Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

Automatic Target Recognition in SAR Imagery using a MLP Neural Network

N.M. Sandirasegaram

Defence R&D Canada - Ottawa
TECHNICAL MEMORANDUM
DRDC Ottawa TM 2002-120
November 2002

Canada

Automatic Target Recognition in SAR Imagery using a MLP Neural Network

N.M. Sandirasegaram
DRDC Ottawa

Defence R&D Canada - Ottawa

Technical Memorandum

DRDC Ottawa TM 2002-120

November 2002

Abstract

In this report, a Multi Layer Perceptron (MLP) Neural Network is used for recognizing military ground vehicles imaged by Synthetic Aperture Radar (SAR). In particular, the classifier is applied to SAR images taken from the MSTAR (Moving and Stationary Target Acquisition and Recognition) data set, which has been made available to the public. Signatures are extracted from the imagery using a Fourier Transform method and features are selected to feed the neural network. A 4-layer (including input and output layers) Neural Network with 38 input nodes, 13 first hidden nodes, 11 second hidden nodes and 3 output nodes, is implemented for this task. Standard delta rule back-propagation algorithm has been used to train the neural network. The MLP neural network is evaluated according to the MSTAR standard evaluation criteria. Training of 3 vehicle classes occurs using a set of SAR images at a 17-degree depression angle with 0-360 degree azimuthal angles, while the testing set contains images at a 15-degree depression angle with 0-360 degree azimuthal angles. The testing set contains both target vehicles that belong to the 3 trained classes and confuser vehicles that do not. Results of MLP neural network evaluation are shown using Receiver Operating Characteristic (ROC) curves and Confusion Matrices.

Résumé

Dans ce rapport, un réseau neuronal perceptron multicouches est utilisé pour la reconnaissance de véhicules terrestres militaires vus par un radar à antenne synthétique (RAS). Plus particulièrement, le classificateur est appliqué à des images RAS de l’ensemble de données MSTAR (Moving and Stationary Target Acquisition and Recognition = acquisition et reconnaissance de cibles mobiles et fixes), qui a été rendu public. Les signatures sont extraites des images au moyen d’une méthode de transformées de Fourier et des caractéristiques sont sélectionnées aux fins du réseau neuronal. Un réseau neuronal à 4 couches (y compris les couches d’entrée et de sortie) avec 38 nœuds d’entrée, 13 premiers nœuds cachés, 11 seconds nœuds cachés et 3 nœuds de sortie, est mis en œuvre pour cette tâche. Un algorithme de rétropropagation du gradient à règle delta standard a été utilisé pour l’entraînement du réseau neuronal. Le réseau neuronal perceptron multicouches est évalué en fonction des critères d’évaluation standard du MSTAR. L’entraînement pour 3 classes de véhicules se fait à l’aide d’un ensemble d’images RAS à un angle de dépression de 17 degrés avec des angles d’azimut de 0 à 360 degrés, tandis que l’ensemble d’essai contient des images à un angle de dépression de 15 degrés avec des angles d’azimut de 0 à 360 degrés. L’ensemble d’essai contient à la fois les véhicules qui font partie des 3 classes de véhicules visées et des véhicules trompe-l’œil qui n’en font pas partie. Les résultats de l’évaluation du réseau neuronal perceptron multicouches sont montrés au moyen de courbes de fonction d’efficacité du récepteur et de grilles de correction.

This page intentionally left blank.

Executive summary

Synthetic Aperture Radar (SAR) based automated target recognition (ATR) system requires a fast and effective classifier to discriminate desired types of targets from man-made targets, natural clutter and background noise. There are many classifiers existing in the filed of ATR and they all have advantages and disadvantages. The Multi Layer Perceptron (MLP) Neural Network has been successfully used in other applications such as mining, signal processing, pattern recognition etc. The author applied an MLP Neural Network to SAR images of military vehicles, and showed the performance result on the MSTAR public data set.

The MLP Neural Network is evaluated using Receiver Operating Characteristic (ROC) curves and Confusion Matrices on the publicly released MSTAR data set. The results of these evaluations are listed under Results and Discussion section and the percent of correct classification of declared targets is about 85%. The rate of correct classification of declared target can be improve by choosing alternative methods for feature extraction and revisiting the architecture of the MLP Neural Network.

Sandirasegaram N. (2002). Automatic Target Recognition in SAR Imagery using a MLP Neural Network. DRDC Ottawa TM 2002-120. Defence R&D Canada - Ottawa.

Sommaire

Le système de reconnaissance de cibles automatisé (RCA) fondé sur le radar à antenne synthétique (RAS) nécessite un classificateur rapide et efficace pour établir une distinction entre les types de cibles désirés, et les cibles artificielles, le fouillis d'échos naturel et le bruit de fond. Il existe un grand nombre de classificateurs dans le domaine de la reconnaissance de cibles automatisée, et ils possèdent tous des avantages et des inconvénients. Le réseau neuronal perceptron multicouches a été utilisé avec succès dans d'autres applications telles que l'exploitation minière, le traitement de signaux, la reconnaissance de formes, etc. L'auteur a appliqué un réseau neuronal perceptron multicouches à des images RAS de véhicules militaires, et a montré le résultat de performance sur l'ensemble de données public MSTAR.

Le réseau neuronal perceptron multicouches est évalué au moyen de courbes de fonction d'efficacité du récepteur et de grilles de correction sur l'ensemble de données public MSTAR. Les résultats de ces évaluations sont indiqués à la section « Results and Discussion » et le pourcentage de classification correcte de cibles déclarées est d'environ 85 %. Le taux de classification correcte de cibles déclarées peut être amélioré par le choix de méthodes de rechange pour l'extraction de caractéristiques et par la révision de l'architecture du réseau neuronal perceptron multicouches.

Sandirasegaram N. (2002). Automatic Target Recognition in SAR Imagery using a MLP Neural Network. DRDC Ottawa TM 2002-120. R&D pour la défense Canada - Ottawa.

Table of contents

List of figures

List of tables

Acknowledgements

I would like to thank Dr. Ryan English for helping me in technical and non-technical problems that I have faced during this task. Also I thank him for letting me to use his Matlab code to extract data from MSTAR image chips.

This page intentionally left blank.

1. Introduction

The area of Automatic Target Recognition (ATR) for SAR imagery is an ongoing research in many braches of the military and large research institutions [1]. General functions of ATR such as target detection, classification, etc. can be found more in details in [2] and [3]. The U.S. Defense Advanced Research Projects Agency (DARPA) has made part of the Moving and Stationary Target Acquisition and Recognition (MSTAR) data set available to the public. The MSTAR public data set contains many spotlight SAR vehicle images including 10 types of former Soviet Union vehicles with 0 to 360 degrees azimuthal angle and the depression angle of 15 and 17degrees.

The COMPASE Center of AFRL developed a standard MSTAR evaluation methodology to evaluate the ATR algorithms using the MSTAR public data set [4]. The standard evaluation method uses Confusion Matrix and Receiver Operating Characteristic (ROC) curves to evaluate the algorithms as was done for the other ATR algorithms [5,6,7] (such as HNeT, template matching, etc.). Here this method is also used to evaluate a Neural Network algorithm.

Neural Networks have been successfully applied to classification problems in the areas of industry, business and science [8]. Neural Networks and statistical classifiers have been compared in different applications [9, 10, 11]. Statistical methods need more space to store all the training data and also they often work very slowly compared to Neural Network (NN) classifier [10]. Maximum Likelihood and NN classifiers are compared by Fauzi *et al.* in characterizing the condition of logged over and unlogged tropical rain forest using satellite remotely sensed data and they show that overall accuracies of NN is better than Maximum Likelihood [11]. Both classifiers are applied to and compared for the same data (multisource remote sensing data) by Benediktsson *et al.*, who determined a three layer NN is more appropriate in multisource classification if the training time is in a reasonable amount of time [9]. However, a NN has a weakness when the size of the training samples become large, the training time can be very long [9]. Additional details about NN technologies for ATR can be found in [12].

Neural Networks attempt to copy abilities of biological neurons [13] and the reader can find an explanation of differences between biological neuron and artificial neuron in [13, 14,15]. The first NN model was introduced by McCulloch and Pitts [16, 17] in the1940s and is still an area of active research today. A NN is characterized by network architecture, node properties, learning rules and connections between neurons [15]. The reader can find different structured or characterized NNs in [13,15]. The most commonly used nonlinear regression and discriminant model is the Multi Layer Perceptron NN [18], which is capable of learning nonlinear function mappings [15]. Multi Layer Perceptron (MLP) NN has been used extensively in various problems [19] more than any other NN [13]. The typical MLP NN is build with an input layer, an output layer and at least one hidden layer [13]. Here, we evaluate an MLP NN ATR algorithm applied to the MSTAR public data set. In this case, a 4-layer MLP neural network is implemented with 38 input nodes, 13 first hidden nodes, 11 second hidden nodes and 3 output nodes. The standard delta rule back-propagation training method is used to train the implemented neural network, allowing it to learn about specific vehicle types from the training set and then, when the testing set is introduced, the neural network is able to predict a classification based on the knowledge learned from the training set.

Preprocessing of the imagery begins by taking a 64 x 64 pixel block from the center of each chip target for Fourier feature extraction, which is explained in Section 2. Section 3 provides an overview of the MLP NN training and testing algorithms. The MSTAR data set used for training and testing is described in the section 4 and the evaluation of results and discussion are given in section 5. Finally, the conclusions are given in section 6.

2. Feature Extraction

Feature extraction is a necessary step in the classification process. It is a preprocessing technique to standardize information provided to the classifier. In addition, it increases the training speed and testing speed of the classifier, as well as reducing the size of the training samples. This feature extraction method is inspired by the one used by HNeT [5]. In the MSTAR problem, there are three classes of former Soviet Union's military vehicles considered and they are BMP2, BTR70 and T72. For this problem, the HNeT classifier is implemented using three binary classifiers, one for each class, and uses 256 features for each class. But here, only one NN classifier is implemented for all the three classes (multi classifier) and 16 features are used instead of 256 features for each class in the HNeT. That is, a total of 48 features for all the three classes. The normalizing method and method of selecting best 16 invariant features are followed in the same manner as in the selection of 256 features in the HNeT method.

A real-to-complex Fourier Transform method is applied to the pixel magnitude of each SAR image, generating a set of Fourier coefficients to be used as our feature space. Fourier Transform coefficients are measures of periodicity. To reduce the computation times of the Fourier transform, the SAR images are first cropped to N x N (N=64) chips (Fast Discrete Fourier Transform (DFT) is faster if the image size is in power of 2). Before applying the DFT to get the Fourier coefficients, the chip size image is normalized as follows:

$$\mu = \frac{\sum_{x=1}^{N} \sum_{y=1}^{N} f(x,y)}{N * N}, \tag{1}$$

where μ is mean and f(x,y) is image pixel magnitude at location x and y.

$$\sigma = \sqrt{\frac{\sum_{x=1}^{N} \sum_{y=1}^{N} (f(x,y) - \mu)^2}{N * N}}, \tag{2}$$

where σ is the standard deviation of the image.

The mean and standard deviation are calculated as shown in Eq.1 and Eq.2. Then the image is normalized as follows,

$$nf(x,y) = \frac{f(x,y) - \mu}{\sigma}, \qquad 1 \le x, y \le N, \tag{3}$$

where nf(x,y) is the normalized image value. Effectively, the mean is set to zero and the standard deviation set to one by doing the above normalization. The DFT algorithm is applied to the normalized image according to

$$F(u,v) = \frac{1}{N} \sum_{x=1}^{N} \sum_{y=1}^{N} nf(x,y) e^{-2\pi i(xu+yv)/N}, \tag{4}$$

where F(u,v) is the Fourier transform of image. The Fourier coefficients from (4) are separated into 4096 real (RF(u,v)) coefficients and 4096 imaginary (IF(u,v)) coefficients. Then, normalized RF(u,v) and IF(u,v) are calculated separately by replacing f(x,y) in (1) and (2) with RF(u,v) and IF(u,v). Whereby (3) becomes

$$NRF(u,v) = \frac{RF(u,v) - \mu_{real}}{\sigma_{real}} \qquad 1 \le u, v \le N, \tag{5}$$

and

$$NIF(u,v) = \frac{IF(u,v) - \mu_{imag}}{\sigma_{imag}} \qquad 1 \le u, v \le N. \tag{6}$$

NRF(u,v) and NIF(u,v) are the normalized real and imaginary Fourier coefficients. In this way, normalized real and imaginary coefficients are computed for all the training samples. Since there are 8192 (4096 real and 4096 imaginary coefficients) Fourier coefficients, if we feed these coefficients to a NN, the training process will require too many inputs, will impede the generalization capability of the NN, as well as needlessly consuming time and computing resources. Therefore, a selected few (16 for each class) of the coefficients are retained. The 16 features are not randomly selected, but are the most invariant coefficients in that particular class of image compared to the coefficients of other classes. To get these 16 features for each class, all the training samples' normalized coefficients are first linearly mapped to polar angle [5] between 0 and π. The linear mapping of each coefficient is given by

$$\theta_k(u,v) = \text{Rconst}(u,v)\,(NF_k(u,v) - \text{mi}(u,v)), \tag{7}$$

where

$$NF_k(u,v) = \begin{cases} NRF_k(u,v) \\ \text{Or} \\ NIF_k(u,v) \end{cases}, \tag{8}$$

$$\theta_k(u,v) = \begin{cases} R\theta_k(u,v) \\ \text{Or} \\ I\theta_k(u,v) \end{cases}, \tag{9}$$

and

$$Rconst(u,v) = \frac{\pi}{ma(u,v) - mi(u,v)}, \tag{10}$$

with

$$mi(u,v) = \min\left(NF_k(u,v)\right), \tag{11}$$

and

$$ma(u,v) = \max\left(NF_k(u,v)\right), \tag{12}$$

subject to $1 \le u,v \le N$ and $1 \le k \le M$.

$R\theta_k$ (u,v) and $I\theta_k$ (u,v) are the real and imaginary rescaled polar angles and $Rconst(u,v)$ is a ratio of the modified range to the original range of the real/imaginary coefficients at (u, v). mi(u,v) and ma(u,v) are the minimum and the maximum values of the real/imaginary coefficients at (u,v) respectively. M is the number of training samples in the training set.

For a given target class, each training sample will belong to one of two groups, the in-class group or the out-class group. Ideally, the 16 features to be selected need to be invariant for in-class samples and random valued over the out-class group. To measure the invariance of the in-class features, the number of in-class samples should be equal to the number of out-class samples. If the number of samples in each group is not equal, then a random selection of samples from the smaller group is added to the samples of that group, thereby making the number samples in both groups equal. Call this amount L. A measure of invariance for each Fourier coefficient can then be calculated according to

$$RR(u,v) = \left| \sum_{k=1}^{L} s(k) e^{jR\theta_k(u,v)} \right|, \tag{13}$$

and

$$IR(u,v) = \left| \sum_{k=1}^{L} s(k) e^{jI\theta_k(u,v)} \right| \tag{14}$$

where

$$s(k) = \begin{cases} +1 & \mathit{if\ it\ is\ in-class} \\ -1 & \mathit{if\ it\ is\ out-class} \end{cases}, \tag{15}$$

and

$$1 \le u,v \le N \ .$$

The 16 longest independent polar vector lengths, RR(u,v) or IR(u,v) are selected for the feature set. Redundancy in the real to complex Fourier transform means that each coefficient appears as a duplicate pair. Only one of each pair is retained in the feature set for each class. Some of these features may be common to more than one class, while others are not. For the 3 MSTAR vehicle classes, the total number of unique selected features has been determined to be 38, less than the maximum possible total of 48 features for the 3 classes. These 38 features are fed into the neural network after being set between $-\pi/2$ and $+\pi/2$ by subtracting $\pi/2$ from the calculated polar angle (Eq.7).

3. Implementation of MLP Neural Network

Using the vector of extracted features, the classifier must be able to correctly decide whether each image is a known target or an unknown target. The MLP NN is able to make an intelligent decision based on learning the sample data. The MLP NN contains many layers and nodes that are connected via adjustable weights. By manipulating these weights, the NN output decisions can be matched to the desired decisions known for the training set. Input and output layers are in contact with the outside world, while the hidden layers are not available for outside connection, but instead are connected between input and output layer, input and another hidden layer, two hidden layers or between a hidden and the output layer.

For application to the MSTAR problem, an MLP NN was selected with two hidden layers: 13 nodes in the first hidden layer and 11 nodes in the second hidden layer, as shown in fig.1. The number of layers and number of nodes in the hidden layers were decided using empirical testing of the NN and these are not optimized numbers. The number of input nodes depends on the number of features chosen and, as we have chosen 38 features (see section 2.0), 38 nodes should be included in the input layers of the NN. Three ground vehicle classes (BMP-2, BTR-70 and T-72) are considered in this task, therefore three output nodes are included in the output layer.

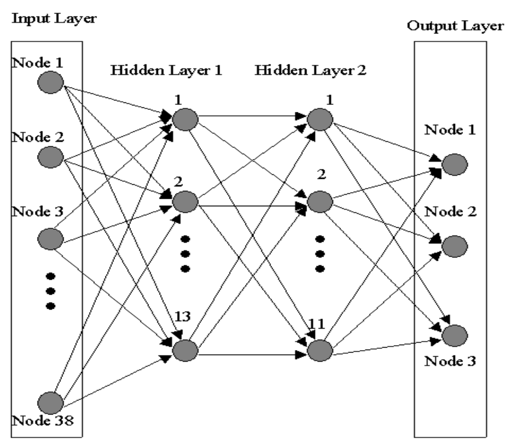

Figure 1. Four layer Multi Layer Perceptron Neural Network

The MLP neural network was trained using a standard back-propagation training method, the delta rule algorithm [20], and it is used to update the weights in this task. Details of the derivation for the learning (training) and testing algorithms are not discussed here, but their implementation is described. The steps of the training algorithm are listed in the table 1. Weights are initialized with small random values and then each training sample is fed into the NN, one by one. For every sample, the output error from the desired result is computed and the partial derivative of the error calculated with respect to each weight. This step is implemented for all the training samples and the partial derivatives are summed up together for each weight. These partial derivatives indicate the direction in which the particular weight has to be varied to minimize the total error. Thus, the weights are updated using the previous weights, partial derivative and preset leaning rate (α) values. The learning rate value determines how far to move the weights in the direction given by the partial derivatives. Through empirical testing, α =0.35 is chosen for this application. If the learning rate is too large in value then the NN will oscillate and will not minimize the error. But, if the learning rate is too small in value, then the learning speed will slow although the NN will eventually converge to a local minimum.

Table 1. MLP training steps using back-propagation training method

Step 1. Initialize weights to small random numbers

Step 2. Bias (b) and sigmoid function's slope (s) set to 0.1 and 0.2, respectively (These values are decided by this author based on his own experience).

Step 3. Input a sample from the training set

Step 4. **Compute output**

$$Net\_inp_j = \sum_{i=1}^{N} \left(x_i w_{ij}\right) + b \quad ,$$

$$y_j = \left(\frac{2}{1 + e^{-1(s \times Net\_inp_j)}} \right) - 1,$$

$$1 \le j \le M \; ,$$

where i is the node of the previous layer, N is the number of nodes in the previous layer, j is the node of the calculating layer, M is the number of nodes in the calculating layer. If calculating layer is first hidden layer then x_i is the i^{th} node input feature, otherwise x_i = y_i (y_i is the output of the i^{th} node of previous layer) and y_j is the output of the j^{th} node of the calculating layer.

Step 5. **Estimate of weight adjustments**

Weights adjustment between last hidden layer and output layer

$$\delta_l = \left(t_l - O_l\right)\left(1 - O_l^2\right)\left(\frac{s}{2}\right),$$

$$\nabla w_{jl} = \nabla w_{jl} + \alpha \times \delta_l \times y_j \, ,$$

where t_l is the desired output value, O_l is the calculated output value, δ_l is the error at output node l, y_j is the calculated output value (at node j) for hidden layers and ∇w_{jl} is the weights adjustment from the hidden layer node j to output layer node 1.

Step 5. **Estimates of weight adjustments (Continued…)**

Weights adjustment between input and hidden layers or between hidden layers

$$\delta_j = \left(1 - y_j^2\right)\left(\frac{s}{2}\right)\left(\sum_{l=1}^{L} \delta_l w_{lj}\right),$$

$$\nabla w_{ij} = \nabla w_{ij} + \alpha \times \delta_j \times y_i ,$$

where δ_l is the error at output node l, δ_j is error at hidden node j, y_i is the calculated i[th] node output at hidden layer (if j is a node of the first hidden layer, then $y_i = x_i$ and x_i is input feature) and ∇w_{ij} is the weights adjustment from node i to node j.

Step 6. **If all the samples are not used for training, then select next sample from the training set and repeat Steps 4 and 5. Otherwise go to Step 7.**

Step 7. **Update of weights**

The weights between the last hidden layer and output layer are determined by

$$w_{jl}(t+1) = w_{jl}(t) + \nabla w_{jl} ,$$

whereas the weights between the input and hidden layers or between adjacent hidden layers use

$$w_{ij}(t+1) = w_{ij}(t) + \nabla w_{ij}$$

Step 8. **Calculate the number of targets correctly classified**

Using the updated weights, calculate the output layer node values as in step3. Then compare the result with the target output.

```
Count=0
Start p=1 and go until s=M, where M = number of training sample
        check=0
        Start j=1 and go until j=N, where N=number of output nodes
                If |t_j^p - O_j^p| > threshold, where threshold set to 0.3
                        check=1
                        get out from the j loop
                End if
        End loop j
        If Count = 0
                Count = count+1
        End if
End loop p
```

If Count=M, then training phase completed and stop the training process, otherwise go back to step 3.

The MLP NN is trained with stopping criterion that the network should recognize the entire training set correctly. However, there is no guarantee in any case that the algorithm will reach the global minimum error. The NN output nodes assigned for each target are as follows, T-72 to node 1, BTR-70 to node 2 and BMP-2 to node 3. For the training phase, output values are limited to –1 to +1 by sigmoid function. For example, for a BTR-70 image to be correctly recognized output nodes 1 and 3 must generate a value below (-1+ threshold) and output node 2 a value above (+1-threshold). In the evaluation experiments, the threshold value is empirically set to 0.3. The time taken for training varies over a large period, normally ranging from 5 to 30 minutes. In this application, if all the samples were not learned within half an hour, the training process should be restarted. Trained weights used for this experiment took 450.937 seconds (Pentium 4 CPU 2.0 GHz computer) to obtain convergence with all the training samples. The memory space needed for storage in the training process is 11,210 Kb (11,199 Kb for training samples binary file, 3 Kb for initialization text file and 8 Kb for weights text file).

Table 2. MLP testing steps

Step 1. **Initialize weights to previously trained weights**

Step 2. **Input a sample from the testing set**

Step 3. **Compute output**

For Hidden layer nodes

$$Net\_inp_j = \sum_{i=1}^{N} \left(x_i w_{ij}\right) + b,$$

$$y_j = \left(\frac{2}{1+e^{-1(s \times Net\_inp_j)}}\right) - 1,$$

$$1 \le j \le M,$$

where i is the node of the previous layer, N is the number of nodes in the previous layer, j is the node of the calculating layer and M is the number of nodes in the calculating layer. If calculating layer is first hidden layer then x_i is the i^{th} node input feature, otherwise $x_i = y_i$ (y_i is the output of the i^{th} node of previous layer) and y_j is the output of the j^{th} node of the calculating layer.

For output layer nodes

$$Net\_inp_j = \sum_{i=1}^{N} \left(y_i w_{ij}\right) + bias,$$

$$O_j = \left(\frac{2}{1+e^{-1(slope \times Net\_inp_j)}}\right),$$

$$1 \le j \le M,$$

where N is the number of nodes in the last hidden layer, M is the number of nodes in the output layer, y_i is the output of the last hidden layer node i and Oj is the output of the j^{th} node of the output layer.

Step 4. **If all the samples are not tested, then select the next sample from the testing set and repeat the Step 3. Otherwise end the testing process.**

The testing algorithm is very simple and the steps are listed in table 2. Each test sample is fed into the input of the trained NN and then the output is computed using the predetermined trained weights. The output values vary from 0 to 2. An output value that is equal to 2 means the test sample is similar to that vehicle type and 0 means it is not similar. It is not practical to have the output values to be exactly 2 or zero, so it is necessary to have some kind of threshold range value to accept the decision. The threshold may be variable, which parameterizes the corresponding Receiver Operation Characteristics (ROC) curve. Conversely, the threshold may be chosen from the ROC curve according to what percentage of detection rate is needed for the application, as described in the Result and Discussion section (section 5.0). If more than one output node have values close to 2 within the threshold range, then the node closest to 2 is selected. If none of the output node values are close to 2 within the threshold range, then the test sample is classified as unknown

4. Data Set

With the increasing need for automated exploitation of SAR images the collection of ground truthed data also increased [4]. The data used for this study is the MSTAR public data set of SAR images collected in spotlight mode at 30 cm resolution [21]. The data was collected in September 1995, November 1996 and May 1997 by the Sandia National Laboratory (SNL) and released to the public by U.S. Air Force Wright Laboratory [21]. The data is divided into target and confuser data sets according to the COMPASE Center evaluation criteria [4]. Then the target set is divided into training set and testing set. During the training stage, the classifier algorithm needs sample images, along with known class type. The training set of this application has three target classes, T-72, BMP-2 and BTR-70. All training images are at 17-degree depression angle and with full aspect of coverage. The target types and the number of samples used for training are listed in the table 3. A single vehicle of each vehicle type is used to train the classifier.

Table 3. Training Set

Targets type and serial number	# of samples	Comments
T-72 (132)	232	All the targets collected at 17 degree depression angle, full aspect coverage and 30 cm resolution
BTR – 70 (c72)	233	
BMP – 2 (9563)	233	
Total	= 698	

The data set (Testing set 1) used to measure the recognition rate is listed in the table 4. All the vehicles used for this test belong to one of the defined class types but may have a different serial number. As well, all test imagery is at a 15-degree depression angle instead of the 17-degree depression angle used for training.

Table 4. Testing set 1 - Testing samples for confusion matrix test

Targets type and serial number	# of samples	Comments
T-72 (812)	195	All the targets collected at 15 degree depression angle, full aspect coverage and 30 cm resolution
T-72 (s7)	191	
T-72 (132)	196	
BTR-70 (c72)	196	
BMP-2 (9563)	195	
BMP-2 (9566)	196	
BMP-2 (c21)	196	
Total	= 1365	

To generate ROC curves, vehicles not belonging to the types used in training are required to measure false alarms. The confuser images are also at the 15-degree depression angle. Table 5 shows the types of confusers and number of samples used to perform the experiment.

Table 5. Testing set 2 - Testing samples for ROC curve test

Targets type and serial number	# of samples	Comments
2S1	274	All the confusers collected at 15 degree depression angle, full aspect coverage and one feet resolution
D7	274	
T62	273	
ZIL-131	274	
BTR-60	195	
ZSU-23/4	273	
Total	= 1563	

Both the training set and testing set samples were extracted from the image chips that were first aligned according to the ground truth heading and the central image block of 64 x 64 pixels extracted. The two hidden layer MLP NN was trained using the training data set according to the procedure outlined in section 3. The results of the evaluation experiments are discussed in the next section.

5. Results and Discussion

The trained NN is applied to testing sets 1 and 2 as described in section 4. To generate the ROC curves, the threshold value is varied from 0 to 2 in increments of 0.01. At each increment, the percentage of detection (P_d) and percentage of the false alarms (P_{fa}) are calculated and then a graph plotted. The graph constitutes a "Receiver Operating Characteristics (ROC) curve" [22] as is shown in the Figure 2.

According to figure 2, for the same P_d, the P_{fa} rate is highest for the BMP-2 while the BTR-70 performance is better compared to other two vehicles. Better classifiers should provide lower P_{fa} rates and higher P_d rates. From figure 2, an optimal threshold value can be found to get the best performance of each classifier node by taking the operating point of the classifier in the ROC curve closest to the point (0,1), i.e., the left upper corner. For this case, the threshold values are 0.03 for T-72, 0.01 for BTR-70 and 0.12 for BMP-2. Using these thresholds, correct classification rates are calculated for testing set 1, as shown in table 6. Asterisks indicate the specific vehicles that appear in both training and testing sets. The results obtained for these vehicles is higher than that of the results obtained with other vehicles of same type, as expected.

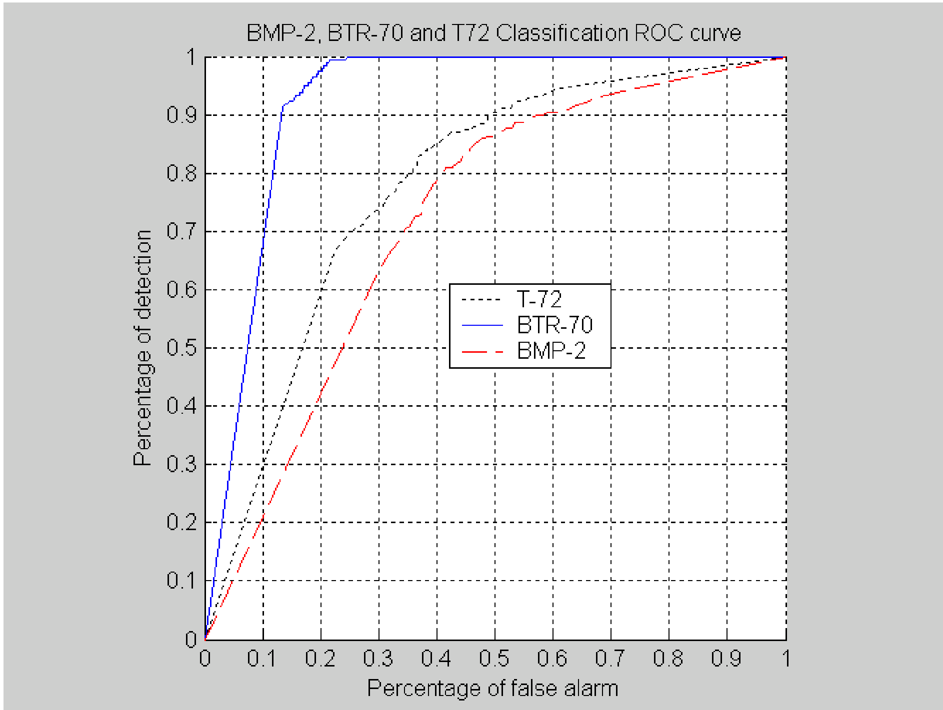

Figure 2. ROC curves of the MLP NN classifier, using the test set against the six confusers.

Table 6. MLP Confusion matrix (At high P_d rate and low P_{fa} rate)

	T-72	BTR-70	BMP-2	Rejected	$P_{cc\|d}$ (%)
T-72(812)	**127**	8	16	44	84.11
T-72(S7)	**112**	11	30	38	73.20
T-72(132)*	**171**	0	6	19	96.61
BTR-70(C72)*	3	**178**	0	15	98.34
BMP-2(9563)*	3	2	**168**	22	97.11
BMP-2(9566)	29	10	**121**	36	75.63
BMP-2(C21)	18	2	**150**	26	88.24
					$P_{cc\|d}$ = 88.15%

Keeping the same thresholds, the misclassification and rejection rates are determined using testing set 2. Misclassification for each confuser vehicle is calculated by dividing the number of vehicle images misclassified by the total number of vehicle images tested. The results as listed in table 7, show the misclassification rate is considerably higher than the rejection rate on some of the vehicles. For instance, the D7 is confused for a BMP-2 and the T-62 is often confused for a T-72.

Table 7. Misclassification rate (%) and confuser rejection rate (%) (At high P_d rate and low P_{fa} rate)

	T-72	BTR-70	BMP-2	Rejected
2S1	12.77	28.47	31.75	27.01
D7	5.47	0.36	85.77	8.39
T62	49.27	6.20	17.15	27.37
ZIL-131	22.63	22.26	28.83	26.28
BTR-60	18.97	23.59	22.05	35.38
ZSU-23/4	42.86	0.00	30.77	26.37

Typically, the evaluation experiment is done with P_d set to 0.9. The thresholds are determined for each class from the ROC curves giving 0.14 for the T-72, 0.14 for the BTR-70 and 0.14 for the BMP-2, with the results is listed in table 8. The over all $P_{cc|d}$ and rejection rates are decreased and the misclassification rate is increased compared to the previous experiment as shown in table 9. Noticeably, none of the ZSU-23/4 targets were misclassified as BTR-70 at previous threshold setting and only 0.37% misclassified at this threshold setting, indicating the features extracted in this case discriminate well between those two classes.

Table 8. MLP Confusion matrix (P_d to 0.9)

	T-72	BTR-70	BMP-2	Rejected	$P_{cc|d}$ (%)
T-72(812)	**141**	13	15	26	83.43
T-72(S7)	**122**	13	30	26	73.94
T-72(132)*	**177**	3	5	11	95.68
BTR-70(C72)*	3	**183**	0	10	98.39
BMP-2(9563)*	5	5	**170**	15	94.44
BMP-2(9566)	31	16	**121**	28	72.02
BMP-2(C21)	23	4	**152**	17	84.92
					$P_{cc|d}$ = 85.20%

Table 9. MLP Misclassification rate (%) and confuser rejection rate (%) (P_d to 0.9)

	T-72	BTR-70	BMP-2	Rejected
2S1	16.42	33.94	31.75	17.88
D7	8.03	0.73	86.86	4.38
T62	54.95	9.16	17.58	18.32
ZIL-131	25.55	27.01	28.83	18.61
BTR-60	24.62	28.72	22.56	24.10
ZSU-23/4	52.75	0.37	31.14	15.75

Next, the experiment is continued without any thresholding on the output nodes. This raises the false alarm rate to 100%. By doing this, each image is forced to be classified as one of the trained classes of vehicles. First, the output is computed for a test image, and then the error is calculated for each output node. After that, the image is classified to the class that contains the minimum error. Therefore testing set 1 is used for this experiment and there is no need to use testing set 2 except to determine which target types each confuser is most similar. The percent of correct classification of declared targets is shown in table 10.

Table 10. MLP confusion matrix – does not reject any vehicle (100% false alarm rate)

	T-72	BTR-70	BMP-2	$P_{cc\|d}$ (%)
T-72 (812)	152	16	27	77.95
T-72 (s7)	135	15	41	70.68
T-72 (132)	185	4	7	94.39
BTR-70 (c72)	3	189	4	96.43
BMP-2 (9563)	6	6	183	93.85
BMP-2 (9566)	41	16	139	70.92
BMP-2 (c21)	27	7	162	82.65
				$P_{cc\|d}$ = 83.88%

Once again, the advantage of using a neural network classifier is that it consumes less memory than many other methods and makes decisions very quickly. The memory space needed for this classifier is as little as 13 KB and the speed of testing one image is 16 ms on a Pentium 4 CPU 2.0 GHz computer using Matlab.

6. Conclusion

Here, a four layer MLP NN is implemented to classify three ground target vehicle types from SAR imagery. The NN is trained using a back-propagation algorithm on imagery with 17-degree depression angle. For testing, data with a 15-degree depression angle is used. Standard evaluation methods for the MSTAR data, using Receiver Operating Characteristic curves and confusion matrices, are used to evaluate the MLP neural network classifier. The MLP NN classifier produces the result very quickly (16 ms per image on a Pentium 4 CPU 2.0 GHz computer using Matlab) and consumes small amount of memory space (13 KB). From the ROC curve, MLP NN classifier performs much better than a random classifier [5]. It is possible to increase the classification performance capacity and decrease false alarm by revisiting signature extraction, MLP NN architecture, choosing data for training set, etc.

References

1 Laskin, O., Ramundo, E., and Okerholm, A. (2002), Automatic Target Recognition in Synthetic Aperture Radar Images. (Online) Carnegie Mellon. http://www.google.com/u/carnegiemellonece?q=Automatic+Target+Recognition&hl=en&lr=&ie=UTF-8&sa=n&oq=Automatic+Target+Recongition (16 Sept. 2002)

2 Schroeder, J.. Automatic Target Detection and Recognition Using Synthetic Aperture Radar Imagery, (Online) National Committee for Radio Science - Australian Academy of Science. http://www.ips.gov.au/IPSHosted/NCRS/wars2002/invited.htm (17 Sept. 2002)

3 Bhanu, B. (1986), Automatic target reconition: State of the art survey, IEEE Trans. Aerospace and Electronic Systems, 22(4), 364-379.

4 Ross, T.D., and Mossing, J.C. (1999). The MSTAR Evaluation Methodology. Algorithms for SAR Imagery VI. 705 - 715, Orlando, USA: SPIE

5 English, R. A. (2001). Automatic Target Recognition Using Hnet, (DREO TM 2001-080). Defence R&D Canada - Ottawa

6 Ross, T.D., Velten, V., Mossing, J.C., Worrell, S., and Bryant, M. (1998). Standard SAR ATR Evaluation Experiments using the MSTAR Public Release Data Set. Algorithms for Synthetic Aperture Radar Imagery V, 566-573. Orlando, USA: SPIE

7 O'Sullivan, J.A., DeVore, M.D., Kedia, V. (1999), Performance Analysis of ATR from SAR Imagery Proceedings of the 33rd Annual Conference on Information Science and Systems, Baltimore, MD, USA.

8 Zhang, G.P. (2000), Neural Networks for classification: A Survey, IEEE Trans. Systems, Man, Cybernetics – c, 30(4), 451-462.

9 Benediktsson, J.A., Swain, P.H., and Ersoy, O.K. (1989), Neural Network Approaches Versus Statistical Methods in Classification of Multisource Remote Sensing Data, Proc. of the IEEE International Geoscience and Remote Sensing Symposium, 489-492. Vancouver: IEEE.

10 Schumann, A. (1997), Neural networks versus statistics: a comparing study of their classification performance on well log data, Proceedings of IAMG'97, part 1, 237-241.

11 Fauzi, A., Hussin, Y. A., and Weir, M. (2001), A Comparison between Neural Networks and Maximum Likelihood Remotely Sensed Data Classifiers to Detect Tropical Rain Logged-Over Forest in Indonesia, Proc. ACRS 2001 - 22nd Asian Conference on Remote Sensing, 616-621. Singapore: CRISP, SISV and AARS.

12 Roth, M.W. (1990), Survey of Neural Network Technology for Automatic Target Recognition, IEEE Trans. Neural Networks, 1(1), 28-43.

13 Anderson, D., McNeill, G. (1992). Artificial Neural Networks Technology. (Rome Laboratory RL/C3C, Griffiss AFB, NY, F30602-89-C-0082). Kaman Sciences Corporation.

14 Champandard A.J. (2001-2002). From Biology to Computing. (Online) Artificial Intelligence Depot. http://neuralnetworks.ai-depot.com/3-Minutes/ (12 Sep. 2002).

15 Nikolaev, N. CIS 311 Neural Networks. (Online) Goldsmiths College, University of London. http://homepages.gold.ac.uk/nikolaev/cis311.htm (13 Sep. 2002).

16 Bertoni, A., Campadelli, P., and Cesa-Bianchi, N. (1997). Multilayer Perceptrons and Learning. (NeuroCOLT Technical Report Series NC-TR-97-009). ESPRIT Working Group in Neural and Computational Learning – Italy

17 Sundararajan, N., and Saratchandran, P. (1998), Parallel Architectures for Artificial Neural Networks, IEEE Computer Society

18 Sarle, W. S. (1994), Neural Networks and Statistical Models, Proceedings of the Nineteenth Annual SAS Users Group International Conference, 1538-1550,NC, USA: SAS Institute Inc.

19 Franco, L., and Cannas, S.A.(200), Generalization and Selection of Examples in Feed Forward Neural Networks, Neural Computation, 12 (10), 2420-2449.

20 Lippmann, R. P. (1987), An introduction to computing with neural networks, IEEE ASSP Mag., 4(2), 4-22.

21 (1998). MSTAR public Targets. (Online) Air Force Research Laboratory (USA). http://www.mbvlab.wpafb.af.mil/public/sdms/datasets/mstar/ (17 Sept. 2002).

22 Biometrics Introduction. (Online) National Institute of Justice. http://www.biometricscatalog.org/Introduction.asp (20 Sept. 2002)

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.) DRDC Ottawa 3701 Carling Avenue Ottawa, ON, K1A 0Z4	2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable) UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

Automatic Target Recognition in SAR Imagery using a MLP Neural Network(U)

4. AUTHORS (Last name, first name, middle initial)

Sandirasegaram, Nicholas, M

5. DATE OF PUBLICATION (month and year of publication of document) November 2002	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.) 15	6b. NO. OF REFS (total cited in document) 22

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

TECHNICAL MEMORANDUM

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

DRDC Ottawa
3701 Carling Avenue
Ottawa, ON, K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant) 3db31	9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.) DRDC Ottawa TM 2002-120	10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

UNLIMITED

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

In this report, a Multi Layer Perceptron (MLP) Neural Network is used for recognizing military ground vehicles imaged by Synthetic Aperture Radar (SAR). In particular, the classifier is applied to SAR images taken from the MSTAR (Moving and Stationary Target Acquisition and Recognition) data set, which has been made available to the public. Signatures are extracted from the imagery using a Fourier Transform method and features are selected to feed the neural network. A 4-layer (including input and output layers) Neural Network with 38 input nodes, 13 first hidden nodes, 11 second hidden nodes and 3 output nodes, is implemented for this task. Standard delta rule back-propagation algorithm has been used to train the neural network. The MLP neural network is evaluated according to the MSTAR standard evaluation criteria. Training of 3 vehicle classes occurs using a set of SAR images at a 17-degree depression angle with 0-360 degree azimuthal angles, while the testing set contains images at a 15-degree depression angle with 0-360 degree azimuthal angles. The testing set contains both target vehicles that belong to the 3 trained classes and confuser vehicles that do not. Results of MLP neural network evaluation are shown using Receiver Operating Characteristic (ROC) curves and Confusion Matrices.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Automatic Target Recognition, ATR, Synthetic Aperture Radar imagery, spot SAR, MSTAR, Mulit-Layer Perceptron, MLP, Neural Network.

Defence R&D Canada

Canada's leader in defence
and national security R&D

R & D pour la défense Canada

Chef de file au Canada en R & D
pour la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca