Journal of Geological Resource and Engineering 4 (2016) 184-193
doi:10.17265/2328-2193/2016.04.004

A Combination Method of Differential Evolution Algorithm and Neural Network for Automatic Identification Oil Spill at Vietnam East Sea

Hang Le Minh and Truong Vu Van

Department of Geodesy and Mapping, Le Quy Don Technical University, Ha Noi, Viet Nam

Abstract: This paper proposes an automatic detection of oil spills on SAR (synthetic aperture radar) images using DE (differential evolution), neutral network and BP (back propagation) algorithm. Here, DE and BP are combined to train a multilayer perceptron (MLP) network for achieving the global extreme with a better convergence speed. The input data of neural networks are the geometrical characteristics of oil spills (e.g. area, perimeter, complexity) and the physical behavior of oil spills (e.g. mean or max backscatter value, standard deviation of the dark formation). The out data are oil spill or look-alike. We experiment ALOS/PALSAR and EnviSAT ASAR on East sea area of Viet Nam. The experimental results show that the combination algorithm converges faster and has significantly better capability of avoiding local optima.

Key words: SAR, oil spill detection, MLP, BP algorithm, DE.

1. Introduction

In recent years, oil spills from unknown sources are often detected at Vietnam East Sea. Oil spills cause pollution to the sea and coastal zone and degrade environment in general. Annually, 48% of the oil pollution in the seas is caused by fuels and 29% by crude oil. Tanker accidents contribute with only 5% of the all pollution entering into the sea. According to the European Space Agency (1998), 45% of the oil pollution comes from operative discharges from ships. From 1986 to 2000, the oil production of Vietnam was enlarged from 40,000 tons to 16,500,000 tons and more 30 accidents causing oil pollution have occurred in the sea. Therefore, there is strong need to develop an effective solution for oil pollution monitoring and surveying which are suitable and useable in condition of Vietnam.

Remote sensing instruments have become one of the main methods in marine oil spill detection. Among which, SAR (synthetic aperture radar) can provide high resolution images with wide area coverage, day and night, and all-weather capabilities. Oil on the sea surface is observed on SAR images as dark areas because the viscoelastic property of oil slicks dampens the short wave field due to both suppression of wave growth and increase of wave dissipation which reduces the radar backscattering from the sea surface [1, 2]. However, not all dark features on the ocean surface are oil spills. Dark formations can be such as oil spills, low wind areas, organic film, wind front areas, areas sheltered by land, rain cells, current shear zones, grease ice, internal waves and upwelling zones [1, 2].

Recent work has demonstrated that ANNs (Artificial Neural Networks) represent an efficient tool for modelling a variety of nonlinear discriminant problems. Kubat et al. [3] developed a neural network for the classification of dark regions detected in a series of nine SAR images that served as a training set of the system. Del Frate et al. [4] also used neural network architecture for semi-automatic detection of oil spills on SAR images using a set of features characterizing a candidate oil spill as input vector. Solberg, A. H. and

Corresponding author: Hang Le Minh, Dr., research fields: photogrammetry and remote sensing, SAR image.

Solberg, R. [5] and Solberg et al. [6] produced a semi-automated classification for oil spill detection, in which the objects with high probability of being oil spills were automatically detected.

In general, most of the studies on oil-spill detection relay on feed-forward ANNs with back-propagation algorithm. However, there are still many issues that have not been clarified such as ANN configuration, initial weight, the problems of under-fitting and over-fitting. This may lead to instability in the classification results. In this research, we will focus on clarifying these issues. In particular, we concentrate on solving ANNs optimization problem to improve accuracy and stability of the identification oil spills.

The paper is organized in five sections. In Section 2 the theoretical foundation is described and in Section 3 the proposed methodology is given. Results and conclusions follow in Sections 4 and 5, respectively.

2. Theoretical Foundation

2.1 ANN (Artificial Neural Network)

The most widely used ANNs for remote sensing applications are MLPs (Multilayer Perceptrons). These are characterized by the feed-forward architecture of an input layer, one or more hidden layers, and an output layer. The architecture of an ANN typically consists of a set of computing units (nodes or processing units) and is divided into several layers (each layer has multiple units), see the example illustrated in Fig. 1. The degree of connection between units is determined by a set of weighted values. The bias parameter is used to increase the resilience of the network to the problem posed. The number of layers and the units in each layer depends on each problem.

The output of an ANN with p input and h hidden nodes is expressed as:

$$y_t = G(\alpha_0 + \sum_{j=1}^{h} \alpha_j F(\alpha\beta_{0j} + \sum_{i=1}^{l} \beta_{ij} y_{t-i})) \quad (1)$$

where, y_{t-i} (i = 1, 2, …, l) are the network inputs; α_{ij}, β_{jk} are the connection weights (i = 1, 2,…, p; j = 1, 2, …, h); α_0, β_{0j} are the bias terms, and F, G are respectively the hidden and output layer activation functions. Normally, the logistic and identity functions are respectively used for F and G, i.e. $F(x) = 1/(1 + \exp(-x))$ and $G(x) = x$. The model, given by the Eq. (1) is commonly referred as a (p, h, 1) ANN model [4].

2.2 Back-Propagation Training

The back-propagation is a supervised training algorithm in which network weights and bias updating is carried out through the minimization of the error function:

$$E = \frac{1}{2}\sum_{t=p+1}^{N}(y_t - \hat{y}_t)^2 \quad (2)$$

where y_t is the network output, calculated by Eq. (1) and $\hat{y}_t$ is the corresponding target. The algorithm starts with an initial vector w_0 of weights and biases which is updated at each step (epoch) i according to the gradient descent rule:

$$\left.\begin{aligned} \Delta w_i &= -\eta \nabla \mathrm{E}(\mathrm{w})|w = w_i + \alpha\Delta w_{i-1} \\ w_i &= w_i + \Delta w_{i-1} \end{aligned}\right\} \quad (3)$$

where, η and α are the learning rate and momentum factor respectively. The training process continues until some predefined minimum error or maximum number of epochs is reached. The obtained final values of w_i are used for all future predictions.

Although it is easy to implement, this algorithm has some disadvantages such as:

First, at areas or some directions that have flat error surfaces, the small gradient values lead to the convergence speed of the algorithm which becomes slowly.

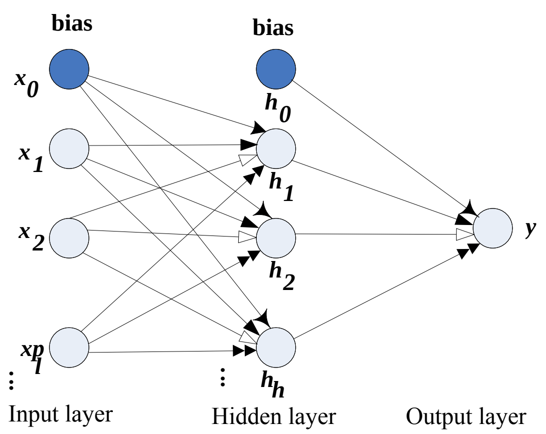


Fig. 1 A feed-forward ANN architecture.

Second, the error surface is not convex; it has different concave parts, it depends on the process of initializing the weights of the network. This results in the algorithm to be clogged at local extremes. However, the BP (back propagation) algorithm is able to converge, so in this study, the authors used BP algorithm to refine the set of weights of the best ANN which were trained by DE (differential evolution) algorithm.

2.3 Differential Evolution

Differential Evolution is a vector population based stochastic optimization method which has been introduced in 1995 by Storn and Price [8]. Like GA (genetic algorithm), this method is able to optimize objective functions which are function of discrete variables. DE is a parallel direct search method which utilizes NP D-dimensional parameter vectors $x_{i,G}$ (i = 1, 2,..., NP) as a population for each generation G. The process can be described as follows:

PROCEDURE Differential Evolution;
BEGIN
i:=0;
create populations X(i);
While not (condition_ stopping) do
Begin
i:= i+1;
Mutation X(i-1)create V;
Crossover X(i-1) based on V create U;
Selection from U and X(i-1) create X(i);
END;

The initial population is chosen randomly and should cover the entire parameter space. For each target vector $x_{i,\,G}$(i = 1, 2,..., NP), a mutant vector is generated, according to:

$$v_{i,\,G+1} = x_{r1,\,G} + \mathrm{F}(x_{r2,\,G} - x_{r3,\,G}) \qquad (4)$$

with random indexes r_1, r_2, $r_3 \in \{1, 2, ..., NP\}$, are three different integers and randomly selected. F is a real and constant factor $\in (0,\ 1)$ which controls the amplification of the differential variation $(x_{r2,j} - x_{r3,j})$ (see Fig. 2).

Due to each new individual is created from three individuals that were randomly chosen over search space thus allowing this new individual to escape the local extreme and reach out to global extremes. On the other hand, according to the Eq. (4) new individual is created in a way that if three individuals in i^{th} generation are close to global extremes, the new individual at $(i + 1)^{th}$ generation will rapidly be created close with these extreme positions. This allows DE convergence faster than other evolutionary algorithms.

2.4 Feature Extraction

Features are very important for the classification because they are used as inputs to the classifier. Therefore, the combination of features which discriminate between the oil spill and the look-alikes plays an important role for the classifier and for the method's accuracy. The features which are usually used for oil spill detection can be generally grouped in three major categories [6, 7]. Features referring to the geometrical characteristics of oil spills (e.g. area, perimeter, complexity), features capturing the physical behavior of oil spills (e.g. mean or max backscatter value, standard deviation of the dark formation or a bigger surrounding area) and features referring to the oil spill context in the image (e.g. number of other dark formations in the image, presence of ships).

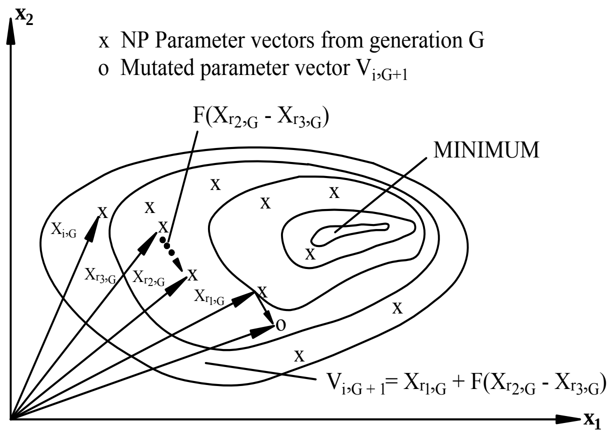


Fig. 2 An example of a two-dimensional cost function showing its contour lines and the process for generating $v_{i,\,G+1}$.

Due to oil stains which were illegal discharged from ships generally have the shape of line prolonged, along the movement of the ship. This is the main characteristic for distinguishing between oil stains and noises on the image. Hence the shape indexes of the oil are important elements to enhance the ability to automatically identify and classify oil stains from the SAR image data. Therefore in this research, we are more interested in the geometrical characteristics of oil spills.

3. Methodology

3.1 Setting the Parameters for the Network ANN

MLP neural networks with error back propagation algorithm are applied more in reality. However, to get good results with ANNs, we need to consider a number of issues such as: the initial weights, momentum constant and network size.

+ *Initializing the weights:* According to research by Wessels and Barnard in 1992 [11], the initializing the weights W_{ij} should be in the range of $[{-3}/{\sqrt{ki}}, {3}/{\sqrt{ki}}]$, k_i is the number of linking between j^{th} neuron and i^{th} neuron.

+ *The momentum constant:* The learning speed of BP algorithm can be very slow if the learning constant is small, but if this value is large, it will cause major fluctuations in the process of finding the minimum value.

+ *Network size:* The studies that are based on Kolmogorov's theorem pointed out that all continuous mappings from $[0.1]^p$ to $[0.1]^n$ can be approximated by a 3-layer perceptron network with p neurons in input layer, n neurons in output layer and $(2p + 1)$ neurons in hidden layers. However, the studies did not point out exactly the number of neurons in the network. Some other studies suggest that the number of neurons in the hidden layer should be smaller $(2p + 1)$. Also, the size of the training data (N), according to Vapnik and Chervonenkis $N = 10 * N_w$, (N_w is the number of network weights). According to some studies, the number of data samples have to satisfy: $N = 4 \times (p + 1) \times L$; with L is the number of hidden layer neurons, p is the number of input layer neurons.

In this study, to avoid over fitting, under fitting and local extremes problem we have selected the network parameters according to the recommendations above.

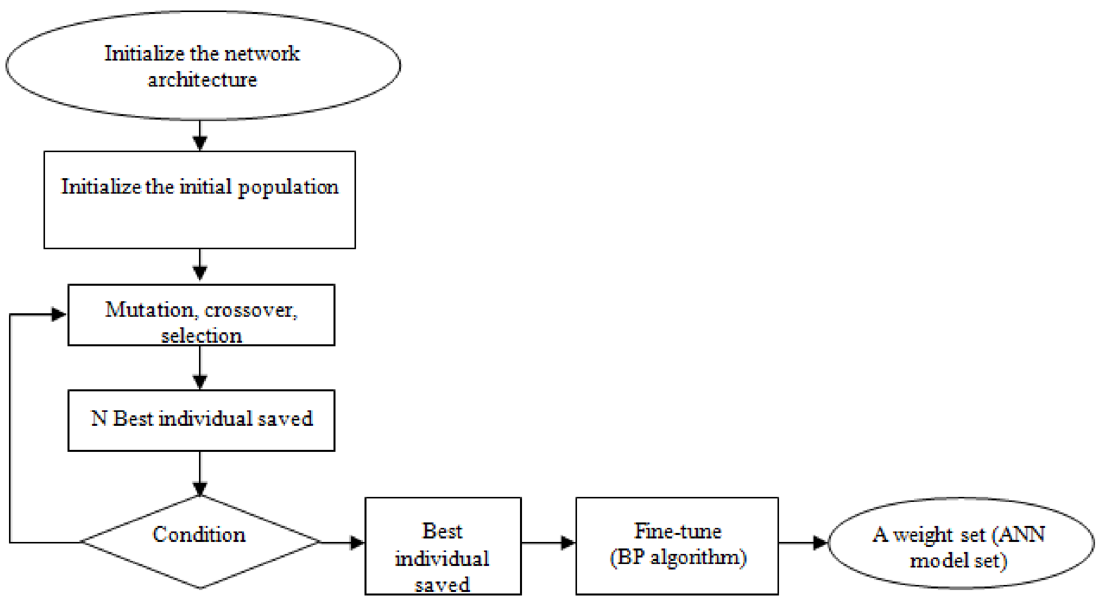


Fig. 3 The block diagram of neural network algorithm combined with DE algorithm.

3.2 Combining DE and BP Algorithm to Optimize ANNs (DE_BP)

3.2.1 General Idea

As indicated above, when using ANNs with BP algorithm, the initial values of the weight are chosen randomly, thus the networks often fall into the local minimum values. However, BP algorithm always ensures the convergence. Whereas, DE algorithm can find out the region which contains the global extreme, but it does not guarantee the convergence. Therefore, the authors propose a combination algorithm DE_BP to create an optimal ANNs. In which, the authors use the DE algorithm to find out the region containing the global extreme, after that we use BP algorithm to refine the set of weights in order to ensure that they reach to the global extreme.

Our proposal works as follows (see Fig. 3): an initial population of N individuals in which each individual is a set of weights in the ANN. DE algorithm evolves the initial population with the selection, crossover and mutation operations. During the evolutionary process, the best individual is maintained. From this individual, the weight of ANN will have an initial value instead of random value.

3.2.2 Verify the Proposed Method

To examine the possibility of converging to global extremes of the proposed method, we conducted experiments on popular XOR problem with 4 samples learning as Table 1:

Table 1 XOR problem.

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	0

Now we will take two trials with two different values of error thresholds: 0.05 and 0.0005. The value 0.05 may be local extreme values or located in region that contain global extreme. The value 0.0005 can be the global extreme value (it means that this network converged). We will compare three methods (BP, DE and DE_BP). Parameters for the whole three methods as follows:

+ A hidden layer with 2 neurons;
+ Learning constant: 0.3;
+ Population size: 100;
+ Hybrid probability: 0.5;
+ Probability of mutations: 0.9;
+ The maximum number of iterations: 10,000.

(*) The experimental results with the threshold error of 0.05 are showed in Table 2.

Table 2 lists the iterations with each algorithm in 100 times of different trials; sign "-" shows that this network is not converged (the loop are more 10,000 times). The table indicates that, DE has the ability to achieve converged requirements (error threshold < = 0.05) or find the region containing extreme global easier than BP algorithm.

(*) The experimental results with the threshold error of 0.0005 are showed in Table 3.

Table 3 shows that DE can reach the region containing the global minimum (corresponding the error value: 0.05) easier than BP (75 compared to 58 times). However, to achieve global minimum (corresponding the error value: 0.0005), the DE is very poor (only 2 times). Whereas most of cases, BP can still get to the region containing the global minimum (65 times).

In particular, the proposed algorithm (DE_BP) totally converged (100/100) with the smaller loop. This indicates that, this algorithm has better and more stable results than BP and DE only.

Table 2 Experimental results comparing DE and BP with the error threshold 0.05.

No	DE	BP	No	DE	BP	No	DE	BP	No	DE	BP	No	DE	BP
1	503	-	21	135	-	41	268	-	61	368	501	81	378	1676
2	-	339	22	173	388	42	181	-	62	356	658	82	-	4782
3	277	454	23	-	-	43	-	713	63	138	-	83	239	400
4	-	274	24	181	311	44	-	306	64	220	291	84	227	442
5	268	-	25	298	-	45	193	326	65	262	459	85	290	360
6	-	-	26	258	-	46	178	291	66	391	491	86	89	-
7	216	-	27	-	658	47	350	-	67	-	371	87	-	255
8	468	301	28	229	-	48	371	-	68	396	-	88	346	389
9	292	-	29	299	429	49	144	545	69	558	699	89	276	-
10	-	-	30	-	-	50	114	671	70	203	-	90	323	-
11	210	-	31	150	307	51	83	307	71	-	-	91	119	-
12	-	791	32	295	354	52	130	422	72	-	524	92	168	-
13	101	392	33	168	-	53	-	412	73	256	-	93	329	-
14	223	-	34	168	-	54	228	-	74	185	658	94	212	-
15	407	614	35	229	364	55	263	537	75	-	-	95	146	-
16	-	-	36	-	-	56	239	337	76	287	366	96	-	1035
17	-	-	37	236	388	57	-	2960	77	256	226	97	-	-
18	221	749	38	-	-	58	169	-	78	722	220	98	245	626
19	293	416	39	254	311	59	317	387	79	181	-	99	361	509
20	720	-	40	405	-	60	316	-	80	138	422	100	-	448

	BP	DE
Number of times network converged	58	75
The average number of iterations	599.6207	119.0943

Table 3 Experimental results comparing DE and BP with the error threshold 0.0005.

	BP	DE	DE_BP
Number of times network converged	65	2	100
The average number of iterations	3,332.797	2,987.653	2,141.44

3.3 Building Algorithm Automatically Detect Oil Slick on SAR Image

Based on the results of research achievement, the authors propose a method for identification and classification of oil spill at sea by SAR images with two different processing divided by the characteristics of oil slick on sea surface. The process is shown in Fig. 4 which applies to the new oil slick. The oil slick on SAR image is contrast with the surrounding surface.

In general, method identification and classification of oil spill at sea by SAR image is a collection of essential steps processing including: (1) preprocessing image; (2) detecting dark spots; (3) identification and classification of oil spill and look-alike in which the detection dark spots on SAR images play an important role. The detection dark spots on SAR image are carried out automatic threshold algorithm or region growing algorithm which improve the automated identification and classification of oil spill at sea by SAR images.

However, almost dark spots on the image are detected by automatic threshold algorithm by Huang so the image is need to treat by pre-processing steps, especially adjust near-far range effect. The near-far range effect creates the difference gray values of oil spill near range and far range at the same image. It makes difficult to use automatic threshold algorithm to detect dark spot on SAR image. We need to use region growing algorithm to detect dark spots on SAR image in the case of oil spill weathered by time and have many gray levels in oil spill.

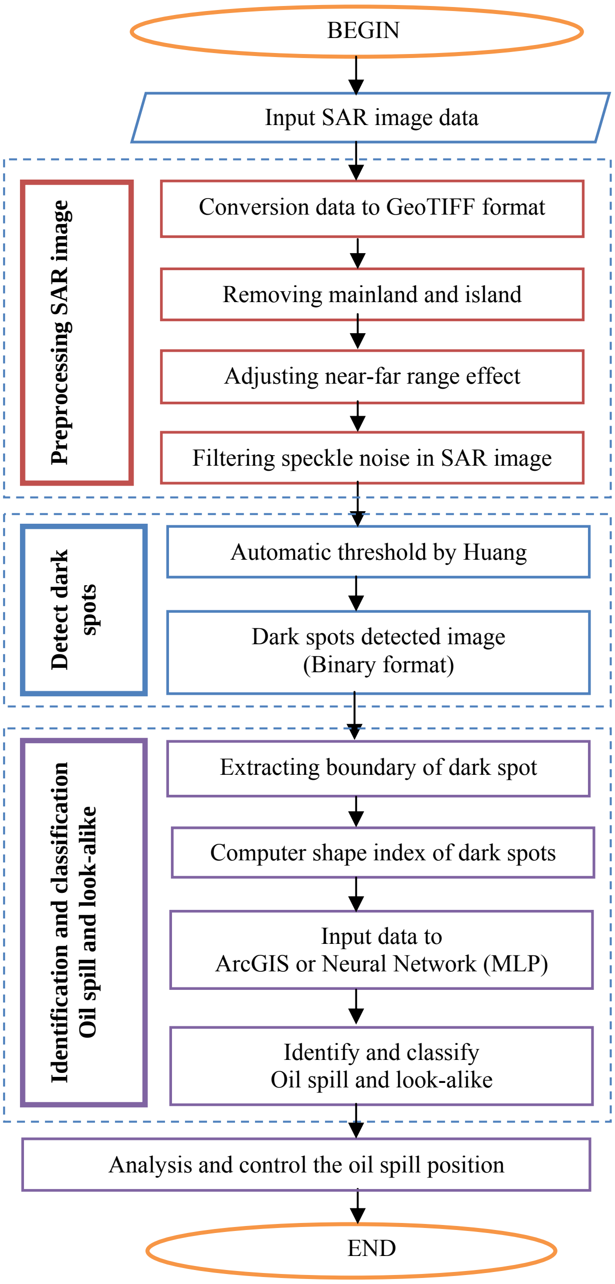


Fig. 4 The method automatic identification and classification oil spill at sea by SAR image.

4. Case Studies

4.1 Experimental Data

The data used in this research are Envisat ASAR, ALOS PALSAR ScanSAR in the South China Sea taken in 2007, 2008, 2009. The total SAR test image data are 16 photos with 120 oil spill and look-alike were discovered on images. The area studies are shown in Fig. 5. The number of traces of oil was detected mainly in the area of south-central coast.

The number of samples is 112, including 72 oil spills and 40 look-alikes. In which we use 82 samples to train network, and 30 standalone samples to test. The network configuration was selected as 4:4:1 with 4 input parameters are the geometry indexes of oil, one hidden layer with 4 hidden neurons and 1 output with value of 1 means oil and 0 is look-alike.

4.2 Experimental Results

Through experiment, classification results are shown in Table 4 and Table 5.

Table 5 shows that the reliability of the classification with neural network model depends heavily on two indexes: the shape Sf and complexity PT. Table 5 also shows clearly that if stains have Sf index large and small PT index will have a high probability of oil. Results for oil No. 3 and No. 9 are 100%. For spots that do not have characteristic shape, such as the No. 1 spot have possibility not high, the possibility of oil and look-alike are 0.33 and 0.55 respectively.

In the results also have some mistakes, such as No. 5. Trace No. 5 was discovered on images as a look-alike. However, due to that it has large size with an area of 272,975 km^2, leading to the result using neural networks is the oil with the possibility is 0.8280.

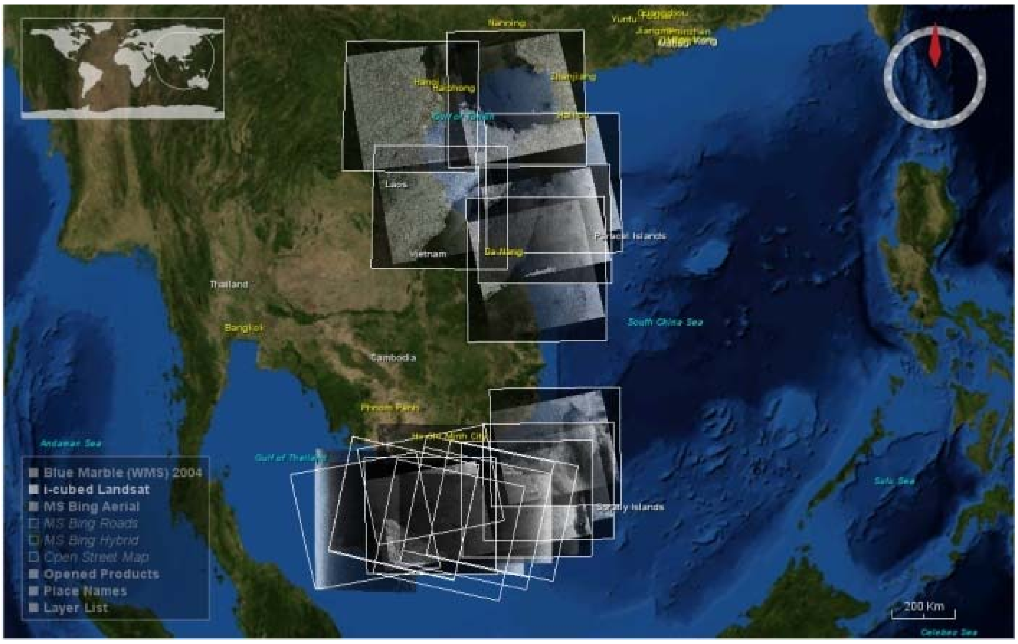
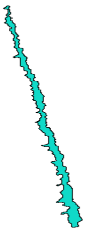


Fig. 5 SAR image data and the area studies.

Table 4 Classification results with MLP neural networks 4:4:1.

No	Oil	P (km)	A (km^2)	Sf	PT	Ability: Oil	Ability: Look-alike	Classify directly on images	Satellite
1		120.281	67.2	1.327	3.668	0.3360	0.5495	Look-alike	Alos palsar
2		2.226	0.158	5.526	1.402	0.8738	0.1419	Oil	Alos palsar
3		55.609	10.368	33.53	4.318	1.0000	0.0000	Oil	Envisat Asar
4		8.76	1.063	7.961	2.125	0.9977	0.0038	Oil	Envisat Asar
5		278.675	272.975	8.467	4.217	0.8280	0.0560	Look-alike	Envisat Asar

(Table 4 continued)

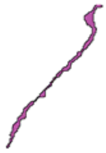
No	Oil	P (km)	A (km^2)	Sf	PT	Ability		Classify directly on images	Satellite
6		12.147	2.828	6.786	1.806	0.9628	0.0366	Oil	Envisat Asar
7		37.421	19.629	1.859	2.112	0.0266	0.9548	Look-alike	Envisat Asar
8		14.424	6.597	2.219	1.404	0.0192	0.9725	Look-alike	Envisat Asar
9		25.87	3.748	10.423	3.341	1.0000	0.0000	Oil	Envisat Asar
10		28.356	14.76	2.376	1.845	0.0084	0.9963	Look-alike	Alos palsar

Table 5 Classification results for the two layer of oil and look-alike.

	Oil	Look-alike	
Oil	71	1	72
Look-alike	1	39	40
	72	40	112
K = 96.11%			

5. Conclusions

This study has demonstrated a combination method of DE algorithm and ANNs using error back propagation algorithm for oil spill automatic detection from SAR image data. Proposed algorithm has overcome the disadvantages and taken advantages of BP and DE algorithm. Experimental results on both of XOR problem and automatic classification oil stain problem demonstrated that the proposed method is capable of extreme converged to a globally with high stability.

References

[1] Alpers, W., Wismann, V., Theis, R., Huhnerfuss, H., Bartsch, N., Moreira, J., and Lyden, J. 1991. "The Damping of Ocean Surface Waves by Monomolecular Sea Slicks Measured by Airborne Multi-frequency Radars during the SAXON-FPN Experiment." In *Proceedings of the International Geoscience and Remote Sensing Symposium (IGARSS91),* 1987-90.

[2] Brekke, C., and Solberg, H. A. 2005. "Oil Spill Detection by Satellite Remote Sensing." *Remote Sens. Environ* 95: 1-13.

[3] Kubat, M., Holte, R. C., and Matwin, S. 1998. "Machine Learning for the Detection of Oil Spills in Satellite Radar Images." *Machine Learning* 30: 195-215.

[4] Del Frate, F., Petrocchi, A., Lichtenegger, J., and Calabresi, G. 2000. "Neural Networks for Oil Spill Detection Using ERS-SAR Data." In *Proceedings of the IEEE Transactions on Geoscience and Remote Sensing,* 2282-7.

[5] Solberg, A. H., and Solberg, R. 1996. "A Large-Scale Evaluation of Features for Automatic Detection of Oil Spills in RS SAR Images." In *Proceedings of the IEEE Symposium on Geoscience and Remote Sensing (IGARSS),* 1484-6.

[6] Solberg, A. H., Storvik, G., Solberg, R., and Volden, E. 1999. "Automatic Detection of Oil Spills in ERS SAR Images." In *Proceedings of the IEEE Transactions on Geoscience and Remote Sensing,* 1916-24.

[7] Zhang, G. P., Patuwo, E. P., and Hu, M. Y. 1998. "Forecasting with Artificial Neural Networks: The State of the Art." *International Journal of Forecasting* 14: 35-62.

[8] Storn, R., and Price, K. 1997. "Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces." *Journal of Global Optimization, Kluwer Academic Publisher* 11: 341-59.

[9] Stathakis, D., Topouzelis, K., and Karathanassi, V. 2006.

"Large-Scale Feature Selection Using Evolved Neural Networks." In *Proceedings of the SPIE, Image and Signal Processing for Remote Sensing XII,* 6365.

[10] Lawrence, S., and Giles, C. L. 2000. "Overfitting and Neural Networks: Conjugate Gradient and Backpropagation." In *Proceedings of the International Joint Conference on Neural Networks,* 114-9.

[11] Wessels, E. B. 1992. "Avoiding False Local Minima by Proper Initialization of Connections." *IEEE Trans. Neural Networks* 3 (6): 899-905.