

# Incremental Learning Algorithm for Support Vector Data Description

Xiaopeng Hua<sup>1,2</sup>

<sup>1</sup> School of Computer Science & Technology, China University of Mining & Technology, Xuzhou, China

<sup>2</sup> School of Information Engineering, Yancheng Institute of Technology, Yancheng, China

E-mail: xp\_hua@163.com

Shifei Ding<sup>1,3</sup>

<sup>1</sup> School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China

<sup>3</sup> Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, China

E-mail: dingsf@cumt.edu.cn

**Abstract**—Support vector data description (SVDD) has become a very attractive kernel method due to its good results in many novelty detection problems. Training SVDD involves solving a constrained convex quadratic programming, which requires large memory and enormous amounts of training time for large-scale data set. In this paper, we analyze the possible changes of support vector set after new samples are added to training set according to the relationship between the Karush-Kuhn-Tucker (KKT) conditions of SVDD and the distribution of the training samples. Based on the analysis result, a novel algorithm for SVDD incremental learning is proposed. In this algorithm, the useless sample is discarded and useful information in training samples is accumulated. Experimental results indicate the effectiveness of the proposed algorithm.

**Index Terms**—support vector data description, incremental learning, Karush-Kuhn-Tucker condition

## I. INTRODUCTION

Support vector data description (SVDD), proposed by Tax and Duin [1,2], is a powerful kernel method for novelty detection or one-class classification, which is inspired by the support vector machine (SVM) [3] and gives the sphere boundary description of the target data points with minimum volume. For its better learning capacity and generalization performance, success of SVDD has recently been shown in various applications requiring novelty detection [4,5,6]. Extension of the original idea of SVDD to pattern denoising is also impressive [7].

There are still some difficulties associated with SVDD application which hampered its effectiveness, efficiency as well as general acceptability in engineering domain. These difficulties are mainly reflected on the two aspects. The first is the SVDD is not efficient for very large data set since it requires solving a quadratic programming (QP) problem in a number of coefficients equal to the number of training examples. The second difficulty is that almost all SVDD algorithms at hand are

not applicable on-line, that is, in cases where data are sequentially obtained and learning has to be done from the first data.

The incremental learning technique opens a novel way to overcome above difficulties. In the early stage, there have been some studies on SVM incremental learning algorithms [8,9,10]. The common feature of these incremental techniques is that they train an SVM incrementally on new data by discarding all previous data except support vector (SV) set. Therefore, they may only give approximate results. An exact solution to the problem of SVM incremental learning was found by Cauwenberghs and Poggio [11]. The key of this algorithm is to retain the Karush-Kuhn-Tucker (KKT) conditions on all previously seen data while adiabatically adding a new data point to the solution. Tax and Laskov extended this algorithm to SVDD and presented an online training of SVDD using a limited sample set [12]. This method sweeps through the data set, but instead of solving the problem by using the whole data samples as the working set, it keeps a portion of the data [13]. At each run, a new sample is added and the most irrelevant one is removed. However, the online SVDD training algorithm has some limitations when the number of training samples increases, since it reduces the size of the optimization problem but still solves it in a canonical manner [13].

The learning result of SVDD is SV set, which usually a small portion of training sample set. They can fully describe the property of the whole data set. By using the relation between sample set and KKT conditions we analyze the possible change of SV set after new samples are added to the training set. Based on the analysis result we present a novel incremental learning algorithm for SVDD training. The main objective of our research is to guarantee the learning accuracy while reducing space and time complexities.

The rest of the paper is organized as follows. Section II reviews the standard SVDD theory. Section III analyzes the process of the SVDD incremental learning. Section IV

describes our proposed algorithm. Section V presented experimental results on two data sets. Section VI gives some concluding remarks.

## II. SVDD

Given a target training set  $T = \{x_i \in R^d\}_{i=1}^l$ , SVDD aims at finding a minimum-volume sphere with center  $a$  and radius  $R$  in the feature space  $F$  such that all, or most of the target training samples are enclosed by the sphere, formulated as:

$$\begin{aligned} \min_R R^2 + C \sum_{i=1}^l \xi_i \quad (1) \\ \text{s.t. } \|\Phi(x_i) - a\|^2 \leq R^2 + \xi_i \quad (2) \\ \xi_i \geq 0, \forall i = 1, \dots, l \end{aligned}$$

where  $C$  is the penalty weight which gives the trade-off between volume of the sphere and the misclassification errors, and  $\xi_i$  is the slack variable.

Constraints (2) can be incorporated into (1) by using Lagrange multipliers. The Lagrangian formula is:

$$L = R^2 + C \sum_{i=1}^l \xi_i - \sum_{i=1}^l \alpha_i (R^2 + \xi_i - \|\Phi(x_i) - a\|^2) - \sum_{i=1}^l \beta_i \xi_i \quad (3)$$

where  $\alpha_i$  and  $\beta_i$  are Lagrange multipliers.

Setting  $\partial L / \partial R = 0$ ,  $\partial L / \partial a = 0$ , and  $\partial L / \partial \xi_i = 0$  can yield the results formulated as:

$$\sum_{i=1}^l \alpha_i = 1 \quad (4)$$

$$a = \sum_{i=1}^l \alpha_i \Phi(x_i) \quad (5)$$

$$C - \alpha_i - \beta_i = 0 \quad \forall i \quad (6)$$

Substituting the results (4), (5) and (6) back into  $L$  can yield the dual problem of (1) and (2), formulated as:

$$\max_{\alpha} \sum_{i=1}^l \alpha_i K(x_i, x_i) - \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j K(x_i, x_j) \quad (7)$$

$$\text{s.t. } \sum_{i=1}^l \alpha_i = 1, \quad (8)$$

$$0 \leq \alpha_i \leq C, \forall i = 1, \dots, l$$

where  $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$  is a kernel function.

After solving the dual problem above, an optimal solution  $\alpha^* = [\alpha_1, \alpha_2, \dots, \alpha_l]$  makes every  $x_i$  satisfy

KKT conditions formulated as [14]:

$$\alpha_i = 0 \Rightarrow d_i^2 \leq R^2 \quad (9)$$

$$0 < \alpha_i < C \Rightarrow d_i^2 = R^2 \quad (10)$$

$$\alpha_i = C \Rightarrow d_i^2 \geq R^2 \quad (11)$$

where  $d_i$  is the distance from sample point  $x_i$  to sphere center  $a$ .

According to the KKT conditions above, the target training data can be classified into three categories [15]: 1) the data points whose  $\alpha_i = 0$  are inside of the sphere; 2) the data points whose  $0 < \alpha_i < C$  are on the sphere boundary; and 3) the data points whose  $\alpha_i = C$  fall outside the sphere and have nonzero  $\xi_i$ . The data points with  $\alpha_i > 0$  are called support vectors (SVs). The expression for the center is given by:

$$a = \sum_{i=1}^{N_{SVs}} \alpha_i \Phi(x_i) \quad (12)$$

where  $N_{SVs}$  is the number of SVs. The radius  $R$  can be determined by taking any  $x_k$  on the sphere boundary and calculating the distance from its image to the center  $a$ , formulated as:

$$R = (K(x_k, x_k) - 2 \sum_{x_i \in SVs} \alpha_i K(x_i, x_k) + \sum_{x_i \in SVs} \sum_{x_j \in SVs} \alpha_i \alpha_j K(x_i, x_j))^{1/2} \quad (13)$$

To test an unseen data  $z$ , the distance to the center of the sphere must be calculated. It is accepted when this distance is smaller or equal than the radius  $R$  in  $F$ , formulated as:

$$f(z) = K(z, z) - 2 \sum_{x_i \in SVs} \alpha_i K(x_i, z) + \sum_{x_i \in SVs} \sum_{x_j \in SVs} \alpha_i \alpha_j K(x_i, x_j) \leq R^2 \quad (14)$$

## III. ANALYSIS OF THE SVDD INCREMENTAL LEARNING

The main task of incremental learning is to decide the ways of how to deal with the newly added training samples, and utilize the result of the previous training effectively to get better classification result and avoid training the same training sample set repeatedly. When penalty weight  $C$  and kernel function type are determined, SV set can fully describe the classification characters of the whole training set. So we focus on the effect of the newly added training set on the SV set. Next, we discuss several properties.

*Property 1:* If samples in the new training set satisfy KKT conditions, they will not change the previous SV set.

*Proof:* The Lagrange multiplier  $\alpha$  of every newly

added training sample is zero. Since all samples in the new training set satisfy KKT conditions, they must be inside of the sphere. So the same result can be produced utilizing previous training set or the union of previous and new training set as training samples. Hence the samples in the new training set will not change the previous SV set.

This property indicates that in a newly added training set the samples satisfying KKT conditions can be discarded during the incremental learning process of SVDD for their information has been included in the previous training set.

*Property 2:* If samples in the newly added training set violate KKT conditions, some of them will become new support vectors surely.

*Proof:* Suppose that all samples in the new training set will not become new support vectors. So the Lagrange multiplier  $\alpha$  of every newly added training sample is zero after training SVDD classifier with the union of previous and new training set. Thus the same result can be produced utilizing the previous training set or the union of previous and new training set as training samples. Further, it can be concluded that the samples in the new training set satisfy KKT conditions. However, this conclusion contradicts with the presupposition of property 2. Hence property 2 holds.

This property indicates that in the newly added training set the samples violating KKT conditions can not be discarded during the incremental learning process of SVDD for their information has not been included in the previous training set.

*Property 3:* If samples in the new training set violate KKT conditions, the non-SV in the previous training set will likely be converted to new SV.

*Proof:* This property can be proved by a special instance as shown in Fig. 1. A1-A5 are previous training samples, and B1-B3 are newly added training samples which obviously violating KKT conditions. A1 and A5 are previous support vectors attained by training a SVDD classifier on the previous training set. The new SV set made up of A2, A5, B1 and B3 can be produced after training a new SVDD classifier on the previous and newly added training set. It is obvious that the non-SV A2 is converted to a new SV during the new training process.

This property indicates that the non-SV samples in the previous training set can not be discarded wholly during the incremental learning process of SVDD (similar to the approximate algorithm [8]), when there exist samples violating KKT conditions in the newly added training set. In the course of experiments, we find that in the previous training set the non-SV near the sphere boundary are most likely to be new support vectors. These samples are formulated as:

$$R - \theta \leq f(z) \leq R, \quad \theta \in [0, R] \quad (15)$$

The parameter  $\theta$  is relative to the distribution of previous training set, and the loose distribution will make the value of  $\theta$  be high. In addition, with the incremental

learning gradually, the value of  $\theta$  will be low for more and more samples located near the previous SV set.

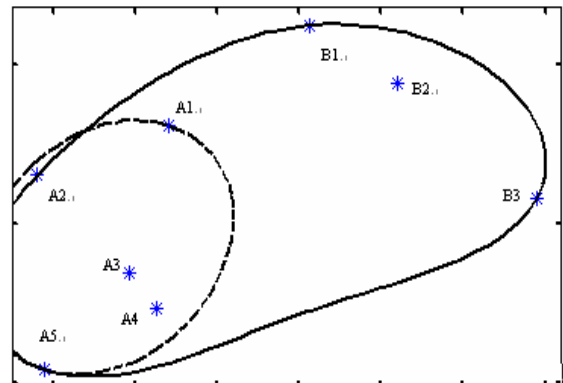


Figure 1. The possible change of SV set after training on incremental set.

#### IV. ALGORITHM

Based on the above analysis, we present a new algorithm for incremental learning of SVDD. During the incremental learning process of SVDD, the real training set should be made up of previous SV set, samples violating KKT conditions in the newly added training set and samples satisfying (15) in the previous training set. In this way, time and space complexities can be decreased greatly.

*Algorithm:*

*Input:* The previous training set  $X_0$ , the newly added training set  $X_1$

*Output:* SVDD classifier  $\Omega$  and the retained training set  $X_0$ .

*Step 1:* By training SVDD classifier  $\Omega$  on  $X_0$ , the previous training set is partitioned into SV set  $SV_0$  and non-SV set  $NSV_0$ .

*Step 2:* Verify that whether there is a training sample in  $X_1$  violating KKT conditions of  $\Omega$ . If there isn't,  $\Omega$  and  $X_0$  will be the result of the incremental learning. Terminate. Otherwise, choose the samples violating KKT conditions of  $\Omega$  and represent them as  $X_1^V$ .

*Step 3:* Choose samples satisfying (15) from  $NSV_0$  and represent them as  $NSV_0^S$ .

*Step 4:* Let  $X_0$  be the union of  $SV_0$ ,  $NSV_0^S$  and  $X_1^V$  and train the SVDD classifier  $\Omega$  on  $X_0$ .

#### V. EXPERIMENTS AND RESULTS

In order to evaluate the performance of the incremental learning algorithm presented in this paper, several experiments are conducted on two different datasets to compare our method with the traditional canonical [3] and online training methods [12]. The experiments are implemented on a Pentium 1.8-G computer with 1G RAM of memory and all the algorithms are programmed in Matlab 7.0. In our method, the parameter  $\theta$  is taken as  $0.2R$ , and each training data set is divided into several parts in order to increase training speed.

A. Banana Data Set

Banana data set is a 2-dimensional dataset with a banana shaped distribution. The data is uniformly distributed along the bananas and superimposed with a normal distribution with standard deviation in all directions. In this experiment, we compare the three training methods using four banana data sets with

different number of training samples. Gaussian kernel  $K(x, y) = \exp(-\|x-y\|^2 / 2\sigma^2)$  is chosen as kernel function, and the optimal parameter values of  $C$  and  $\sigma$  are founded using 10-fold cross validation method.

From TABLE I, it can be seen that the proposed SVDD training technique runs faster than both online and canonical algorithms and its training time is linear with the data set size. As expected, both online and incremental SVDD training methods are considerably faster than the canonical training of the classifier, and the training time of a canonical SVDD for 2000 training samples is not available since it takes hours to finish the experiment. Because of the fact that it retains more unnecessary support vectors, the online SVDD runs slower than the proposed method for larger data sets.

TABLE I.  
THE COMPARISON OF TRAINING TIME IN BANANA DATA SET

Size of Training Set (pts)	Incremental SVDD		Online SVDD		Canonical SVDD	
	(C, $\sigma^2$ )	CPU Sec.	(C, $\sigma^2$ )	CPU Sec.	(C, $\sigma^2$ )	CPU Sec.
100	(10,3.5)	0.23s	(1,4)	0.34s	(1,3.5)	0.34s
200	(5,3)	0.73s	(0.05,3)	0.87s	(0.1,2.5)	5.78s
500	(4,4)	1.32s	(0.02,3)	2.53s	(0.05,3)	113.53s
1000	(5,4)	3.44s	(0.01,5)	5.52s	(0.01,4)	1232.87s

From TABLE II, it can be observed that our method has less support vectors than the other two methods and both online and canonical SVDD training algorithms increase the number of support vectors as the size of the data set increases while our method keeps almost a constant number of support vectors. This can be interpreted as discarding useless training samples in previous training sets reasonably. Since the classification speed of a SVDD classifier is in proportion to the number

of support vectors, the SVDD classifier trained by the proposed technique can predict an unknown sample more quickly. In addition, by increasing the number of training samples our algorithm requires less memory than both online and canonical algorithms. This makes the proposed algorithm very suitable for applications in which the number of training sample increase by time, i.e. in the case of growing data sets.

TABLE II.  
THE COMPARISON OF THE NUMBER OF SUPPORT VECTORS IN BANANA DATA SET

Size of Training Set (pts)	Incremental SVDD		Online SVDD		Canonical SVDD	
	(C, $\sigma^2$ )	Number of SVs	(C, $\sigma^2$ )	Number of SVs	(C, $\sigma^2$ )	Number of SVs
100	(10,3.5)	14	(1,4)	14	(1,3.5)	14
200	(5,3)	21	(0.05,3)	30	(0.1,2.5)	31
500	(4,4)	16	(0.02,3)	55	(0.05,3)	58
1000	(5,4)	15	(0.01,5)	103	(0.01,4)	105

In Fig. 2 the classification boundaries of the three SVDD training algorithms on four different banana datasets with different number of training samples are shown. From Fig. 2 it can be observed that the decision

boundaries of the classifier trained using the incremental algorithm (solid curve) is objectively more accurate than those trained by online (dashed curve) and canonical (dotted curve) methods.

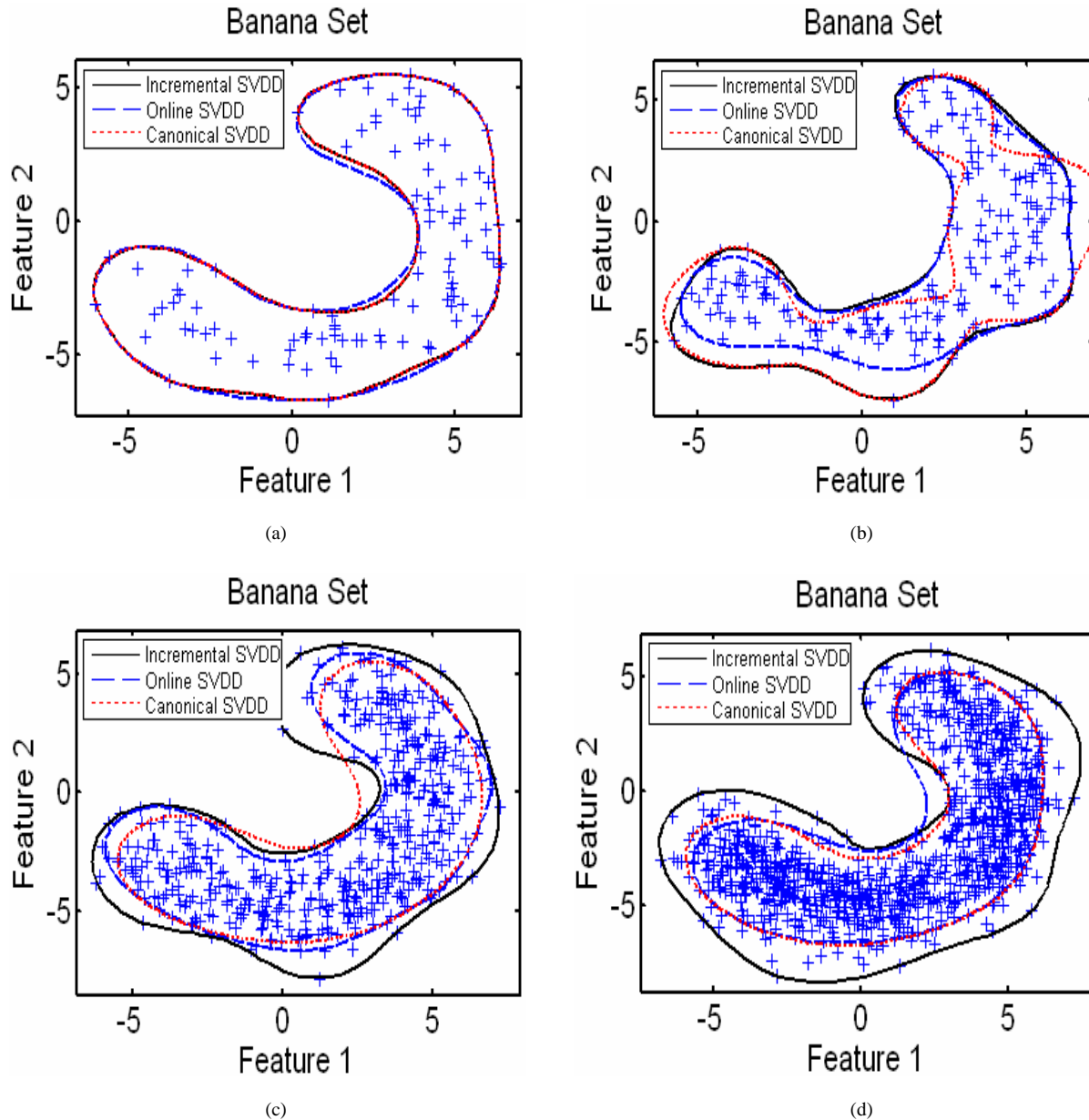


Figure 2. Comparison of classification boundaries of incremental with online and canonical SVDD: (a) 100 training samples; (b) 200 training samples; (c) 500 training samples; (d) 1000 training samples.

### B. MNIST Data Set

The second set of experiments is performed on the MNIST data set [16], which contains  $28 \times 28$  grey-level images of the digits 0, 1, ..., 9. In this experiment, we treat the digit one images as normal patterns and all others as outliers. Sample images are shown in Fig. 3. A total of 6,742 digit one images are used to form the training set, while the test set has 10,000 images, with 1,135 of them belonging to the digit one. Again, we use the Gaussian kernel as kernel function, and the optimal

parameter values are found using 10-fold cross validation method.

Besides CPU running time and the number of support vectors, two new criteria are used to compare the proposed algorithm with online and canonical SVDD. They are TP and FP based on the ROC (receiver operating characteristic) graph which plots the true positive (TP) rate on the Y-axis and the false positive (FP) rate on the X-axis [17]. TP and FP are defined by

$$TP = \frac{\text{positives correctly classified}}{\text{total positives}},$$

$$FP = \frac{\text{negatives incorrectly classified}}{\text{total negatives}},$$

Respectively. Here, outliers are treated as negatives while target patterns as positives.

From TABLE III, it can be seen that the incremental SVDD proposed in this paper is more accurate than the

online and canonical SVDD in terms of TP and FP especially for large-scale training set. The training speed of our algorithm is faster than the other two methods, especially on large data set. With 2,000 or more training samples, the canonical SVDD cannot be run on the machine since it takes hours to finish the experiment. Finally, other trends as discussed in experiments in banana data set can also be observed here.

TABLE III.  
EXPERIMENTAL RESULTS ON THE MNIST DATA SET

Size of Training Set (pts)	Algorithm	Number of SVs	CPU Sec.	TP (%)	FP (%)
100	Incremental SVDD	17	0.37	84.84	0.04
	Online SVDD	17	0.39	84.84	0.04
	Canonical SVDD	17	0.39	84.84	0.04
300	Incremental SVDD	25	1.56	94.18	1.24
	Online SVDD	25	1.93	94.18	1.24
	Canonical SVDD	38	11.82	94.12	1.33
600	Incremental SVDD	27	2.96	95.68	2.00
	Online SVDD	27	7.64	94.57	2.30
	Canonical SVDD	65	108.78	94.68	2.15
1000	Incremental SVDD	27	8.93	96.38	1.81
	Online SVDD	38	29.70	93.28	1.75
	Canonical SVDD	134	1572.67	95.76	1.78
2000	Incremental SVDD	36	13.9	98.76	3.71
	Online SVDD	49	93.07	98.59	4.21
3000	Incremental SVDD	33	40.12	99.12	5.77
	Online SVDD	66	210.32	98.94	7.95
6000	Incremental SVDD	38	103.23	99.47	6.81
	Online SVDD	114	1031.95	98.99	13.69

VI. CONCLUSIONS

SVDD training is a quadratic programming (QP) problem. By increasing the number of training samples, solving this QP problem becomes intractable both in terms of memory requirements and speed. In this paper, we discuss the relation between samples and KKT condition of SVDD classifier. Based on the relation we analysis the possible change of support vectors after incremental samples are added to training

set. Furthermore, an incremental algorithm based on above analysis is presented. Experiments prove that our algorithm can condense the training sample set effectively while classification precision and generalization ability are guaranteed.

ACKNOWLEDGEMENT

This work is supported by the Basic Research Program (Natural Science Foundation) of Jiangsu Province of China (No. BK2009093), the National Nature Science

Foundation of China (No. 41074003), the Opening Foundation of Beijing Key Lab of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, and the Special Foundation for Researchers of Jiangsu Province.

#### REFERENCES

- [1] D. Tax and R. Duin, "Support vector domain description," *Pattern Recognit. Lett.*, vol. 20, pp. 1191-1199, November 1999.
- [2] D. Tax and R. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, pp. 45-66, January 2004.
- [3] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, USA: Springer Verlag, 1999.
- [4] A. Banerjee, P. Burlina, and C. Diehl, "A support vector method for anomaly detection in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pp. 2282-2291, August 2006.
- [5] Y. H. Liu, S. H. Lin, Y. L. Hsueh, and M. J. Lee, "Automatic target defect identification for TFT-LCD array process inspection using kernel FCM based fuzzy SVDD ensemble," *Expert Syst. Applicat. Int. J.*, vol. 36, pp. 1978-1998, March 2009.
- [6] L. Nanni, "Machine learning algorithms for T-cell epitopes prediction," *Neurocomputing*, vol. 69, pp. 866-868, March 2006.
- [7] J. Park, D. Kang, J. Kim, J. T. Kwok, and I. W. Tsang, "SVDD-based pattern denoising," *Neural Comput.*, vol. 19, pp. 1919-1938, July 2007.
- [8] N. A. Syed, H. Liu and K. Sung, "Incremental learning with support vector machines", *International Joint Conference on Artificial Intelligence*, 1999, pp. 876-892.
- [9] P. Mitra, C. A. Murthy and S. K. Pal, "Data condensation in large databases by incremental learning with support vector machines", *International Conference on Pattern Recognition*, 2000.
- [10] C. Domeniconi and D. Gunopulos, "Incremental support vector machine construction", *Proceedings of IEEE International Conference on Data Mining*, 2001, pp.589-592.
- [11] G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," *Advances in Neural Information Processing Systems*, vol. 13, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. Cambridge: MIT Press, 2001, pp. 409-415.
- [12] D. M. J. Tax and P. Laskov, "Online SVM learning: from classification to data description and back," *In C. et al. Molina, editor, Proc. NNSP*, 2003, pp. 499-508.
- [13] A. Tavakkoli, Mircea Nicolescu, Monica Nicolescu, and G. Bebis, "Incremental SVDD training: improving efficiency of background modeling in videos", *Signal and Image Proceeding (SIP08)*, August 2008.
- [14] Xu Tu, Luo Yu, and He Da-ke, "SMO training algorithm for hyper-sphere one-class SVM", *Computer Science*, vol. 35, pp.178-180, June 2008.
- [15] Y. H. Liu, S. H. Lin, Y. L. Hsueh, and M. J. Lee, "Automatic target defect identification for TFT-LCD array process inspection using kernel FCM based fuzzy SVDD ensemble," *Expert Syst. Applicat. Int. J.*, vol. 36, pp. 1978-1998, March 2009.
- [16] M. Markau and S. Singh, "Novelty detection: A review, Part I: Statistical approaches," *Signal Processing*, vol. 83, pp. 2481-2497, December 2003.
- [17] A. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern*

*Recognition*, vol. 30, pp. 1145-1159, July 1997.



**Xiaopeng Hua** was born in Jiangsu Province, China, in 1975. He received the B.S. and M.S. degree, both in Computer Science and Technology, from Nanjing University of Science and Technology in 1998 and 2006 respectively. From 1998 to 2009, he served as a lecture with School of Information Engineering at Yancheng Institute of Technology. He is currently pursuing the Ph.D. degree with School of Computer Science and Technology at China University of Mining and Technology. His research interests include support vector machine, pattern classification, fuzzy logic and clustering.



**Shifei Ding** received his bachelor's degree and master's degree from Qufu Normal University in 1987 and 1998 respectively. He received his Ph.D. degree from Shandong University of Science and Technology in 2004. He received postdoctoral degree from Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences.

He is a professor and Ph.D. supervisor at China University of Mining and Technology. His research interests include intelligent information processing, pattern recognition, machine learning, data mining, and granular computing et al. He has published 3 books, and more than 80 research papers in journals and international conferences.

He is a senior member of China Computer Federation (CCF), and China Association for Artificial Intelligence (CAAI). He is a member of professional committee of distributed intelligence and knowledge engineering, CAAI, professional committee of machine learning, CAAI, and professional committee of rough set and soft computing, CAAI. He acts as an editor for *Journal of Convergence Information Technology (JCIT)*, *International Journal of Digital Content Technology and its Applications (JDCTA)*. Meanwhile, he is a reviewer for *Journal of Information Science (JIS)*, *Information Sciences (INS)*, *Computational Statistics and Data Analysis (CSTA)*, *IEEE Transactions on Fuzzy Systems (IEEE TFS)*, *Applied Soft Computing (ASOC)*, *Computational Statistics and Data Analysis (CSDA)*, *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)* et al.



(a)



(b)

Figure 3. Examples of handwritten digits from MNIST data set: (a) digit 1 images; (b) outlier images.