Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 7, No. 1, February 2016), Pages: 143-156
www.jacr.iausari.ac.ir

Classification of Sonar Targets Using OMKC, Genetic Algorithms and Statistical Moments

Mohammad Reza Mosavi✉[1], Mohammad Khishe[1], Ehsan Ebrahimi[2]

1) Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran
2) Department of Electrical Engineering, Imam Khomeini University of Maritime Sciences, Nowshahr, Iran

m_mosavi@iust.ac.ir; m_khishe@elec.iust.ac.ir; ehsanebrahimi1985@yahoo.com

Received: 2015/03/13; Accepted: 2015/08/18

Abstract

Due to the complex physical properties of the detected targets using sonar systems, identification and classification of the actual targets is among the most difficult and complex issues of this field. Considering the characteristics of the detected targets and unique capabilities of the intelligent methods in classification of their dataset, these methods seem to be the proper choice for the task. In recent years, neural networks and support vector machines are widely used in this field. Linear methods cannot be applied on sonar datasets because of the existence of higher dimensions in input area, therefore, this paper aims to classify such datasets by a method called Online Multi Kernel Classification (OMKC). This method uses a pool of predetermined kernels in which the selected kernels through a defined algorithm are combined with predetermined weights which are also updated simultaneously using another algorithm. Since the sonar data is associated with higher dimensions and network complexity, this method has presented maximum classification accuracy of 97.05 percent. By reducing the size of input data using genetic algorithm (feature selection) and statistical moments (feature extraction), eliminating the existing redundancy is crucial; so that the classification accuracy of the algorithm is increased on average by 2% and execution time of the algorithm is declined by 0.1014 second at best.

Keywords: *Sonar, Classification, OMKC, Genetic Algorithm, Statistical Moments, Clutter*

1. Introduction

Detected target by Sonar systems includes: true target, noise, reverberation and clutter. Noise can be categorized as thermal, ambient, continental, harbors, etc [1]. Since the noise dosn't have Sonar ping genre, it can be easily distinguished from the true target. Echoes, also called reverberation, are the result of reflected sonar pings created from their collision with sea floor and surface [2]. Considering the fact that echoes have equal and homogeneous amplitude, they can be easily distinguished from actual targets because the genre is made of sonar ping [3].

In case that the sea floor surface is uneven, especially in small dimensions, there will be various reflected echoes coming from that surface which contain features comparable to the features of actual target such as the Probability Density Function (PDF). These false targets are called clutter [4].

Due to the so many similir features between clutters and actual targets their distinction is very difficult. In recent years, neural networks and support vector machines have become a common method for adressing this problem and various attempts have been made for solving that [5].

Two active research topics in category of machine learning are online and kernel learning, which have been studied separately for several years. Online learning is designed to develop a prediction model based on sequential feedbacks from previous questions and possible additional information [6]. This method is distinguished from typical supervised learning algorithms which are designed to develope classification models from a collection of given training examples. The purpose of kernel learning is to find an effective kernel function for a given learning task using training data [7-9]. An example of kernel learning is Multiple Kernel Learning (MKL) [10], which finds the optimal combination of multiple kernels in order to incresae the performance of kernel based learning methods (as Figure 1). Againest the various existing algorithms [11], several studies are dedicated to compare the kernel techniques in online learning settings [12]. However, in most of the existing kernel based online learning algorithms, the kernel function is assumed to be specified, which is significantly limited their real-world applications. As an attempt to overcome this limitation, Online Multiple Kernel Classification (OMKC), which is aimed to simultaneously train multiple kernel classifiers and their optimal linear combinationes, is used in this paper. The main challenge of OMKC algorithm is that the both optimal kernel classifiers and their linear combinations are required to be trained simultaneously. More importantly, the solution of kernel classifiers and their linear combinations are strongly correlated which makes the problem more challenging compared to a typical online learning problem. Kernels fuse two kinds of online learning techniques: the perceptron algorithm [13] that traines a classifier for a given kernel and the Hedge algorithm [14] that linearly combines multiple classifiers. Furthermore, in this paper a novel kernel selection strategy has been developed for choosing a random subset of kernels for model updating and combination tasks, which results in significant improvement in learning efficiency.

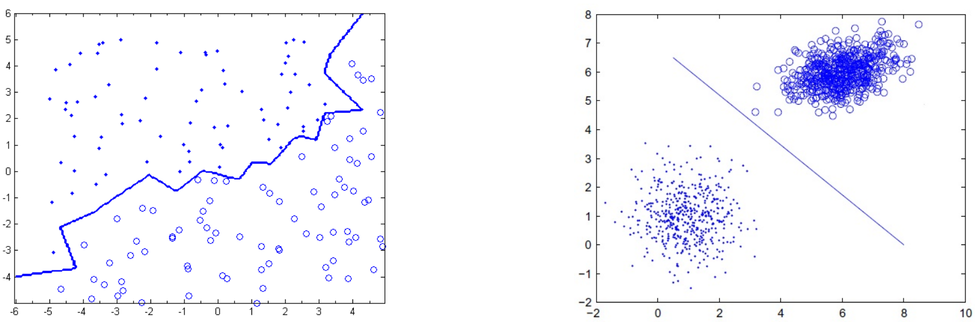


Figure 1. Converting non-linear input space to linear one by use of kernels.

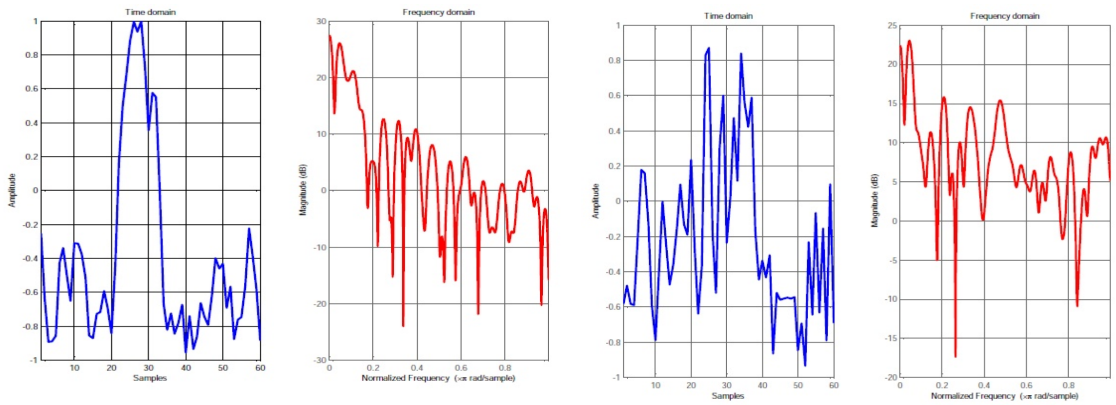
However this method was proved to be inefficient due to the high computational time and complexity. High dimensions of Sonar data and existing redundancy decreases the performance of algorithm for target classification purposes. One way to overcome this problem, is to reduce the dimension size of the input dataset using Genetic Algorithm (GA) and Statistical Moments (SM) [15]. Also selecting the kernels using stochastic methods and updating the elite kernels rather than the whole kernels, have been used properly in the original algorithm. Using these simple methods, the spatial and time complexity reduce to an acceptable level.

The rest of the paper is organized as follows. First, Sonar data is defined and the manner in which the data dimension is redusced using GA and SM will be studied. Section two describes the classic OMKC algorithm. In section 3, the modified algorithm has been explained and compared with benchmarks methods that have been utilized in articles with high impact factor, as well as the standard OMKC method. Conclusions and suggestions for the further works are represented in section 4.

2. Sonar Dataset

The data used in this paper are extracted from Gorman and Sejnowski researches [6,7]. In this study, a 5 feet metal cylinder and a rock with the same size of the sandy floor of the sea have been placed accordingly and a broadband linear FM pulse chirp (ka = 55/6) was sent to them. Reflected echoes within 10 meters of the test objects have been collected and 208 echoes between 1200 samples were chosen based on their SNR which should been within the range of 4dB and 15dB. Amonge the collected echoes, 111 were reflected from metal cylinders and the others were received from rocks.

An example of reflected echoes PDF function received from rock and metal cylinder is shown in Figure 2. As it can be seen, reflected echoes from diffrent targets have different characteristics in terms of the frequency and time domain. These differences shape the idea of extracting new features for classification purposes based on SM methods which are described in the next section.



Rock ***Metal Cylinder***

Figure 2. Examples of the reflected echoes from rock and metal cylinder.

2.1. Reducing the Data Dimension using GA and SM

Considering high dimensions of the Sonar data (which is equal to 60 in this paper) and OMKC network complexity, occurrence of numerous redundancies in the process of target classification can not be avoided. The result of these redundancies are reduced performance of the network's classification capability and it's increased time complexity.

Using conventional methods of feature extraction (methods based on SM) along with methods for choosing the features with high impact on target classification (GA), dimension of the input data can be reduced from D to d where $d < D$, so that information processing time is reduced and the network classification accuracy is increased [16].

Selecting the value of *d* is the other challenging issue of this method. In this paper, scree test [17] is used to determine the value of *d*. The point where the graph has the sharpest drop will be selected, but cannot be proven yet until it is chosen in small amounts so that the selection of *d* is possible. The result of this experiment is depicted in Figure 3. As can be seen *d* is set to 6 for sonar data.

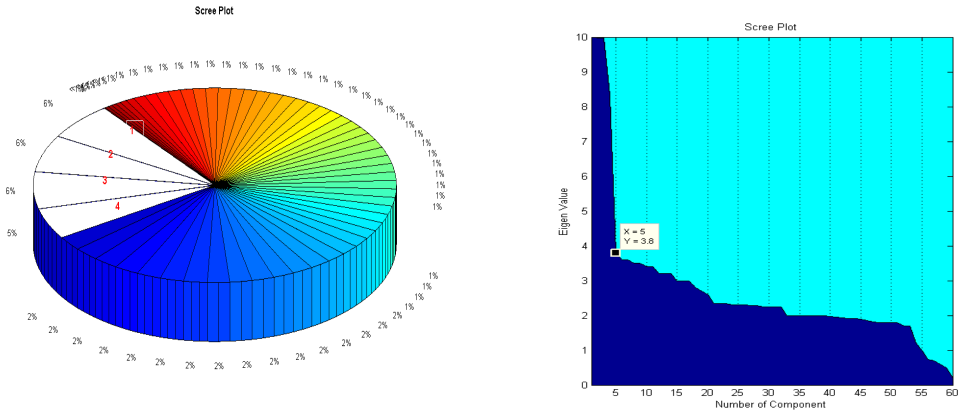


Figure 3. The result of the scree test on the Sonar data.

It has been proved [18] that the noises, reverberations, clutters or reflected echoes from targets, possess their own PDF functions with parameters that are almost identical. The PDF functions related to reverberation, clutter and actual target are rayleigh, K-distribution and dominant one plus rayleigh (1D+rayleigh), respectively [19].

Considering the differences between PDF functions of true and false targets, and also the fact that the third (Skewness) and fourth (Kurtosis) moments are changed as their PDF fuctions are changed.

In this part, using the main data, instead of taking 60 samples using the PDF function related to each test, the third and fourth moments are calculated. Each moment is considered as a new feature. For an amplitude distribution of $(x_1,...,x_N)$ with the mean value of $\overline{x}$, and the variance of σ, third and fourth moments are calculated using equations (1) and (2), respectively:

$$\sigma^2 = (1/N)\sum_{m=1}^{N}(x_m - \overline{x})^2 \quad (1)$$

$$\text{k-th moment} = ((1/N)\sum_{m=1}^{N}(x_m - \overline{x})^k)/\sigma^k \quad (2)$$

In order to have an optimal result, GA is used to select the other four required features between the 60 available features more effectively. Techniques based on information theory for selecting variables in the time series prediction and model statistical detection are very useful. These techniques are based on the maximization of mutual information between the input and output of the system.

However, the procedure of finding the entropy requires a lot of computational power. To avoid this requirement, indirect methods can be used not only to minimize the redundancy, but also to maximize the mutual information. Moreover, the combined optimization methods (i.e. examine all the variables) also involve quite a lot of calculations. In this paper, GA is used to avoid this problem [20].

The main idea of this approach is to select a set of N_n features as $\{n_i\}, i=(1,2,...,N_n)$ and then try to maximize the mutual information $I(y; yh_1,..., yh_{N_n})$ using selected features, between the target (the class) and outputs yh_i which are resulted from n_i. Equation (3) shows the number of solutions for the combinatorial optimization where M is the total number of attributes.

$$\frac{M!}{2\left(M-N_n\right)!} \tag{3}$$

In order to reduce the required calculations combined optimization for $I(y; yh_1,..., yh_{N_n})$ has been implemented using modified GA in which instead of liner combination of parents in crossover process, one of the parents is chosen randomly. Fitness function has been chosen so that it can maximize the relevancy and minimize the redundancy. The fitness function is given by following equation [20]:

$$\max_{i_1 \dots i_{N_n}} \Phi \tag{4}$$

In which $\Phi = V - P$ and

$$V = \frac{1}{N_n} \sum_{i=1}^{N_n} I(yh_i; y) \tag{5}$$

$$p = \frac{1}{N_n^{\,2}} \sum_{j=1}^{N_n} \sum_{i=1}^{N_n} I(yh_i; yh_j) \tag{6}$$

The V and P represented in equations (5) and (6) are representing the relation and redundancy between N_n features, respectively. Note that V is the average value of mutual information $I(yh_i; y)$ between classes and selected features outputs and P is the average value of mutual information $I(yh_i; yh_j)$ between selected features outputs.

3. Online Multi Kernel Classification

Two active research topics in category of machine learning are online and kernel learning, which have been studied separately for several years. Online learning is designed to develop a prediction model based on sequential feedbacks from previous questions and possible additional information [6]. This kind of algorithms are different from supervised learning algorithms in which a set of training data are used for training the classification model [21].

Training an efficient kernel function is the aim of the kernel based learning. An example of such algorithm is the MKL in which the optimal combinations of kernel function are sought in order to achive the highest possible performance. In most online methods it is assumed that the kernel function is predefined which makes them more preferable to be employed in real world issues [22,23].

In this paper, OMKC algorithm is used to overcome this limitation by training the Multi Kernel Classifier (MKC) and optimizing their linear combination simultaneously. This is the main challenge associated with OMKC algorithm. The close relation between the classifiers and their coefficients and also the need for their simultaneous update increses the complexity of purposed algoritm with respect to other methods. For this reason the OMKC algorithm focuses on two online training method as: a) the perceptron, which trains a classifier for given kernals [24,25], and b) the Hedge [26,27], which linearly combines the classifiers [24].

In this section, the standard multi kernel learners algorithm is presented [25] and it will be used for developing the OMKC algorithm in the next section [21,28]. Training set is defined as $D=\{(x_i, y_i), i=1,2,....,n\}$, $x_i \in R^d$, $y_i \in \{-1,1\}$ $i=1,...,n$ and a collection of m kernel functions as $k=\{\kappa_i : \chi \times \chi \to R, \mathrm{i}=1,...,\mathrm{m}\}$. The goal of MKL is to train a kernel-based prediction function through calculating the optimal coefficients for combining the kernels, denoted by $\theta=(\theta_1,...,\theta_m)$. The optimization is performed by minimizing the classification error with the cost function given by equation (7) [21]:

$$\min_{\theta\in\Delta} \min_{f\in H_{\kappa(\theta)}} \frac{1}{2}|f|^2_{H_{\kappa(\theta)}} + C\sum_{i=1}^{n}\ell(f(x_i), y_i) \tag{7}$$

Where

$$\Delta=\{\theta\in\mathbb{R}_+^m \mid \theta^T 1_m = 1\}$$

$$K(\theta)(.,.)=\sum_{i=1}^{m}\theta_i\kappa_i(.,.)$$

$$\ell(f(x_i), y_i)=\max(0, 1-y_i f(x_i))$$

In the above equation, 1_m refers to an m-dimensional vector that all of its elements are one. The general block diagram of this method is shown in Figure 4.

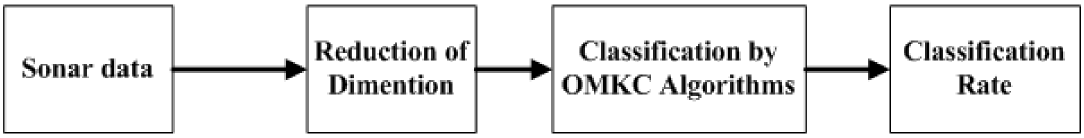


Figure 4. The general trend for proposed algorithm.

3.1. Online Multi Kernel Classifier

The proposed OMKC structure is based on the combination of two learning methods: perceptron and Hedge algorithms. Perceptron algorithm trains the classifier based on kernels and Hedge algorithm updates the weights for their linear combination. Figure 5 illustrates the proposed algorithm in detail.

In this format, $w_i(\mathrm{t})$ is the weight of the i-th kernel classifier at time t, which is set to 1 for $t=0$. The weights are updated using the Hedge algorithm in equation (8):

$$w_{t+1}^j = w_t^j \beta^{z_t^j} \tag{8}$$

Where $\beta\in(0,1)$ is the weight decline coefficient and is used to modify the kernels with inaccurate predictions. If the result of the classification process is satisfactory,

$Z_i(\mathrm{t})$ takes the values of 0 and the weights are remain unchanged. Otherwise $Z_i(\mathrm{t})$ is set to 1 and a decreased weight according to the value of β is replaced with the old one.

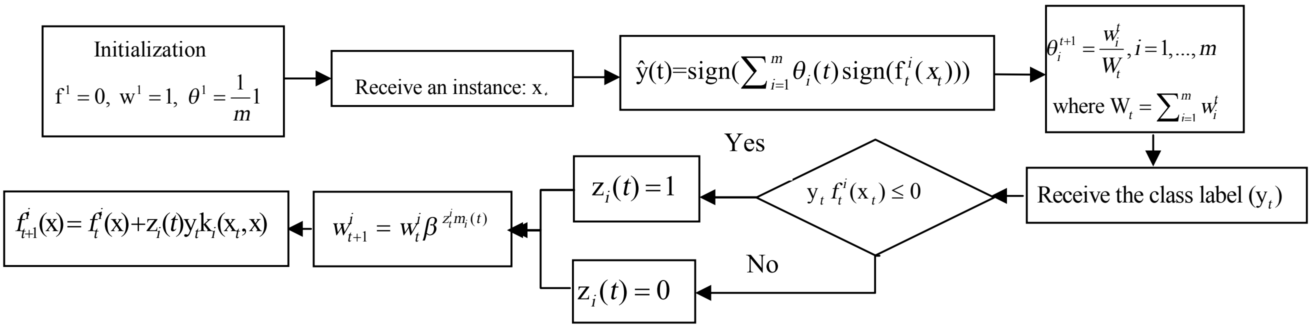


Figure 5. Procedure of OMKC$_{(D,D)}$.

The results indicates the inefficiency of the algorithm 1 due to the its complexity and execution time. In the next section, new algorithms are developed for compensating this defect.

3.2. OMKC Algorithm by Kernels Stochastic Combination

The first attempt for improving the computational efficiency of the algorithm 1, is to select a subset of kernels and use combination of the selected elements. Figure 6 shows the detailed steps. Where $q_i(t)$ which is the selection probability of the *i*-th kernel is calculated by equation (9):

$$q_i(t) = \mathrm{w}_i(t) / \left[\max_{1\le j\le m} \mathrm{w}_j(t) \right] \tag{9}$$

Only selected the selected kernels are combined in this method. This procedure is called OMKC$_{(D,\ S)}$, which represents stochastic selection and deterministic update of kernels based on their involvement in combination.

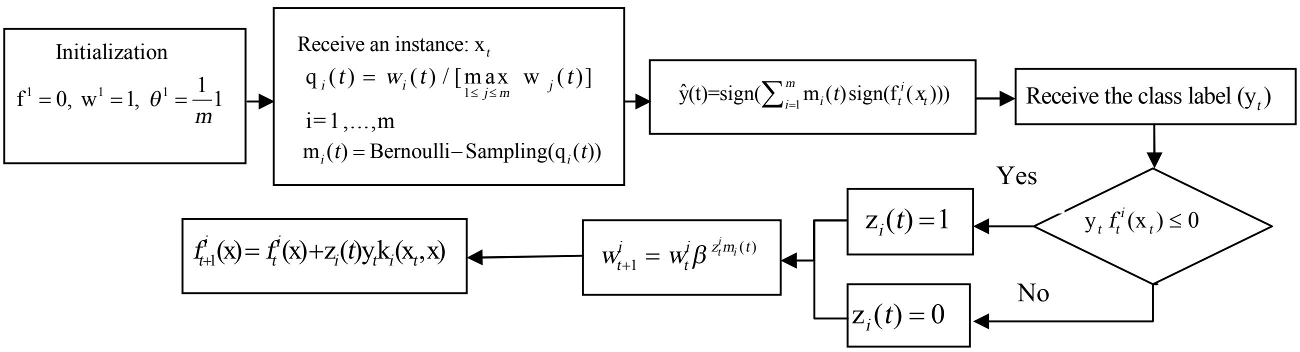


Figure 6. Procedure of OMKC$_{(D,S)}$.

3.3. OMKC by Updating Stochastic Weights

The next approach for imroving the efficiency of the algorithm 1, is to only update the weights of the stochastically selected kernals rather than updating all of them. To do so, $p_i(t)$ is defined as:

$$p_i(t)=(1-\delta)q_i(t)+\delta/m,\ i=1,...,m \tag{10}$$

The smoothing parameter δ in equation (10) ensures that the sampling probability of all kernels be at least δ/m. According to equation (10) and bernoulli's sampling process given by equation (11) a subset of kernels are selected and their corresponding weights are updated.

$$m_i(t)=\text{Bernoulli}-\text{Sampling}(p_i(t)) \tag{11}$$

In this equation, $m_i(t)\in\{0,1\}$ specifies the result of sampling, so that the *i*-th kernel is selected, if and only if $m_i(t)=1$. Figure 7 shows the mentioned procedure in details. Thus this procedure is called $OMKC_{(S,\ D)}$, which represents deterministic selection and stochastic update of kernels based on their involvement in combination.

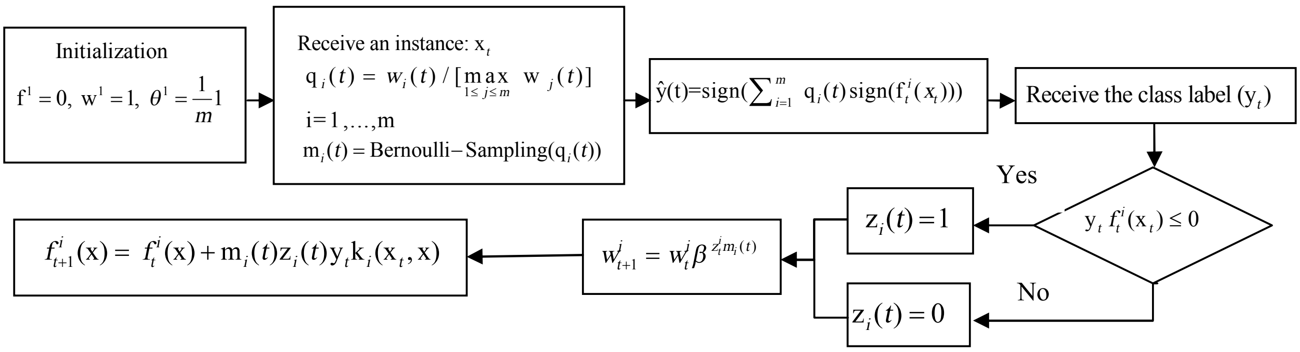


Figure 7. Procedure of $OMKC_{(S,D)}$.

3.4. OMKC with Stochastic Combination and Updating

Employing two previous methods simultaneously is the last strategy used in this paper to improve the performance of OMKC algorithm is a combination of the two previous stages (i.e. stochastic selection of kernels based on their involvement in learning process and stochastic update of the weights between kernels). This method is called $OMKC_{(S,S)}$. Combining these two methods clearly reduces the computational complexity which also results in reduction of the time complexity classification accuracy. Details of the $OMKC_{(S,S)}$ algorithm are illustrated in Figure 8.

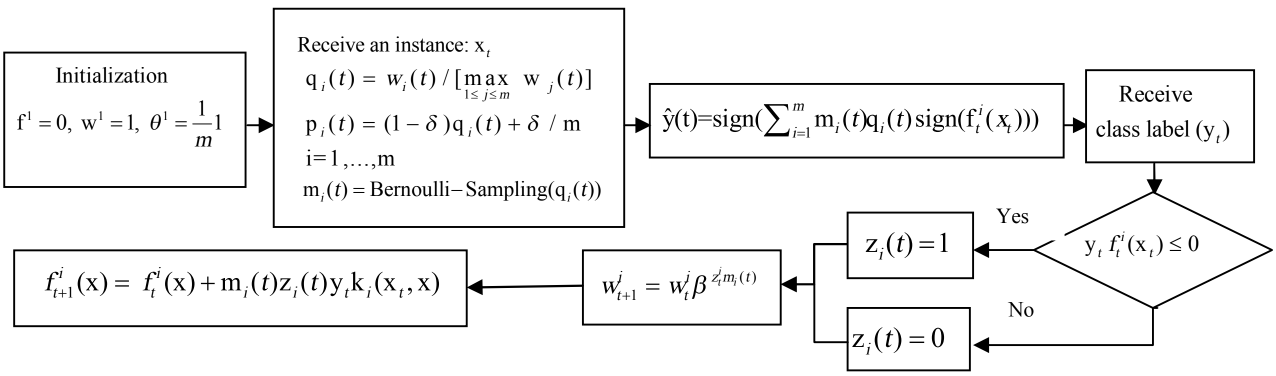


Figure 8. Procedure of OMKC$_{(S,S)}$.

4. Simulation Results

The simulation section is aimed to answer these questions: 1) whether the proposed OMKC algorithms for online classification has higher efficiency compared to the conventioonal mono kernel algorithms (i.e. perception) or not?, 2) are the efficiency of the proposed OMKC algorithms surpass the standard MKL methods for classification purposes or not?, 3) how the performance of the OMKC algorithm is effected while using stochastic kernels selection and stochastic weights update? and 4) what is the effect of reducing the dimension of input data in algorithm's classification accuracy and its execution time?

4.1. Kernels Pool

Using the past experiences and results of previous researches experiments [28,29], the kernel pool is formed consisting of 16 kernel functions, including 3 polynomial kernels (i.e. $k(\mathrm{x}_i, \mathrm{x}_j) = (\mathrm{x}_i^T \mathrm{x}_j)^P$) with *P=1,2,3* degrees and 13 Gaussian kernels (i.e. $k(\mathrm{x}_i, \mathrm{x}_j) = \exp(-\|\mathrm{x}_i - \mathrm{x}_j\|^2 / 2\delta^2)$) with the δ parameter width between $[2^{-6}, 2^{-5}, ..., 2^{5}, 2^{6}]$.

4.2. Comparison

In order to examine the proposed algorithm's performance, it is compared to 4 benchmark algorithms which are widely used in papers. These algorithms are: a) perceptron with linear kernels, b) perceptron (u): perceptron with bias less and uniform composition kernels [30], c) perceptron (*): in this algorithm ten percent of the training data is used to determine the most suitable kernels and the other 90 percent is used to train the network [31], and d) OM-2: which is a benchmark algorithm and a subset of MKL [32].

4.3. Measurement Metrics

For examining the designed network in terms of accuracy, time complexity and structural complexity, following metrics are used respectively: a) error rate: the percentage of occurred errors in prediction process or online receiver b) the algorithm's execution time which includes model update and online prediction time, and c) norm-2 of the classifier's support vectors.

4.4. Set Parameters

To perform the comparison between the algorithms with equal conditions and obtain the best performance using different calculations and sequential experiments [33], the parameters β and δ are set to be 0.8 and 0.01, respectively [34-36]. To obtain an stable average, all online experiments are performed recursively with 20 iteration for each data set. It should be noted that all simulations are executed using a PC with 2.3GHz processor and 16GB of RAM in Matlab's simulation environment.

4.5. Tests Results

Test results for raw sonar data and the data with reduced dimensions are represented in Table 1. The comparison between OMKC algorithm and classical methods is represented in this table. Student t-tests are performed and the best results are highlighted and performance of the different algorithms are discussed. First the performance of three perceptron based algorithms (perceptron, perceptron (u) and perceptron (*)), OM-2 and classic OMKC are examined.

As can be seen, OMKC algorithm provides higher classification accuracy compared to other methods. However, in this method the complexity of the network is increased and its performance is reduced due to the existent redundancies. It is clear from the second row of the table that decreasing the dimension of input dataset increses the classification accuracy and reduces dataset processing time. Table 2 also indicates that the stochastic selection and composition of the kerneles results in increased classification accuracy through reducing the network complexity and available redundancies.

From the results represented in Table 2 it can be concluded that kernels deterministic selection and their stochastic update leads to a relatively better outcome. It is also observed that using the data with reduced dimensions results in higher accuracy and reduced computational time. The reason is that, through reducing the data dimension, additional information and their resultant redundancies is reduced and calculations accuracy is increased due to the reduction of calculations. However the effect of this action is limited because of the small ratio of data to examples.

Figure 9-a shows the sonar data classification error for different algorithms with and without utilizing GA and SM. Mean value of the support vectors and execution time of the algorithms are depicted in Figures 9-b and 9-c, respectively. The horizontal and vertical axes of these figures indicat different classification algorithms such as perceptron, perceptron (u), perceptron (*), OM_2, $OMKC_{(D, D)}$, $OMKC_{(D, S)}$, $OMKC_{(D, S)}$ and $OMKC_{(S, S)}$ and their comparison metrics, respectively.

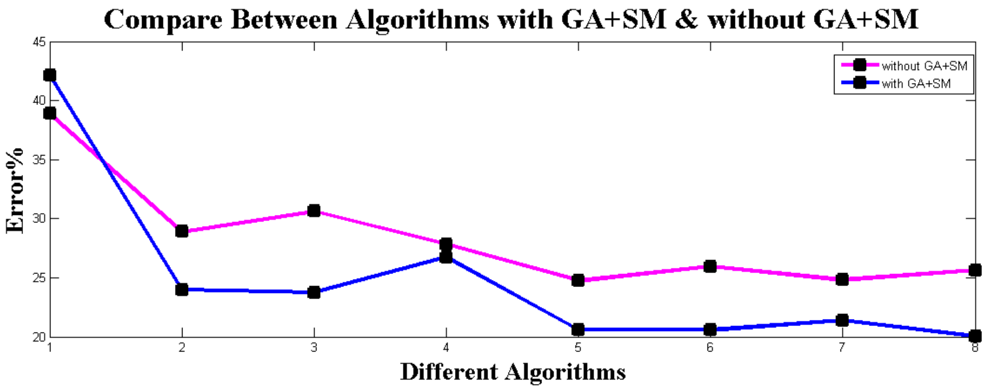


Figure9-a. Comparison of error rates for Sonar data using the GA + SM without the use of GA + SM.

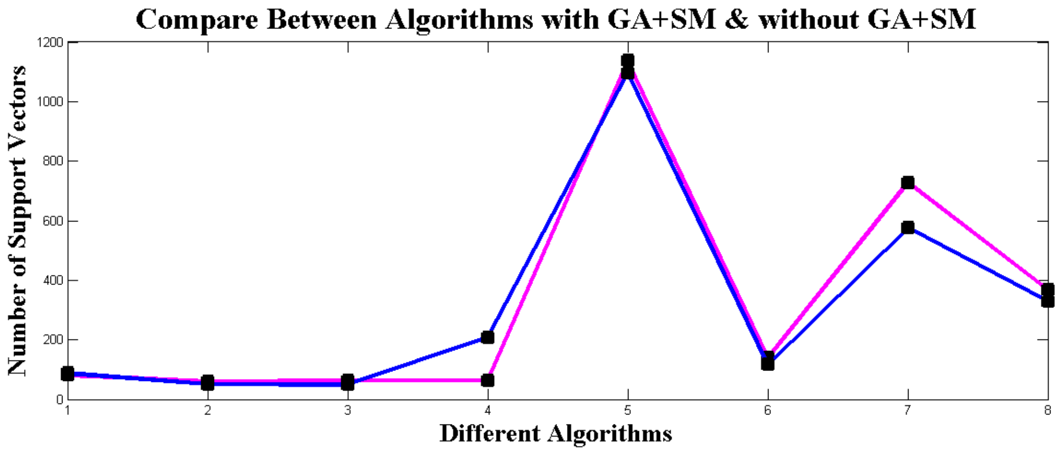


Figure9-b. Comparison of the mean support vector for Sonar data in various algorithms.

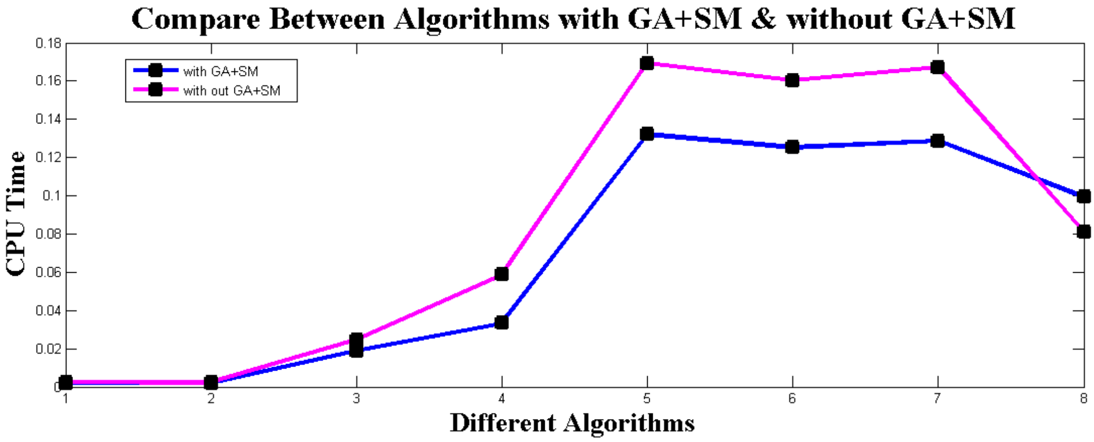


Figure9-c. Comparison algorithm runtime for Sonar data in various algorithms.

Table 1. Comparison of OMKC with classical methods.

Data	Metrics	Perceptron	Perceptron (u)	Perceptron(*)	OM-2	$OMKC_{(D,D)}$
Sonar	Mistake (%)	38.89 ± 2.23	28.89 ± 2.07	30.65 ± 6.66	27.84 ± 1.84	**24.74 ± 1.84**
	SV (#)	80.9 ± 4.6	**60.1 ± 4.3**	63.8 ± 13.9	63.3 ± 4.4	1136 ± 38.6
	Time (s)	**0.0019 ± 0.005**	0.0019 ± 0.0007	0.0188 ± 0.001	0.0335 ± 0.001	0.1321 ± 0.003
Sonar data with decline dimensions	Mistake (%)	42.12 ± 77	23.99 ± 3.00	23.75 ± 8.47	26.78 ± 4.05	**20.58 ± 2.32**
	SV (#)	87.6 ± 3.7	**49.9 ± 6.2**	49.4 ± 17.5	208.0 ± 0.0	1097.0 ± 77.1
	Time (s)	0.0026 ± 0/0002	**0.0023 ± 0.0001**	0.0246 ± 0.0021	0.0587 ± 0.006	0.1694 ± 0.0249

Table 2. Comparison of various OMKC methods with each others.

Data	Metrics	OM-2	$OMKC_{(D,D)}$	$OMKC_{(D,S)}$	$OMKC_{(S,D)}$	$OMKC_{(S,S)}$
Sonar	Mistake (%)	27.84 ± 1.84	24.74 ± 1.84	25.96 ± 1.88	24.83 ± 2.1	25.65 ± 1.59
	SV (#)	63.3 ± 4.4	1136 ± 38.6	139.1 ± 75.5	728.1 ± 28.5	366.6 ± 159.6
	Time (s)	0.0335 ± 0.001	0.1321 ± 0.003	0.1252 ± 0.004	0.1287 ± 0.005	0.0944 ± 0.004
Sonar data with decline dimensions	Mistake (%)	26.78 ± 4.05	**20.58 ± 2.32**	20.58 ± 2.07	21.39 ± 2.25	**20.05 ± 2.51**
	SV (#)	208.0 ± 0.0	1097.0 ± 77.1	**117.8 ± 62.3**	57.60 ± 37.0	329.7 ± 76.4
	Time (s)	0.0587 ± 0.006	0.1694 ± 0.0249	0.1603 ± 0.0184	0.1673 ± 0.0253	0.0809 ± 0.0028

5. Conclusion

In this paper, OMKC algorithm is used for distinguishing between actual targets detected by sonar system and clutter. This method increases the classification accuracy of the detected targets by inceasing the data dimension using various kernels. Using these kernels and data with high dimensions results in various redundancies which in turn reduces the networks performance. To solve the problem, GA and SMs are used as a preprocessing method. It is observed that in this case the classification accuracy of the eight mentioned algorithms are improved and the number of support vectors which is a metric of measuring the networks complexity is reduced. Also the processing time is slightly improved in OMKC algorithms and remained almost the same for others.

Acknowledgments

The authors are thankful to Nasser Asadi and Homa Asadi for their valuable support during the preparation of paper.

References

[1] M. R. Mosavi, M. Khishe and H. Hardani, "Classification of Sonar Targets using OMKC," Iranian Journal of Marine Science and Technology, Vol.72, No.3, pp.25-35, 2015.

[2] D. P. Williams and E. Fakiris, "Exploiting Environmental Information for Improved Underwater Target Classification in Sonar Imagery," IEEE Transactions on Geoscience and Remote Sensing, Vol.52, No.10, pp.6284-6297, 2014.

[3] M. E. G. D. Colin and S. P. Beerens ,"False-Alarm Reduction for Low-Frequency Active Sonar with BPSK Pulses: Experimental Results," IEEE Journal of Oceanic Engineering, Vol.36, No.1, pp.52-59, 2011.

[4] R. Fandos, A. M. Zoubir and K. Siantidis, "Unified Design of a Feature-based ADAC System for Mine Hunting using Synthetic Aperture Sonar," IEEE Transactions on Geoscience and Remote Sensing, Vol.52, No.5, pp.2413-2426, 2014.

[5] W. Shuang and H. Leung, "A Markov Random Field Approach for Sidescan Sonar Change Detection," IEEE Journal of Oceanic Engineering, Vol.37, No.4, pp.659-669, 2012.

[6] S. Shalev-Shwartz, "Online learning: Theory, Algorithms, and Applications," Ph.D Thesis, Hebrew University, 2007.

[7] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui and M. I. Jordan, "Learning the Kernel Matrix with Semidefinite Programming," Journal of Machin Learning, Vol.5, pp.27-72, 2004.

[8] S. Sonnenburg, G. R¨atsch, C. Sch¨afer and B. Sch¨olkopf, "Large Scale Multiple Kernel Learning," Journal of Machin Learning, Vol.7, pp.1531-1565, 2006.

[9] C. Chen, J. Zhang, X. He and Z. Zhou, "Non-parametric Kernel Learning with Robust Pairwise Constraints," International Journal of Machine Learning and Cybernetics, Vol.3, No.2, pp.83-96, 2012.

[10] F. R. Bach, G. R. G. Lanckriet and M. I. Jordan, "Multiple Kernel Learning, Conic Duality, and the Smo Algorithm," International Conference on Machine Learning, pp.1-9, 2005.

[11] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz and Y. Singer, "Online Passive-Aggressive Algorithms," International Journal of Machine Learning and Cybernetics, Vol.7, pp.551-585, 2006.

[12] A. J. S. Jyrki Kivinen, and R. C. Williamson, "Online Learning with Kernels," IEEE Transactions on Signal Processing, Vol.52, No.8, pp.2165-2176, 2004.

[13] F. Rosenblatt, "The Perceptron: A probabilistic Model for Information Storage and Organization in the Brain," Psychological Review, Vol.65, pp.386-407, 1958.

[14] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learningand an Application to Boosting," Journal of Computer and System Sciences, Vol.55, No.1, pp.119-139, 1997.

[15] M. Khishe and M. Aghababaee, "Reduction of Active Sonar's Clutter by use of Statistical Momemnts Method," 1-th Iranian Conference on Naval Technologies, pp.75-83, 2013.

[16] A. M. de Theije and J. C. Sindt, "Single-Ping Target Speed and Course Estimation using a Bistatic Sonar," IEEE Journal of Oceanic Engineering, Vol.31, No.1, pp.236-243, 2006.

[17] http://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Sonar,+Mines+vs+Rocks).

[18] R. P. Gorman and T. J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets," Neural Networks, Vol.1, pp.75-89, 1988.

[19] L. J. P. van der Maaten, E. O. Postma and H. J. van der Herik, "Dimensionality Reduction: A Comparative Review," MICC, Maastricht University, 2009.

[20] S. Ullman, M. Vidal-Naquet and E. Sali, "Visual Features of Intermediate Complexity and Their Use in Classification," Nature Neuroscience, Vol.5, No7, pp.682-687, 2002.

[21] M. J. Fialkowski and C. R. Gauss, "Methods for Identifying and Controlling Sonar Clutter," IEEE Journal of Oceanic Engineering, Vol.35, No.2, pp.330-354, 2010.

[22] C. A. M. Moll, M. A. Ainslie, F. P. A. Lam and J. Janmaat, "The Most Likely Distribution of Target Echo Amplitudes," International Conference & Exhibition on Underwater Acoustic Measurements: Technologies & Results, pp.363-365, 2010.

[23] O. Ludwig and U. Nunes, "Novel Maximum-Margin Training Algorithms for Supervised Neural Networks," IEEE Transaction on Neural Networks, Vol.21, No.6, pp.972-984, 2010.

[24] S. C. H. Hoi, R. Jin, P. Zhao and T. Yang, "Online Multiple Kernel Classification," Machine Learning, Vol.90, No.2, pp.289-316, 2013.

[25] R. K. Jade, L.K. Verma and K. Verma, "Classification using Neural Network and Support Vector Machine for Sonar Data Set," Intrnational Journal of Computer Trends and Technology, Vol.4, No.2, pp.116-119, 2013.

[26] R. Jin, S. C. Hoi and T. Yang, "Online Multiple Kernel Learning: Algorithms and Mistake Bounds," Lecture Nodes in Computer Science on Algorithmic Learning Theory, Vol.6331, pp.390-404, 2010.

[27] O. Barak and M. Rigotti, "A Simple Derivation of a Bound on the Perceptron Margin using Singular Value Decomposition," Neural Computation, Vol.23, No.8, pp.1935-1943, 2011.

[28] Y. Freund and R. E. Schapire, "Large Margin Classification using the Perceptron Algorithm," Machine Learning, Vol.37, No.3, pp.277-296, 1999.

[29] K. Chaudhuri, Y. Freund and D. Hsu, "A Parameter-Free Hedging Algorithm," Advances in Neural Information Processing Systems, Vol.22, pp.297-305, 2009.

[30] S. C. H. Hoi, J. Wang and P. Zhao, "Exact Soft Confidence-Weighted Learning," International Conference on Machine Learning, pp.1-8, 2012.

[31] X. Jiejun, V. Jagadeesh and B. S. Manjunath, "Multi-Label Learning with Fused Multimodal Bi-Relational Graph," IEEE Transactions on Multimedia, Vol.16, No.2, pp.403-412, 2014.

[32] C. Gentile, "A New Approximate Maximal Margin Classification Algorithm," Journal of Machine Learning Research, Vol.2, pp.213-242, 2001.

[33] L. Wang, "Feature Selection with Kernel Class Separability," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.30, No.9, pp.1534-1546, 2008.

[34] D. Pachauri, C. Hinrichs, M. K. Chung, S. C. Johnson and V. Singh, "Topology-based Kernels with Application to Inference Problems in Alzheimer's Disease," Vol.30, No.10, pp.1760-1770, 2011.

[35] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz and Y. Singer, "Online Passive-Aggressive Algorithms," Journal of Machine Learning Research, Vol.7, No.2, pp.551-585, 2006.

[36] K. Crammer and Y. Singer, "Ultraconservative Online Algorithms for Multi-Class Problems," Journal of Machine Learning Research, Vol.3, No.1, pp.951-991, 2003.