Machine Learning Algorithms for Multiple Autonomous Unmanned Vehicle Operations

A Fast Detection Algorithm

Hang Shao and Nathalie Japkowicz
Nathalie Japkowicz Consulting Services

Prepared By:
Nathalie Japkowicz Consulting Services
5714 Queen Mary Rd
Hampstead, Quebec
H3X 1X6
Contract Project Manager: Nathalie Japkowicz
PWGSC Contract Number: W7714-115078/001/SV
CSA: F.-A. Bourque, Defence Scientist, 613-992-3206

The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of Defence R&D Canada.

DRDC CORA CR 2013–059
April 2013

Defence R&D Canada
Centre for Operational Research and Analysis

Maritime Operational Research Team

National Defence   Défense nationale

Canada

Machine Learning Algorithms for Multiple Autonomous Unmanned Vehicle Operations

A Fast Detection Algorithm

Hang Shao and Nathalie Japkowicz
Nathalie Japkowicz Consulting Services

Prepared By:
Nathalie Japkowicz Consulting Services
5714 Queen Mary Rd
Hampstead, Quebec
H3X 1X6
Contract Project Manager: Nathalie Japkowicz
PWGSC Contract Number: W7714-115078/001/SV
CSA: F.-A. Bourque, Defence Scientist, 613-992-3206

The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of Defence R&D Canada.

Defence R&D Canada – CORA

Contract Report
DRDC CORA CR 2013-059
April 2013

Principal Author

Original signed by Hang Shao

Hang Shao

Research Scientist

Approved by

Original signed by Dr. R. E. Mitchell

Dr. R. E. Mitchell

Head, Maritime and Air Systems Operational Research

Approved for release by

Original signed by Paul Comeau

Paul Comeau

Chief Scientist

Defence R&D Canada – Centre for Operational Research and Analysis (CORA)

© Her Majesty the Queen in Right of Canada, as represented by the Minister of National Defence, 2013

© Sa Majesté la Reine (en droit du Canada), telle que représentée par le ministre de la Défense nationale, 2013

Abstract

Autonomous Underwater Vehicles (AUVs) are planned to conduct Mine Countermeasure missions in the future. With the help of high resolution imagery produced by the sonar systems mounted on AUVs, mines and other objects of interest can be detected. In this work, the existing approaches for Mine-Like Object (MLO) detection are first reviewed, then, considering the limitation of the exiting works, a novel machine learning method is designed for MLO detection. The experimental result on real side scan sonar images show that the new learning method can provide reliable and fast MLOs detection.

Résumé

L’utilisation de véhicules sous-marins autonomes (VSA) est prévue pour de futures missions de lutte contre les mines. En effet, des mines et d’autres objets présentant un intérêt peuvent être détectés à l’aide des images à haute résolution produites par les systèmes de sonar installés sur des VSA. Dans le cadre des présents travaux, les approches existantes en matière de détection d’objets ressemblant à une mine sont d’abord examinées, puis, en tenant compte des limites des travaux existants, une nouvelle méthode d’apprentissage automatique est conçue pour la détection de tels objets. Le résultat expérimental sur de vraies images de sonar latéral démontre que la nouvelle méthode d’apprentissage peut permettre une détection fiable et rapide des objets ressemblant à une mine.

This page intentionally left blank.

Executive summary

Machine Learning Algorithms for Multiple Autonomous Unmanned Vehicle Operations: A Fast Detection Algorithm

H. Shao; N. Japkowicz; DRDC CORA CR 2013-059; Defence R&D Canada – CORA; April 2013.

Introduction: Autonomous Underwater Vehicles (AUVs) are powerful tools that perform undersea tasks, for both commercial and military purposes. Requiring no operators, they are ideally suited to perform dangerous tasks remotely. Mine countermeasure operations is one such mission. Presently, AUVs are used to collect sonar images of the sea bottom, which are then processed on the mother ship. Processing includes two stages: detection of Mine-Like Objects (MLO) (Stage I) and classification of MLOs into several categories (Stage II). Usually, the MLOs are expected to be detected in a single pass in Stage I and the classification task in Stage II. One of the major challenges with the first stage is to design fast algorithms that allow on-board processing of images. This work focuses on MLOs detection (Stage I). The aim of this contractor report is to introduce a fast machine learning algorithm for the MLOs detection.

Results: The novel learning algorithm developed in this work is able to support fast and accurate MLOs detection. With the proposed algorithm, the efficiency of mine countermeasure missions can be improved. More importantly, the risks to humans can be reduced. The designed algorithms will improve the effectiveness of AUV search and lay a solid foundation for the whole mine countermeasure mission.

Sommaire

Machine Learning Algorithms for Multiple Autonomous Unmanned Vehicle Operations: A Fast Detection Algorithm

H. Shao; N. Japkowicz ; DRDC CORA CR 2013-059 ; R & D pour la défense Canada – CARO; avril 2013.

Introduction: Les VSA sont des outils puissants qui servent à accomplir des tâches sous-marines, à des fins commerciales et militaires. Ils ne nécessitent aucun opérateur et conviennent parfaitement à la réalisation de missions dangereuses à distance, parmi lesquelles on compte les opérations de lutte contre les mines. À l'heure actuelle, les VSA servent à recueillir des images de sonar du fond de la mer, qui sont ensuite traitées sur le navire-mère. Le traitement comprend deux étapes, soit la détection d'objets ressemblant à une mine (étape I) et la classification de tels objets en diverses catégories (étape II). Habituellement, on s'attend à ce que ce type d'objet soit détecté en un seul passage à l'étape I et soit classifié à l'étape II. L'un des principaux défis posés par la première étape est la conception d'algorithmes rapides qui permettent un traitement des images à bord du navire. Les présents travaux portent principalement sur la détection des objets ressemblant à une mine (étape I). Le présent rapport d'entrepreneur porte sur un algorithme d'apprentissage automatique visant la détection d'objets ressemblant à une mine.

Résultats: Le nouvel algorithme d'apprentissage conçu dans le cadre des travaux en question peut appuyer une détection rapide et précise d'objets ressemblant à une mine. Grâce à l'algorithme proposé, l'efficacité des missions de lutte contre les mines peut être améliorée, mais surtout, les risques pour les humains peuvent être réduits. Les algorithmes conçus permettront d'améliorer l'efficacité de la recherche effectuée au moyen de VSA et d'établir des bases solides pour les missions de lutte contre les mines.

Table of contents

List of figures

List of tables

This page intentionally left blank.

1 Introduction

The first two steps of any Mine CounterMeasure (MCM) mission are to detect and classify targets in specified areas of interest in order to reduce the risk for potential ships passing through a region. In the future, Autonomous Underwater Vehicles (AUVs) are envisioned to perform these tasks, thus, keeping operators away from exposure to potential dangers.

The AUVs used in MCM are able to image and survey the seabed, with onboard high-resolution side-looking systems, for instance, the Side Scan Sonars (SSS) [1] or Synthetic Aperture Sonars (SAS) [2]. The underwater conditions can be studied by analyzing the images produced by the sonar systems. In most cases, sonar images produced by SAS will have higher resolution and better quality than those produced by SSS.

The first two steps involved in MCM are Mine-Like Objects (MLOs) detection and classification. In the detection stage, the results are not necessarily real mines. Given the large seabed area that has to be surveyed by the AUVs, usually MLOs have to be detected by a single pass. In the classification stage, the MLOs detected may receive further examination and discrimination. This research will focus on the MLOs detection only.

Currently, the sonar images produced by AUVs are processed by the operators on the mother ship. The introduction of machine learning methods will increase the intelligence and efficiency of the AUVs and free the operators from the tedious and time consuming decision making tasks. Thus, it is of great practical interest to design and apply proper machine learning algorithms to MLOs detection task.

As a military task, there normally exists a time limit for the MLOs detection. As mentioned before, considering the large sea area that needed to be covered as well as the limited computational capability of the CPUs equipped on the AUVs, the speed of the detection algorithm becomes crucial. The applied machine learning method to be developed must then be able to support fast MLOs detection.

2 Existing Work

Many previous works have looked into the MLOs detection task. In some of the works, the detection task is conducted by directly applying certain threshold function(s) on the properties of the sonar images to discriminate between the MLOs and non-MLOs. There is no intelligence or learning process inside such approaches, and in fact, they fall into signal processing techniques [3,4].

An example of such method is the fast detection algorithm recently proposed by David P. William. et. al. [5]. The algorithm can be divided into a three-stage cascade, which are shadow detection, ripple detection and echo/side echo detection. This method is built on the assumption that the shadow regions will have lower greyscale values (darker), while the object regions will have larger greyscale values (brighter), than the surrounding background.

This fast detection algorithm is built on high resolution SAS images where shadows and ripples are detectable. However, for the low quality SSS data, the assumptions behind the fast detection algorithm may not perfectly hold anymore. The shadow detection might be infeasible since it can mix with the background.

In Tucker. et. al's work [6], a method called Canonical Coordinate Analysis (CCA) is used to process the data, then the following detection of Region of Interests (ROIs) is performed by BP (Backpropagation) neural networks. In F. Langner, et. al's work [7], the objects are analyzed by looking at certain contour-based properties after identification of the ROIs. Then the Probabilistic Neural Network (PNN) is applied to find the target of interests. However, usually PNN has a large space complexity that requires large memory and the classification process of PNN can be very slow, which would limit its application to fast detection systems.

3 Proposed Approach

In this research, a novel fast Multi-Layer Perceptron (MLP) based learning algorithm is designed for the MLOs detection. The proposed approach views MLP learning from a perspective that is much different than the traditional one, resulting in a learning model that has its own advantages over other related learning methods for the MLOs detection application.

3.1 Overview of Multi-Layer Perceptron Artificial Neural Network

Multi-Layer Perceptron Artificial Neural Network (ANN) is a family of learning models inspired by the real neural systems in the human brain. Similarly to biological neural networks, ANN consists of many neurons. In some cases, the artificial neurons are grouped into several layers that are interconnected. Such a network is called the MLP. The neurons inside the network will be fired if certain conditions are satisfied. Many real life problems have been well solved by this approach.

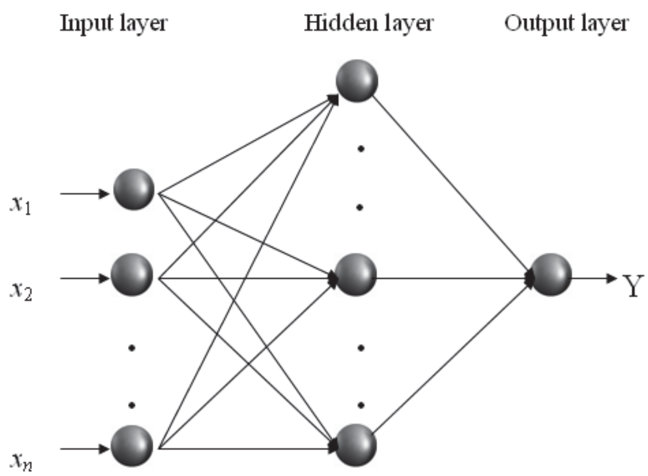

Figure 1: Example of the MLP

Figure 1 shows an example of MLP with one hidden layer, which is also called a Single hidden Layer Feedforward Network (SLFN). In theory, this kind of network is able to perform universal approximation [8], which means that it has the ability to approximate any continuous function within any given error. Therefore, any continuous classification boundary can also be modeled by this approach. In this example, the output layer is composed of only one neuron.

The neurons in the human brain either fire or do not fire. The two states can be represented by a step function, which is

$$g(x) = \begin{cases} 1, x > 0 \\ 0, \textit{otherwise} \end{cases} \tag{1}$$

Unfortunately, the step function is not smooth or differentiable at the origin. Instead, the S shaped sigmoid is usually used as the activation function. Given the input vector $\boldsymbol{x}$ and the input connections (weights) $\boldsymbol{a}$, the sigmoid function can be written as

$$g(\boldsymbol{a}, \boldsymbol{x}) = \frac{1}{1 + \exp[-(\boldsymbol{a}^{\mathrm{T}} \boldsymbol{x} + b)]} \tag{2}$$

In many cases, we may not require the sigmoid function to always pass through the origin, so a bias term b can be introduced to enable possible shift.

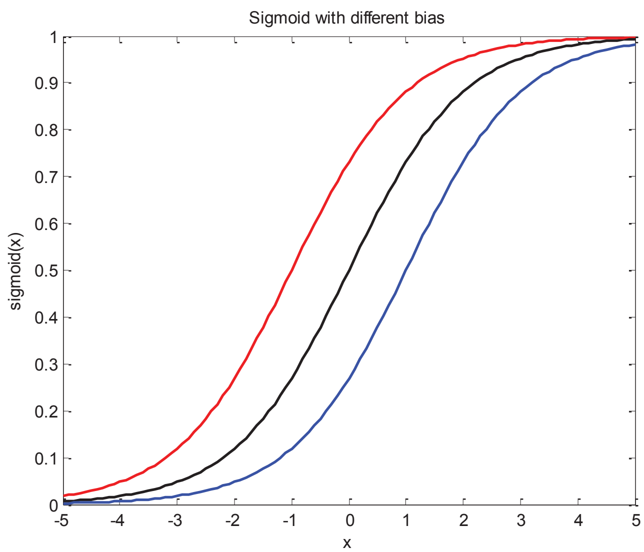

Figure 2: Sigmoid with different bias b

3.2 Fast MLP based Learning Model

Traditionally, the MLP is trained to approximate the given training data according to gradient or gradient related information [9]. Such methods focus on how to reach the (local) minimum of the error surface. Much time is consumed to force the network approximating the training data by tediously adjusting weights, while ignoring the property of the weights during this slow learning process. Such an approach is very likely to overfit the data, which means the network tends to memorize the data rather than learning knowledge from the data. Generally there are two problems the traditional methods suffer from, overfitting and slow learning speed.

Based on the knowledge learned from the training dataset, we are more interested in the network's ability to predict on other general, new and unseen examples, rather than its ability to memorize (overfit) those it has seen in the past. Such generalization ability is closely related to the norm of the weights.

Considering the sigmoid neurons and a linear output layer, the nonlinearity of the network comes from the hidden layer. For the same training data, if the norm of the input weights is small, the sigmoid will work in the nearly linear region, resulting in an output function close to linear in the input space. On the contrary, if the norm of the input weights is large, the output function will gain more complexity and non-linearity. If the weights keep growing, the sigmoid function will be saturated, and it will produce an output either too close to 0 or 1. In such a case, any variation or noise in the input data will be magnified by the weights and lead to a great variation in the network output. In this case, any training error will be heavily magnified, and such networks are very likely to memorize. Unfortunately traditional methods focusing on minimizing training error are very likely to end up with such a network.

The solution of the MLP is, in nature, a set of weights. As we have seen, the norm of input weights is important to the performance of the MLP. Therefore, in the proposed approach, to deal with the overfitting problem, we will not only focus on the training error, but also on the norm of the network weights.

To improve the learning speed, we propose to use the feature mapping idea from kernel learning [10] to construct SLFN. In this way, the hidden layer acts as feature projector and time is saved from optimizing the hidden layer parameters (input weights). In kernel machines, the original non-linear problem can be transferred to a linear problem into another higher dimensional feature space, which becomes much easier to solve. However, different from kernel machines that resort to implicit feature projection where the feature space is invisible. The proposed approach is able to explicitly map the data from the input space to another visible feature space so that the coefficients related to the decision boundary can be directly and explicitly solved.

The proposed approach is a maximum margin classifier in the feature space, which means that it will result in a classifier that maximizes the separating space between the positive data points and negative data points. The margin is shown by the space between the blue line and red line in Figure 2. The positive points are assumed to lie above the blue line while the negative ones are supposed to lie below the red line. However, most datasets in real applications have a certain amount of noise, so sometimes we may allow some data points to fall into the margin or even on the wrong side of the margin.

Given N training instances ($\boldsymbol{x}_i$, t_i), $\boldsymbol{x}_i \in \mathbf{R}^M$, $t_i \in (-1, 1)$, (i=1,2,…,N), where $\boldsymbol{x}_i$ are the attributes, t_i is the class label and M is the number of features, when using a linear output neuron, the output y of a SLFN with L hidden neurons can be written as

$$y_j = \sum_{i=1}^{L} w_i g(\boldsymbol{a}_i, \boldsymbol{x}_j),\ j = 1, 2, ..., N \tag{3}$$

where $g(\boldsymbol{a}, \boldsymbol{x})$ is the hidden layer activation function, which is set to be sigmoid in this study, $\boldsymbol{a}_i$ are the input weights (the hidden bias can be included in the input weights) and w_i are the output weights.

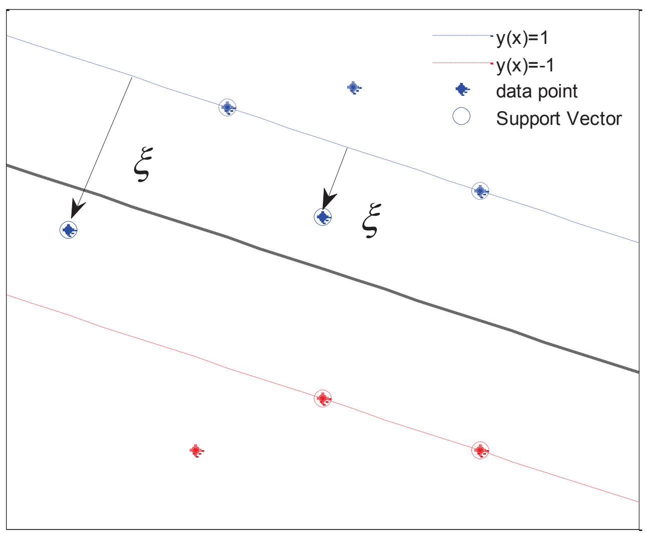

Figure 2. Separating margin of the proposed approach

As previously mentioned, some points may fall inside the margin or on the wrong side of the margin. In such cases, an additional cost will be assigned for allowing such points. To deal with such points, a non-negative slackness variable can be introduced [11],

$$t_i y(\boldsymbol{x}_i) \geq 1 - \xi_i,\ \xi_i \geq 0,\ i = 1, 2, \ldots, N \tag{4}$$

It is observed that the proposed approach only penalizes points that lie inside the margin ($0<\xi_i<1$) and those on the wrong side of the decision function ($\xi_i \geq 1$).

In order to control the norm of the weights, a weight constraint can be applied, so the objective function is to minimize the training error as well as the norm of weights. The objective function can be written as

$$\begin{aligned}&\min Loss(\boldsymbol{w},\boldsymbol{a})=\frac{1}{2}[\sum_{i=1}^{N}E(\xi_i)+\lambda_1\sum_{i=1}^{L}w_i^2+\lambda_2\sum_{i=1}^{L}\sum_{j=1}^{M}a_{ij}^2]+const.\\&subject\ to\ t_i y(\boldsymbol{x}_i)=1-\xi_i, i=1,2,...,N\end{aligned} \tag{5}$$

where $\boldsymbol{w}=[w_1, w_2,\ldots,w_L]^{\mathrm{T}}$ are the output weights and a_{ij} are the input weights, M is the number of input neurons, and λ_1 and λ_2 are the tradeoff parameters. In our specific case, considering the large noise and poor quality of the side scan sonar images, the error function is chosen to be a robust Huber-like function [12], which is defined as

$$E(\xi)=\begin{cases}u\xi-0.5u^2, \xi>u\\0.5\xi^2 \qquad ,0\le\xi\le u\\0 \qquad\quad ,\xi<0\end{cases} \tag{6}$$

Furthermore, we define the matrix **H**

$$\mathbf{H}(\boldsymbol{a}_1,...,\boldsymbol{a}_L,\boldsymbol{x}_1,...,\boldsymbol{x}_N)=\begin{bmatrix}\boldsymbol{\phi}(\boldsymbol{a},\boldsymbol{x}_1)^{\mathrm{T}}\\\vdots\\\boldsymbol{\phi}(\boldsymbol{a},\boldsymbol{x}_N)^{\mathrm{T}}\end{bmatrix}=\begin{bmatrix}g(\boldsymbol{a}_1,\boldsymbol{x}_1)\cdots g(\boldsymbol{a}_L,\boldsymbol{x}_1)\\\vdots \quad \cdots \quad \vdots\\g(\boldsymbol{a}_1,\boldsymbol{x}_N)\cdots g(\boldsymbol{a}_L,\boldsymbol{x}_N)\end{bmatrix}_{N\times L} \tag{7}$$

$\phi(\boldsymbol{a},\boldsymbol{x})$ in (7) is the hidden layer feature mapping. Let $\boldsymbol{y}=[y_1,\ldots, y_N]^{\mathrm{T}}$, therefore, Equation (3) can be compactly written as

$$\boldsymbol{y}=\mathbf{H}\boldsymbol{w} \tag{8}$$

The details of how to train the proposed method is given in Annex A.

4 Experiment

In this section, the performance of the proposed approach is verified and compared to several related learning algorithms. The sonar data used in this study is provided by the Ocean Systems Laboratory, Heriot-Watt University, Edinburgh, UK. The sonar images were collected by an AUV fitted with side scan sonar from a trail on Loch Earn (Scotland) on November 10th and November 11th, 2010. The sonar images gathered on November 10th are used as the training set and the data gathered on November 11th are used as the testing set. The resolution of each sonar image is 500×1024.

4.1 Data Pre-processing

Raw sonar images have to be properly pre-processed before machine learning algorithms are applied. We are only interested in the foreground objects. Therefore, the large amount of background (seabed) data has to be filtered out.

It is reasonable to assume that the foreground objects have a more complex texture than the seabed. Thus, the foreground object areas are obtained by using local range and standard deviation filters [13].

The objective of the image processing procedures at this point is data reduction rather than MLOs detection. Thus, a relatively high false alarm rate is acceptable. Figure 3 shows an example of image processing result. In this example, the blue object is a real mine, while the green is not.

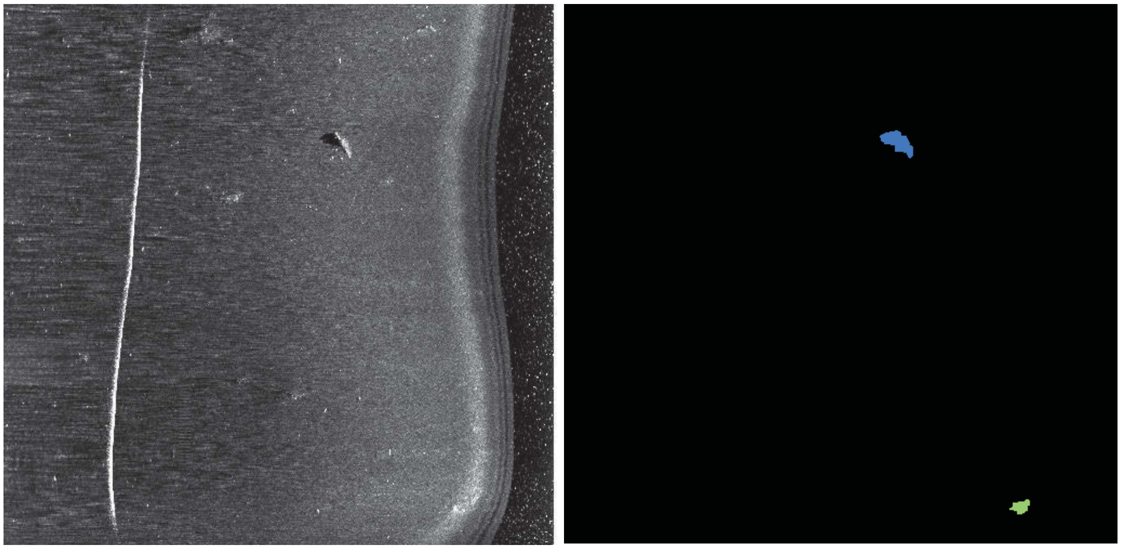

Figure 3: Example of image pre-processing results (blue and green objects are MLOs)

For the object detection task, an object should be detected through a single view, no matter where and how it lies on the seabed. Therefore, the feature used should be robust to the location and orientation of the object. The grayscale histogram, a simple but informative statistical feature, is considered. In many image recognition systems, many complex features are used, but such features will inevitably increase the computational complexity, impeding the detection speed. The histogram is easy to calculate and robust to rotation, moreover, the distribution of the grayscale value can be well described by this feature.

In our experiment, the grayscale value (0-255) is divided into 16 bins with width 16. The grayscale histogram is normalized to the frequency that a pixel value falls into each bin. The dataset information is listed in Table 1.

Table 1: Side Scan Sonar Dataset information

	Training Set	*Testing Set*
# positive instances	18	17
# negative instances	2202	1130
# pos./ # neg.	0.0082	0.0150
Instances in total	2220	1147
# features	16	16

4.2 Experimental Set-up

For the proposed method, the number of hidden neurons L is fixed at 200 and λ_2 is fixed at 0.5. The initial input weights are randomly generated from the input hyper-plane such that for each hidden neuron $i, \sum_{j=1}^{16} a_{ij} = 1$. For other models, the parameters are all optimized.

It is found the dataset is highly imbalanced. When training on highly imbalanced data, many learning methods are likely to focus on learning the majority class, while ignoring the minority. In our experiment, to exclude such impact, the positive and negative instances are weighted differently according to their ratio in the training set for all methods. For a fair comparison, all models are implemented in Matlab, including the QP optimizer for SVMs [14]. The experiment is carried out on the same computer with a 2.00GHz CPU.

4.3 Results

In this study, the AUC (Area Under ROC Curve), is used to quantified the learning results [15]. AUC does not consider the distribution of the positive and negative instances, so it is robust to class imbalance. The TP (True Positive) rate and FP (False Positive) rate are also given for reference. Table 2 shows the comparison of testing results (LR is short for Logistic Regression). Table 3 shows the model parameters used as well as the training and classification time.

From the result tables we can see that the proposed approach is able to beat all other neural network based models in prediction performance. The proposed approach can produce a prediction result close to or even better than the result obtained by kernel methods. In terms of training and classification time, the performance of the proposed method largely outperformed all kernel methods.

Moreover, for kernel machines, the classification speed is directly related to the number of support vectors and the kernel function. When trained on large datasets, the classification speed of kernel machines will tend to be slow. In LS-SVMs and Kernel LR, every data point will be used to build the final decision boundary, so every data point will become a support vector. For SVMs, when using the same hyper parameter, in our case (C, γ), a large dataset is more likely to result in more support vectors. In addition, when the dataset becomes very large, kernel methods could be practically infeasible due to the large kernel matrix inside.

However, similarly to other SLFN methods, the classification speed of the proposed approach is only related to the parameter L, independent of the size of the training set N. Therefore, the classification speed can be controlled by properly setting the value of L when we first build the network.

Table 2: Comparison of performance on Side Scan Sonar Data

Method	*Function*	*TP rate*	*FP rate*	*AUC*
SVMs	Gaussian	0.8824	0.0442	0.9865
LS-SVMs	Gaussian	0.9412	0.0540	0.9885
Kernel LR	Gaussian	0.9412	0.0416	0.9858
BP	Sigmoid	0.9353	0.3145	0.9231
ELM	Sigmoid	0.9059	0.0686	0.9747
PNN	Gaussian	1	0.1693	0.9792
LR	\	0.4117	0.1876	0.7989
Proposed Approach	Sigmoid	0.9177	0.0443	0.9871

Table 3:Model parameters and comparison of time on Side Scan Sonar Data

Method	*Pameters*	*Sparsity(%)*	*Training time(s)*	*Classification time(s)*
SVMs	$C=2^{-5}$, $\gamma=2^{4}$	27.8	94.17	0.9063
LS-SVMs	$C=2^{2}$, $\gamma=2^{2}$	100	13.07	2.1719
Kernel LR	$C=2^{-6}$, $\gamma=2^{0}$	100	33.37	2.2031
BP	$L=20$	/	9.173	0.0313
ELM	$L=40$	/	0.090	<0.01
PNN	$\gamma=2^{4}$	100	/	2.1563
LR	$C=2^{-2}$	/	2.093	<0.01
Proposed Approach	$\lambda_1=2^{-8}$, $\eta=2^{-7}$	48.6	1.123	0.0391

Ideally, an *optimal* classifier is expected to produce the largest AUC value with the shortest time. However, from our result, none of the classifiers, including the proposed one, is able to dominate in both time and accuracy. How to properly balance the time and accuracy (AUC here) depends largely on the real time situation in the Mine Countermeasure Mission application.

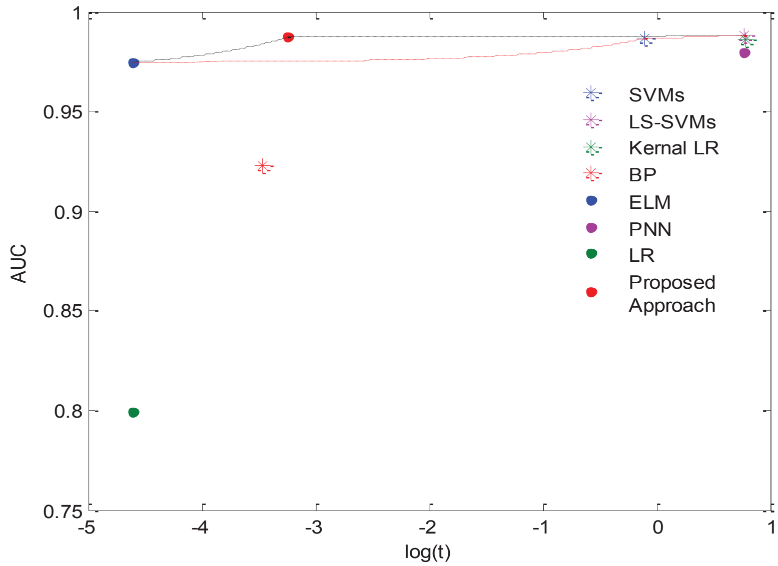

Figure 4: AUC-Time plane

From another perspective, the performance of the each classifier can be projected on the AUC-Time plane shown above. Each point on this plane corresponds to a classifier. The ideal classifier is expected to be located on the top left region of the plane which has the largest AUC value and the fastest detection speed.

Assuming that the AUV can choose from different classifiers in the Mine Countermeasure Mission, any combinations of the existing algorithm can result in a point on or below the red curve on Figure 4. However, the proposed approach can lift the curve, shown by the black dash line, and enable more space on the AUC-Time plane. From this point of view, introducing the proposed approach can improve the efficiency of the MLOs detection.

5 Conclusion

In this work, a novel fast learning method is proposed and applied to the MLOs detection task. The proposed approach borrows the feature mapping idea from kernel methods, but it is implemented via the building of a MLP. In this way, the decision-making space becomes visible and accessible. Together with the inherent sparse representation result from the Huber-like loss, both the learning and prediction processes are largely sped up compared to most traditional methods. Thus, the fast detection requirement in the Mine Countermeasure application can be met. Moreover, unlike the kernel machines, the detection speed of the proposed approach is independent of the size of the dataset, and it can be directly controlled by properly setting the value of L when we first build the network. The experimental result shows that the proposed approach can produce a large AUC value with a fast detection speed.

This page intentionally left blank.

References

[1] Fernandez, J.E., Christoff, J.T., Cook, D.A.: Synthetic aperture sonar on AUV. In: OCEANS 2003. Proceedings, pp. 1718-1722 Vol.1713. (2003)

[2] Glynn, J.M., Buffman, M.: A side scan sonar system for autonomous underwater vehicles. In: Autonomous Underwater Vehicle Technology, Proceedings of the 1996 Symposium on, pp. 154-159. (1996)

[3] Saisan P., Kadambe S.: Shape Normalized Subspace Analysis for Underwater Mine Detection. Image Processing, 15th IEEE International Conference on, pp. 1892-1895 (2008)

[4] Geethalakshmi S. N., Subashini P., Ramya S.: A Study on Detection and Classification of Underwater Mines Using Neural Networks. International Journal of Soft Computing and Engineering 1. 150-157 (2011)

[5] Williams D.P., Groen J., Fox.W.L.J.: A fast detection algorithm for autonomous mine countermeasures. (2011)

[6] Tucker J. D., Azimi-Sadjadi M.R., Dobeck G. J.: Canonical Coordinates for Detection and Classification of Underwater Objects from Sonar Imagery. OCEANS 2007, pp. 1-6. (2007)

[7] Langner F., Knauer C., Jans W., Ebert A.: Side Scan Sonar Image Resolution and Automatic Object Detection, Classification and Identification. OCEANS 2009, pp. 1-8. (2009)

[8] Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks 2, 359-366 (1989)

[9] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature 323, 533-536 (1986)

[10] Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press (2004)

[11] Cortes, C., Vapnik, V.: Support-vector networks. Machine learning 20, 273-297 (1995)

[12] Huber, P.J.: Robust Statistics. John Wiley & Sons, Inc., NJ (1981)

[13] Gonzalez, R., Woods, R.: Digital Image Processing. Addison Wesley (1992)

[14] Canu, S., Grandvalet, Y., Guigue, V., Rakotomamonjy, A.: SVM and Kernel Methods Matlab Toolbox Perception Systèmes et Information, INSA de Rouen, Rouen, France (2005)

[15] Swets, J.A.: Signal detection theory and ROC analysis in psychology and diagnostics : collected papers. Lawrence Erlbaum Associates, Mahwah, NJ (1996)

This page intentionally left blank.

Annex A Optimization Algorithm

The optimization process of the proposed approach can be divided into two steps. The output weights $\boldsymbol{w}$ can be calculated given the input weights a_{ij}, then the input weights can be updated based on the value of output weights. Such a process can be iteratively performed up to a desired convergence. In our experiment, one iteration is performed as it generates satisfying results.

For the input weights, the gradient descent is used to conduct the optimization We can use the chain rule to propagate $Loss(\boldsymbol{w},\boldsymbol{a})$, defined by (5), back to the input layer. Taking the derivative of $Loss(\boldsymbol{a},\boldsymbol{w})$ with respect to a_{ij}, we have

$$\frac{\partial Loss(\boldsymbol{a},\boldsymbol{w})}{\partial a_{ij}} = \frac{\partial E}{\partial a_{ij}} + \lambda_1 \frac{\partial \|\boldsymbol{w}\|^2}{\partial a_{ij}} + \lambda_2 a_{ij} . \tag{A.1}$$

While optimizing $\boldsymbol{a}$ based on the current output weights, we consider $\frac{\partial \|\boldsymbol{w}\|^2}{\partial a_{ij}} = 0$. Now in (A.1) the only unknown term is $\frac{\partial E}{\partial a_{ij}}$. Let net_i^h be the input of hidden neuron i (the value before passing the sigmoid activation function). Since there is no activation function in the output layer, so the output neuron will copy its input net^o as the final network output $y(\boldsymbol{x})$, we have

$$net_i^h = \sum_j a_{ij} x_j = \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x},\ net^o = y(x) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x}) . \tag{A.2}$$

Using the chain rule, we have

$$\begin{aligned} \frac{\partial E}{\partial a_{ij}} &= \frac{\partial E}{\partial net_i^h} \times \frac{\partial net_i^h}{\partial a_{ij}} \\ &= \frac{\partial E}{\partial net_i^h} \times x_j \end{aligned} \tag{A.3}$$

where x_j is the jth attribute value of the input data. Furthermore,

$$\frac{\partial E}{\partial net_i^h} = \sum_{j \in output\ layer} \frac{\partial E}{\partial net_j^o} \times \frac{\partial net_j^o}{\partial net_i^h} . \tag{A.4}$$

Since only one output neuron will be used in binary classification, equation (A.4) can be written as

$$\frac{\partial E}{\partial net_i^h} = -\delta^o \times \frac{\partial net^o}{\partial net_i^h} \quad . \tag{A.5}$$

where the delta term for the output neuron is defined as $\delta^o = -\frac{\partial E}{\partial net^o}$. $\boldsymbol{\phi}_i(\boldsymbol{x})$ is the output of hidden neuron i, equation (A.5) can be written as

$$\begin{aligned}\frac{\partial E}{\partial net_i^h} &= -\delta^o \times \frac{\partial net^o}{\partial net_j^h} \\ &= -\delta^o \times \frac{\partial net^o}{\partial \boldsymbol{\phi}_i(\boldsymbol{x})} \times \frac{\partial \boldsymbol{\phi}_i(\boldsymbol{x})}{\partial net_i^h} \\ &= -\delta^o \times w_i \times \boldsymbol{\phi}_i(\boldsymbol{x})[1-\boldsymbol{\phi}_i(\boldsymbol{x})]\end{aligned} \tag{A.6}$$

It is worth mentioning that for the first order derivative of sigmoid function, $\frac{\partial g(x)}{\partial x} = g(x)[1-g(x)]$ .This property is used in the above equation. Furthermore, define

$$\delta_i^h = -\frac{\partial E}{\partial net_i^h} = \delta^o \times w_i \times \boldsymbol{\phi}_i(\boldsymbol{x})[1-\boldsymbol{\phi}_i(\boldsymbol{x})]. \tag{A.7}$$

Since $E(\xi)$ is a Huber-like function, we have

$$\begin{aligned}&\delta^o = (1-t_i y(x_i)) \times \mathrm{W}_i \\ &\mathrm{W}_i = \begin{cases} u/\xi_i, \xi_i > u \\ 1 \quad , 0 \le \xi_i \le u \\ 0 \quad , \xi_i < 0 \end{cases}\end{aligned} \tag{A.8}$$

Considering (A.3), (A.7) and (A.8), equation (A.1) becomes

$$\begin{aligned}\frac{\partial Loss(\boldsymbol{a},\boldsymbol{w})}{\partial a_{ij}} &= \frac{\partial E}{\partial a_{ij}} + \lambda_2 a_{ij} \\ &= -\delta_i^h x_j + \lambda_2 a_{ij}\end{aligned} \tag{A.9}$$

As mentioned above, the input weights are optimized according to gradient descent, so a_{ij} is updated according to the gradient direction. Therefore, the learning rule for the input weights is

$$a_{ij} \Leftarrow a_{ij} + \Delta a_{ij}, \Delta a_{ij} = -\eta \frac{\partial Loss(\boldsymbol{a}, \boldsymbol{w})}{\partial a_{ij}} = \eta(\delta_j^h x_i - \lambda_2 a_{ij}) \tag{A.10}$$

where η is a constant.

Unlike the input weights that have to go through a sigmoid hidden layer, the output weights can be solved in closed form. With proper reformulation, Equation (5) can be rewritten as

$$Loss(\boldsymbol{w}, \boldsymbol{a}) = \frac{1}{2}[\| \mathbf{W}(\mathbf{H}\boldsymbol{w} - \mathbf{T}) \|^2 + \lambda_1 \| \boldsymbol{w} \|^2 + \lambda_2 \| \boldsymbol{a} \|^2] \tag{A.11}$$

where **W** is a diagonal matrix whose diagonal elements are W_i defined by (A.8).

For simplicity, consider $\| \boldsymbol{a} \|^2$ is fixed and set the derivative of $Loss(\boldsymbol{w},\boldsymbol{a})$ with respect to $\boldsymbol{w}$ to zero:

$$\begin{aligned} \frac{\partial Loss(\boldsymbol{a}, \boldsymbol{w})}{\partial \boldsymbol{w}} &= -\mathbf{H}^{\mathrm{T}}\mathbf{W}(\mathbf{T} - \mathbf{H}\boldsymbol{w}) + \lambda_1 \boldsymbol{w} \\ &= \mathbf{H}^{\mathrm{T}}\mathbf{W}\mathbf{H}\boldsymbol{w} + \lambda_1 \boldsymbol{w} - \mathbf{H}^{\mathrm{T}}\mathbf{W}\mathbf{T} = 0 \end{aligned} \tag{A.12}$$

The output weights $\boldsymbol{w}$ can be solved by Iterative Reweighted Least Square (IRLS)

$$\boldsymbol{w}^{t+1} = (\mathbf{H}^{\mathrm{T}}\mathbf{W}^t\mathbf{H} + \lambda_1 \mathbf{I})^{-1} \mathbf{H}^{\mathrm{T}}\mathbf{W}^t\mathbf{T}. \tag{A.13}$$

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. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.) Nathalie Japkowicz Consulting Services 5714 Queen Mary Rd Hampstead, Quebec H3X 1X6	2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.) UNCLASSIFIED (Non-Controlled Goods) DMC: A Review: GCEC June 2010

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.)

Machine Learning Algorithms for Multiple Autonomous Unmanned Vehicle Operations: A Fast Detection Algorithm

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Shao, H.; Japkowicz, N.

5. DATE OF PUBLICATION (Month and year of publication of document.) April 2013	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.) 32	6b. NO. OF REFS (Total cited in document.) 15

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.)

Contract Report

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

Defence R&D Canada – CORA
101 Colonel By Drive
Ottawa, Ontario K1A 0K2

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.) 10bz04	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.) W7714-115078
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.)	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.) DRDC CORA CR 2013-059

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

Unlimited

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), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

Autonomous Underwater Vehicles (AUVs) are planned to conduct Mine Countermeasure mission in the future. With the help of high resolution imagery produced by the sonar systems mounted on AUVs, the mines and other objects of interest can be detected. In this work, the existing approaches for Mine-Like Objects (MLOs) detection are first reviewed, then considering the limitation of the exiting works, a novel machine learning method is designed for MLOs detection. The experimental result on real side scan sonar images shows that the new learning method is able to result in reliable and fast MLOs detection.

L’utilisation de véhicules sous-marins autonomes (VSA) est prévue pour de futures missions de lutte contre les mines. En effet, des mines et d’autres objets présentant un intérêt peuvent être détectés à l’aide des images à haute résolution produites par les systèmes de sonar installés sur des VSA. Dans le cadre des présents travaux, les approches existantes en matière de détection d’objets ressemblant à une mine sont d’abord examinées, puis, en tenant compte des limites des travaux existants, une nouvelle méthode d’apprentissage automatique est conçue pour la détection de tels objets. Le résultat expérimental sur de vraies images de sonar latéral démontre que la nouvelle méthode d’apprentissage peut permettre une détection fiable et rapide des objets ressemblant à une mine.

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.)

AUV; Unmanned Vehicles; Autonomous Vehicles; Machine Learning; Detection

Defence R&D Canada

Canada's Leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE DÉFENSE

www.drdc-rddc.gc.ca