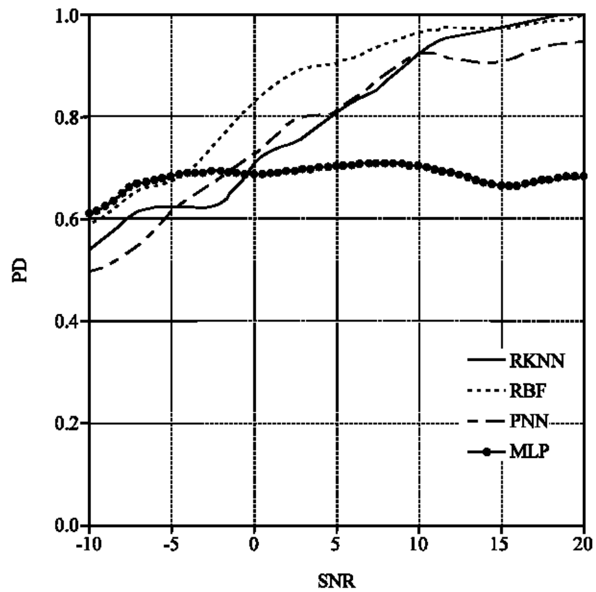

Fig. 5: Performance curves in Weibull noise α = 0.75

The other techniques behave the same way for all values of SNR but do not reach a probability of detection of 0.9 except for the PNN when SNR>12 bB. RBFNN and MLPNN are quite similar in this case.

- **For σ = 0.5:** RKNN and PNN behave the same way as for the preceding case (σ = 0.1). PNN gives better performance for SNR<15 dB. The two networks reach 0.9 for SNR>10 dB (Fig. 4)

Performance of the MLP decreases significantly in this case. The network has poor results in this type of noise. This may be caused by a wrong choice of the network's initial parameters. This problem is one of the most important disadvantages of MLP neural networks. There is actually no rule when it comes to choosing the parameters except repeating tests until satisfaction.

Weibull noise: In Weibull noise, it is obvious that RKNN outperforms other neural network based detectors (Fig. 6) when the SNR is greater than 5 dB. But when the SNR is less than 5 dB, the response is not really in favour of the RKNN since the probability of detection is not very reliable (i.e., the system can not decide whether there is a target or not, unless we accept a high probability of false alarm). In practice this has a great impact on the detecting system, since we speak no more about a human-independent (automatic) system, but about a human-assisted detection.

From Fig. 5, we can observe that the best results were generated by the RBF neural network for almost all values of SNR. The RKNN has better performance for values of SNR greater than 15 dB.

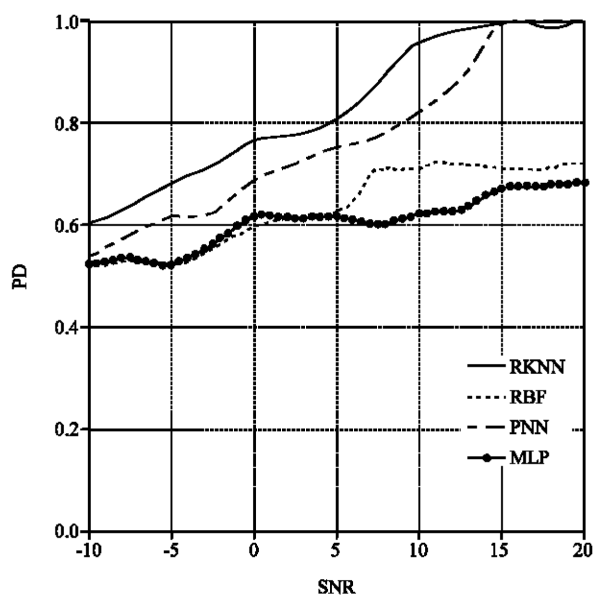

Fig. 6: Performance curves in Weibull noise α = 1

In this case of noise, we can see that results presented by the RBF neural network with an EM training algorithm are good and close to those of a RKNN in some cases.

Both RKNN and RBF networks presented here give better performance for low values of SNR and outperform most traditional Radar target detectors in presence of background unwanted clutter (Vassileva, 2006; Sangston and Gerlach, 1994). The problem here is that the clutter signals are often as strong as or stronger than the signals returned from the desired target and makes it difficult to separate the two wave forms.

In many applications it is common for the SNR to be low. In these situations, accurate and robust estimation of features from the spectrogram or its derivatives is very difficult and leads to poor performance (Gurbuz *et al.*, 2007).

CONCLUSION

When we talk about Radar target detection in non Gaussian noise, the methodology of resolution is most of time not evident. In this document we presented the RKNNs applied to Radar target detection. Simulation results presented here are very promising and show that neural networks can be applied successfully where other techniques fail or find serious difficulties due to the complexity of the problem (Vassileva, 2006). For almost all cases presented in this document, the RKNN has shown a stable behaviour when the other classifiers behaved according to the nature of the noise. Hence, the RKNN can be applied to radar detection regardless to the nature of the noise. This point should be addressed in more depth for further studies. The use of a boosting algorithm with probabilistic neural networks can increase and smooth performance. Results given by the RBF are also good and deserve more attention especially with the EM training algorithm. Many neural networks can be used with a modified boosting algorithm to increase performance and overcome some of their weaknesses (Bolat and Yildirim, 2003).

ACKNOWLEDGMENTS

The authors would like to thank all members of the SIMPA laboratory at the University of Sciences and Technology for their support and great patience.

REFERENCES

Alippi, C., V. Piuriand and F. Scotti, 2001. Accuracy vs. complexity in RBF neural networks: A system level design approach. IEEE Instrumentat. Measurement Magazine, 2: 32-36.

Bolat, B. and T. Yildirim, 2003. Performance increasing methods for probabilistic neural networks. Inform. Technol. J., 2: 250-255.

Burcu, E. and Y. Tulay, 2006. Statistical Neural Network Based Classifiers for Letter Recognition. In: Intelligent Computing in Signal Processing and Pattern Recognition, Huang, D.S., K. Li and G.W. Irwin (Eds.). Springer-Verlag, Berlin Heidelberg, pp: 1081-1086.

Galleske, I. and J. Castellanos, 2002. Optimization of the kernel functions in a probabilistic neural network analyzing the local pattern distribution. Neural Comput., 14: 1183-1194.

Ghwanmeh, S., R. Al-Shalabi, G. Kanan and L. Alnemi, 2006. Enhanced neural networks model based on a single layer linear counterpropagation for prediction and function approximation. Inform. Technol. J., 5: 45-50.

Gurbuz, S.Z., W.L. Melvin and D.B. Williams, 2007. Detection and identification of human targets in radar data. Proc. SPIE Int. Soc. Opt. Eng., 6567: 65670I-65670I.

Joo, M., E.S. Wu, J. Lu and H.L. Toh, 2002. Face recognition with radial basis function (RBF) neural networks. IEEE Trans. Neural Networks, 13: 697-710.

Lu, Y. and M. Ye, 2007. Oracle model based on RBF neural networks for automated software testing. Inform. Technol. J., 6: 469-474.

McDonough, R.N. and A.D. Whalen, 1995. Detection of Signals in Noise. 2nd Edn., Acadamic Press, London, Orlando, pp: 206.

Misra, B.B. and S. Dehuri, 2007. Functional link artificial neural network for classification task in data mining. J. Comput. Sci., 3: 948-955.

Samet, S. and A. Miri, 2009. Privacy-preserving bayesian network for horizontally partitioned data. Int. Conf. Comput. Sci. Eng., 3: 9-16.

Sangston, K.J. and K.R. Gerlach, 1994. Coherent detection of radar targets in a non-gaussian background IEEE Trans. Aerospace Elect. Syst., 30: 330-340.

Shen, H. and X.L. Yan, 2008. Probability density estimation over evolving data streams using tilted Parzen Window. Proceedings of the 13th IEEE Symposium on Computers and Communications, July 6-9, Marrakech, Morocco, pp: 1-21.

Specht, D.F., 1988. Probabilistic neural networks for classification, mapping or associative memory. Proc. IEEE Int. Conf. Neural Networks, 1: 523-525.

Vassileva, B., 2006. VHF radar target detection in the presence of clutter. Cybernetics Inform. Technol., 1: 99-106.

Wahab, N.I.A., A. Mohamed and A. Hussain, 2007. Transient stability assessment of a power system using PNN and LS-SVM methods. J. Applied Sci., 7: 3208-3216.

Zaknich, A., 2003. Neural Networks for Intelligent Signal Processing: Series on Innovative Intelligence, Vol. 4, World Scientific Pub. Co. Inc., New Jersey, USA., ISBN: 978-981-279-685-1, pp: 14-17.

Zhong, M., D. Coggeshall, E. Ghaneie, T. Pope and M. Rivera *et al.*, 2002. Gap-based estimation: Choosing the smoothing parameters for probabilistic and general regression neural networks. Neural Comput., 19: 2840-2864.