ESANN'2005 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 27-29 April 2005, d-side publi., ISBN 2-930307-05-6.

# UWB radar target identification based on linear RBFNN

Min Wang[1], Shuyuan Yang[2], Shunjun Wu[1]
[1] National Lab of Radar Signal Processing, Department of Electrical Engineering
Xidian University Xi'an,China, 710071
[2] Institute of Intelligence Information Processing, Department of Electrical Engineering
Xidian University Xi'an,China, 710071

**Abstract.** In this paper, a radical-basis-function neural network(RBFNN) with efficient linear learning algorithm is presented for the identification on target profiles of Ultra Wideband(UWB) radar. This linear RBFNN has both good localization approximation and linear computation complexity with the number of dimension and number of inputs. Its performance is comparable with support vector machine (SVM) for tasks of pattern recognition with a rapider speed. We applied it to the identification of target profile in UWB radar, which needs excessively fast processing. The experimental results are achieved including higher recognition rate and shorter consumed time, which is superior to its counterparts.

## 1 Introduction

As a popular model in the community of artificial neural networks, radical-basis-function neural network (RBFNN) has attracted intense researching interests [1]. It is characterized of universal approximations, compact topology and fast learning, so it has found many applications in diverse engineering fields [2]. The basic architecture of RBFNN is a three-layer network. Each layer is fully connected to the following one and the hidden layer is composed of a number of nodes with radial activation functions called radial basis functions. The input layer is simply a fan-out layer. The second (or hidden layer) can fulfill a non-linear mapping from the input space into a (usually) higher dimensional space in which the patterns become linearly separable. The final layer is only a linear weighted output. Up to now, many learning algorithms for RBFNN have been proposed. However, they are commonly of high computation complexity that is related with the number of input samples and hidden neurons. For example, the complexity of gradient leaning algorithm approaches $O(n^3)$ (where $n$ is the number of training samples) due to the computation of an inverse matrix. Orr proposed a number of approaches to reduce the hidden units [3]. Beatson proposed a $O(nlogn)$ learning algorithm using polyharmonic spline functions [4], which make an improvement of the classical RBFNN. To achieve the goal of a fast processing in the identification on target profiles of Ultra Wideband(UWB) radar, a linear RBFNN (LRBFNN) with linear learning is presented in this paper. It is characteristic of a linear computation complexity in time and space with the dimension and number of the input samples. So it is of rapid processing and good performance in the target identification of UWB radar system. The comparison results with other traditional methods also prove its superiority.

# 2 A Linear RBFNN with efficient learning algorithm

## 2.1 The structure and learning of RBFNN

The architecture of a *L-M-N* RBFNN is shown in figure 1, whose input layer, hidden layer and output layer have *L*, *M* and *N* neurons respectively. Denote $r_i = \|X - C_i\|$ as the distance of input X=[$x_1, x_2, \ldots, x_L$] to the *i*-th center $C_i$=[$c_{i1}, c_{i2}, \ldots, c_{iL}$], where ||.|| indicates the Euclidean norm on the input space. Usually the Gaussian function Φ is preferred among all possible radial basis functions due to the fact that it is factorizable. So the output of the hidden layer is $\boldsymbol{Z} = [z_1, z_2, .. z_M]^T$ with each component $z_i$:

$$z_i = \Phi(r_i) = \exp[-\sum_{j=1}^{L} \frac{(x_j - c_{ij})^2}{2\sigma_i^2}] \quad (i=1,\ldots,M) \tag{1}$$

where $\sigma_i$ is the radius of the *i*-th Gaussian function in the hidden layer, and $c_{ij}$ is the *j*-th component of the *i*-th center. Hence the input vectors are mapped to higher dimension by the radical basis function. A linear layer then follows and we get the output Y=[$y_1, y_2, \ldots, y_N$], where with each component

$$y_i = w_{i0} + \sum_{j=1}^{M} w_{ij} Z_j \quad (i=1,\ldots,N) \tag{2}$$

where $y_i$ is the output of *i*-th neuron, $w_{ij}$ is the connect weight of the *j*-th hidden neuron to the *i*-th output neuron and $w_{i0}$ is the threshold of the *i*-th output neuron.

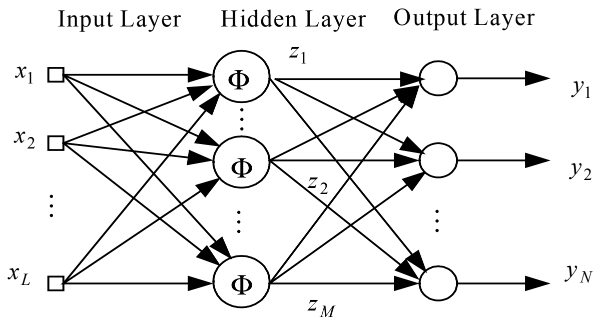


Fig. 1: The structure of a RBFNN

As we have described above, R=[$r_1, r_2, .. r_M$] is the distance between the input and the center measured by the Euclidean norm. Therefore, the network receives an input pattern X and the centers are known, the distance from the centers can be measured. Furthermore, this distance measure is non-linear, so that a pattern can give a value close to 1 if it is in an area that is close to a center, however beyond this area, the value drops dramatically. $\sigma$ defines the width or radius of Gaussian function and is something that has to be determined cautiously. Because a linear combination of spherical Gaussian functions can approximate any function with arbitrarily small error, the training task is reduced to searching for appropriate centers, radius and connected weights for the network. To determine the parameters of the Gaussian functions, supervised and unsupervised learning algorithm can both be employed. Gradient descent technique is a very popular supervised learning algorithm, and there are also some unsupervised methods such as K-means algorithm, Kohonen algorithm and P-nearest neighbour algorithm. Kohonen algorithm and K-means algorithm are both clustering methods used to determine the centers. P-nearest neighbour algorithm is employed to determine the radius of the Gaussian function. Having trained the hidden layer, the next step is to train the output layer, that is, find the weights between hidden and output layer. Gradient-based methods are usually used such as least mean squares algorithm.

## 2.2 Leaning algorithm based on Kernel Smoothing

In this section, we discuss a linear algorithm based on the classical kernel smoothing method for RBFNN[5], which has linear computation complexity with the number and dimension of training samples. The basic idea of kernel smoothing algorithm is to place a Gaussian function at each sample. Denote the distance between two samples in $R^N$ as $\boldsymbol{h}\delta=(h_1\delta,h_2\delta\ldots,h_N\delta)$, where $\delta$ is a very small real number and $h_1, \ldots h_N$ are all integers. Suppose one sample $x_h$ has $k$ nearest neighbours $x_h^1$, $x_h^2$,.., $x_h^k$. When the samples are very dense, that is, $\delta$ approaches 0, the function values of the $k$ neighbours $x_h^1$, $x_h^2$,.., $x_h^k$ are approximately equal to that of $x_h$, that is,

$$f(x_h^1)\approx f(x_h^2) \approx\cdots\approx f(x_h^k) \approx f(x_h)$$

Lay a Gaussian function at each sample, then the radius of Gaussian functions and the corresponding weights in output layer are approximately equal: $w_h^1\approx w_h^2\cdots\approx w_h^k\approx w_h$, $\sigma_h^1 \approx\sigma_h^2\cdots\approx\sigma_h^k\approx\sigma_h$ . Denote

$$q(x)=\sum_{h=-\infty}^{\infty}\exp\left(-\frac{\|x-h\delta\|^2}{2\sigma_i^2}\right) \tag{3}$$

$q(x)$ proves to be bounded by $2.50662827\pm1.35\times10^{-8}$ if $\sigma_i=\delta$ [5], then we get:

$$w_i.\{Min[q(x)]\}^N \le w_i[\sum_{h_1=-\infty}^{\infty}\ldots\sum_{h_N=-\infty}^{\infty}\exp(-\frac{\|x-(h_1\delta,\ldots,h_N\delta)\|^2}{2\delta^2})]\le w_i.\{Max[q(x)]\}^N \tag{4}$$

If a sample is located at $x^i$, the weights of its corresponding Gaussian function is approximately $w_i=f(x^i)/(2.5066)^N$, which is of linear learning complexity. Such result is under an assumption of $\sigma_i=\delta$, and we can describe it in a more generalized way:

$$\hat{f}(x)=\frac{1}{\lambda^N}\sum_{x^i}f(x^i)\cdot\exp(-\frac{\|x-x^i\|^2}{2\beta^2\delta^2}) \tag{5}$$

where $\lambda=\sum_{j=-\infty}^{\infty}\exp(-j^2/2\beta^2)$, $\beta=\sigma_i/\delta$ . From above we can see that the kernel smoothing function is actually a weighted average of the sampled function values. Therefore, selecting a larger value $\beta$ implies that the smoothing effect will be more significant. According to our experimental results, the value of $\beta$ essentially has no effect on the result, as long as it is set to a value within [0.6,2]. Our suggestion is set $\beta=1$. Then the equation of the network can be written as:

$$\hat{f}(x_l) = \frac{1}{(2.5066)^N} \sum_l f(x_l) \cdot \exp\left( -\frac{\|x_l - h\delta\|^2}{2\delta^2} \right) \qquad l = 1,..,n \tag{6}$$

where $n$ is the number of training samples and $N$ is the dimension of the input space. Then we can get the time complexity of constructing a RBFNN—$O(Nn)$, and the time complexity for predicting the function values of $m$ test samples is $O(Nm)$.

### 2.3 Discussion of the linear RBFNN

Such a learning algorithm is very efficient because it has time and space complexity linear with the dimension and number of the input samples. The algorithm has been proved to exhibit good results in pattern recognition and regression, at least delivering the same level of accuracy as the available efficient tools, such as support vector machine (SVM) [5].

Although the proposed algorithm has comparable performance with existing algorithms, the discussions so far are based on the assumption that the sampling density is sufficiently high, i.e, in the case of δ approaching 0, which may not hold for some practical data sets. So firstly if the assumption of uniform sampling does not hold, then some sort of interpolation can be conducted to obtain the approximate function values at the crosses of the grid formed by $\delta^N$. Secondly, since the above algorithm is distance-based, removal of redundant training samples will lower the complexity of RBFNN when the input dimension is very high. So it is interesting to develop some sort of data reduction mechanisms to reduce the space complexity of the RBFNN constructed. Thirdly, for the existence of noises in practical cases, some measures should be considered to handle random noises and the compensation of smoothing effect. Some further work which aim to solve the mentioned problems are being done by our group.

## 3 Target Identification for UWB radar using Linear RBFNN

Ultra Wideband (UWB) technology has been successfully applied in wireless communication and radar system in recent years. It can fulfill the transmission and reception of high-peak power using extremely short impulses [6]. For example, the UWB radar system developed at the Radar Systems Lab (University of Kansas) is a system with a bandwidth of 1.775 GHz and a pulse duration of less than 1 ns. Using ultra-wideband (UWB) radar signals appears to be the most promising approach to building radar systems with new and better capabilities and direct applications to civil uses and environmental monitoring. UWB signals can provide high range resolution for better imaging than available narrowband synthetic-aperture radar (SAR) systems.

Recently the identification on target profile of UWB radar is a challenging subject in radar signal processing, which anticipate a higher recognition rate for very similar and small targets. The targets can be recognized using one-dimensional or two-dimensional image. Figure 2 shows the 2-D image of three kinds of planes—B-1, MI-8 and MI-6. A one-dimensional image is a vector sum of all echoes of corresponding two-dimensional image on perpendicular direction. Commonly speaking, a two-dimensional image is much easier to be recognized, but in practical it is difficult to acquire it. It is well known that a one-dimensional image (or the range profile of radar) can represent a spatial distribution of microwave reflectivity, which is sufficient to characterize the targets [7][8]. So in actual radar recognition system we only use one-dimensional image. RBFNN has got successful application in radar target identification in recent years. Unlike the traditional radar systems, UWB radar needs excessively rapid processing for its very short impulses, which brings about more samples in a defined time duration or higher dimensional radar image. However, the training complexity in time and space of most available RBFNN increase exponentially with the increased dimension and number of input samples. So for target identification of UWB radar, where the images of targets are of high dimension, a real processing can't be accomplished. However the linear RBFNN has a linear relation with the dimension and number of the input space, so it can achieve a fast processing speed required by UWB radar identification.

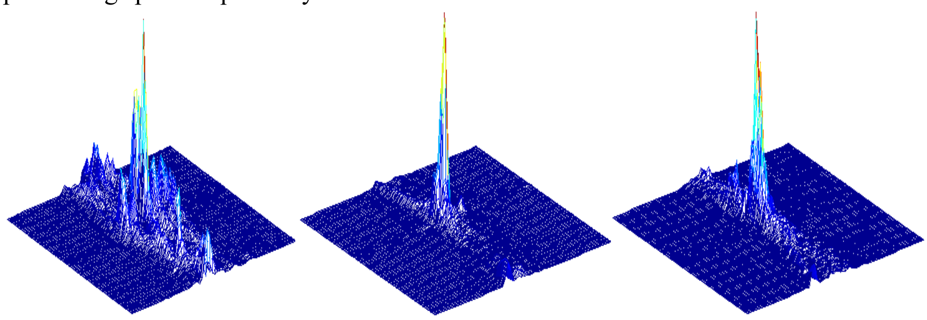

Fig. 2: The 2-D image of B-1, MI-8 and MI-6

## 4 Simulation Results

We used the linear RBFNN above to recognize the one-dimensional image of three-class planes whose models are B-1, MI-8 and MI-6 respectively (their 2-d images are shown in figure 2). Our data are obtained in a microwave darkroom with imaging angle from 0 to 179 degree. Totally we get 322, 311 and 451 images of three classes of plane respectively. Here the input space of one-dimensional image recognition is of 100 dimensions. In the model, we let $\beta$=1 and $\delta$=0.01. To get better learning and generalization of the network, we normalized the data to [0,1] before feeding them into the network. 100 samples are taken from each class to form training samples. The totally 1084 images are taken as the test samples. We compare our method with a generalized RBFNN (GRBFNN) that adaptively adjusts all the parameters using a supervised gradient descent algorithm for all the parameters of the network. At the same time, the Gaussian SVM algorithm—a very good classification method is also considered to give its result for this problem. Adjust the parameters in three methods to make the training error small enough (with a recognition rate>95%). For the three methods, GRBFNN expends a long training time and finally get an unstable result; the recognition rate of SVM is higher with shorter training time than GRBFNN; while our method consumes least time and obtains a relatively good recognition rate comparable to SVM. The results of recognition rates and the consumed time of three methods are shown in table 1. From it we can see that our network can recognize these planes with high recognition rate. Moreover, it is characteristic of a rapid learning speed, which is preferred in practical processing.

| MODEL | Identification Rates | | |
|---|---|---|---|
| | GRBFNN | SVM | LRBFNN |
| B-1 | 96.3% | 98.0% | 97.8% |
| MI-8 | 93.2% | 95.5% | 95.6% |
| MI-6 | 94.6% | 96.1% | 96.9% |
| Time(s) | 9.3 | 5.0 | 2.2 |

Table 1: Identification results of three planes

# 5 Conclusion

In this paper, a linear RBFNN with an efficient learning algorithm is applied to the identification on target profiles of Ultra Wideband radar. The linear RBFNN is based on the classical kernel smoothing method and it has the linear complexity in time and space with the dimension and number of input data. Accordingly it is characteristic of fast learning and relative high accuracy in the UWB identification system, which is demonstrated through the comparison results with other traditional methods.

# Reference

[1] J. Park, I. W. Sandberg, Universal approximation using radial-basis-function networks, *Neural Computation*,Vol.3,No.2, pp.246-257,1991.

[2] L C. Jiao, Neural Network System Theory. (in Chinese) *Xidian University press*, 1990.

[3] Orr, M.J.L., Introduction to radial basis function networks. *Centre for Cognitive Science, University of Edinburgh,* April. 1996.

[4] Beatson, R. K., Cherrie, J. B., and Ragozin, D. L. Fast evaluation of radial basis functions: Methods for four-dimensional polyharmonic splines. *SIAM J. Math. Anal*. Vol 32, No 6, pp1272-1310. 2001.

[5] Yen-Jen Oyang., An Efficient Learning Algorithm for Function Approximation with Radial Basis Function Networks, *Proceedings of the Ninth International Conference on Neural Information processing*, Singapore, 2002.

[6] Malek G.M. Hussain,Principles of Space-Time Array Processing for Ultrawide-Band Impulse Radar and Radio Communications, *IEEE Transations on Vehicular Technology*, Vol.51, No.3, May 2002.

[7] S. Huden and D. Psaltis, Correlation filters for aircraft identification from radar range profile. *IEEE Trans. on Aerospace and Electronic Systems*, Vol.29, No.3, pp741-748, July 1993.

[8] M. R. Inggs and A.D. Robinson, Ship Targe Recongnition Using Low Resolution Radar and Neural Network. *IEEE Trans. on Aerospace and Electonic Systems*, Vol.35, No.2, pp 386-392, Apr.1999.