Increasing Efficiency of SVM by Adaptively Penalizing Outliers

Yiqiang Zhan[1,2,3] and Dinggang Shen[2,3]

[1] Dept. of Computer Science, The Johns Hopkins University, Baltimore, MD
[2] Center for Computer-Integrated Surgical Systems and Technology, The Johns Hopkins University, Baltimore, MD
[3] Section of Biomedical Image Analysis, Dept. of Radiology, University of Pennsylvania, Philadelphia, PA
yzhan@cs.jhu.edu, Dinggang.Shen@uphs.upenn.edu

Abstract. In this paper, a novel training method is proposed to increase the classification efficiency of support vector machine (SVM). The efficiency of the SVM is determined by the number of support vectors, which is usually large for representing a highly convoluted separation hypersurface. We noted that the separation hypersurface is made unnecessarily over-convoluted around extreme outliers, which dominate the objective function of SVM. To suppress the domination from extreme outliers and thus relatively simplify the shape of separation hypersurface, we propose a method of adaptively penalizing the outliers in the objective function. Since our reformulated objective function has the similar format of the standard SVM, the idea of the existing SVM training algorithms is borrowed for training the proposed SVM. Our proposed method has been tested on the UCI machine learning repository, as well as a real clinical problem, i.e., tissue classification in prostate ultrasound images. Experimental results show that our method is able to dramatically increase the classification efficiency of the SVM, without losing its generalization ability.

1 Introduction

Support vector machine (SVM), proposed by Vapnik in 1995 [1], is a new generation of learning systems, based on statistical learning theory. Considering a two-class classification problem with m labeled training samples,

$$\{(\bar{x}_i, y_i) \mid \bar{x}_i \in R^n, y_i \in \{-1,1\}, i = 1 \cdots m\},$$

SVM is able to generate a hypersurface that has maximum margin to separate these two classes. During the applications, a testing sample $\bar{x}$ is classified by calculating its distance to the hypersurface:

$$d(\bar{x}) = \sum_{i=1}^{m} \alpha_i y_i K(\bar{x}_i, \bar{x}) + b \qquad (1)$$

where α_i and b are the parameters determined by SVM's learning algorithm, and $K(\bar{x}_i, \bar{x})$ is the kernel function. Samples $\bar{x}_i$ with nonzero parameters α_i are called "support vectors".

A. Rangarajan et al. (Eds.): EMMCVPR 2005, LNCS 3757, pp. 539 – 551, 2005.

SVM has several striking properties. *First*, SVM is developed based on the idea of structural risk minimization [1]. It can achieve high generalization ability by minimizing the Vapnik-Chervonenkis dimension. *Second*, by using the kernel trick [2], the samples are implicitly mapped to a higher dimensional space. Therefore, SVM can generate a convoluted hypersurface to non-linearly separate different classes. *Finally*, the training procedure of SVM can be eventually formulated as a constraint quadratic optimization problem, which has a unique global minimum.

Accordingly, SVM shows superior performances in pattern recognition problems and has drawn considerable attentions in various research areas [3-8]. However, while confronting large data classification problem, SVM usually needs a huge number of support vectors to parameterize the separation hypersurface. Since it is computationally expensive to calculate the decision function with many non-zero parameters α_i in Eq. (1), SVM exhibits substantially slower classification speed than that of neural network [9]. This disadvantage unavoidably limits the capability of SVM in the applications that require a massive number of classifications [5] or real-time classification [11].

In this paper, we will propose a novel training method to increase the classification efficiency of SVM. The basic idea of our method is to prevent the separation hypersurface from being locally over-convoluted, usually incurred by extreme outliers in the training set. To achieve this goal, we reformulate the objective function of the standard soft-margin SVM by designing an adaptive penalty term to outliers. Therefore, the objective function will be no longer dominated by extreme outliers and the separation surface can be simplified without losing its generalization ability. We will further show that the reformulated objective function can be eventually transformed to a quadratic optimization problem with adaptive constraints, which has similar format as the dual problem of the standard soft-margin SVM. In this way, our training method can be easily implemented in an iterative framework, which can be embedded by any existing SVM training methods.

The remaining of this paper is organized as following. In Section 2, we will first analyze the problem in details, and then reformulate SVM by an adaptive penalty term to outliers. The training method of reformulated SVM will also be provided. Section 3 will present the experimental results of our method on the UCI machine learning repository, as well as a real clinical problem, i.e., tissue classification in a set of prostate ultrasound images. This paper concludes in Section 4.

2 Methods

2.1 Problem Description

As indicated in Eq. (1), the computational cost of SVM is directly related to the number of the support vectors, i.e. training samples with non-zero parameters α_i. According to their relative positions to the separation hypersurface, support vectors can be categorized into two types. The first type of support vectors are the training samples that exactly locate on the margins of the separation hypersurface, i.e., $d(\bar{x}_i) = \pm 1$, such as red circles/crosses shown in Fig 1. The second type of support vectors are the training samples that locate beyond their corresponding margins, i.e., $y_i d(\bar{x}_i) < 1$, such as blue and green circles/crosses shown in Fig 1. For a SVM, the second type of

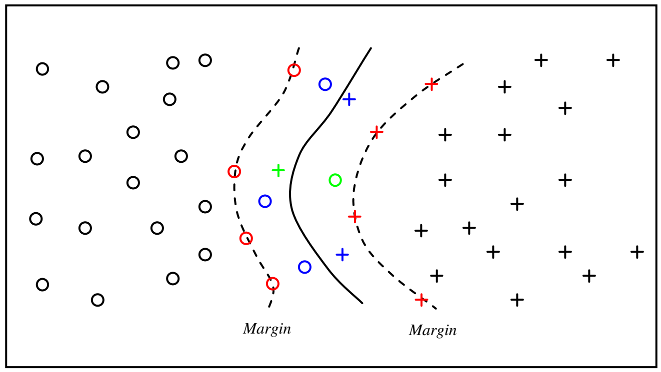


Fig. 1. Schematic explanation of the separation hypersurface (solid curves), margins (dashed curves), and support vectors of SVM (colored circles/crosses). The positive and the negative training samples are indicated by circles and crosses, respectively.

support vectors are regarded as misclassified samples, although some of them still locate at the correct side of the hypersurface (shown as blue circles/crosses).

SVM usually has a huge number of support vectors, when the distributions of the positive and the negative training samples from a large dataset highly overlap with each other. This unfavorable situation can be contributed to two reasons: (1) a large number of the first-type support vectors are needed to construct a highly convoluted hypersurface, in order to separate two classes; (2) even the highly convoluted separation hypersurface has been constructed, a lot of confounding samples will be misclassified, thus selected as the second type of support vectors.

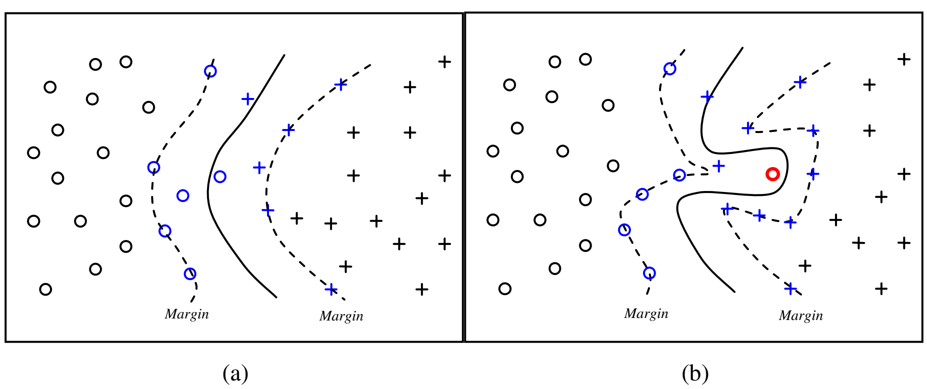


Fig. 2. Schematic explanation of the over-convoluted separation hypersurface incurred by an outlier. The solid and dashed curves denote the separation hypersurface and margins, respectively. Circles and crosses denote the positive and the negative training samples, respectively. The blue ones denote the support vectors. The red circle in (b) is an outlier that incurs the hypersurface over-convoluted. Notably, except the red circle in (b), other samples in (a) and (b) are identical.

Some support vectors might be redundant to parameterize the separation hypersurface. Based on this hypothesis, researchers have proposed efficient SVM training methods [11][13]. Compared to [13], the method proposed by Osuna *et al* in [11] is more feasible, and it offered a principle for controlling the accuracy of approximation. This method approximates the separation hypersurface with a subset of the support vectors by using a Support Vector Regression Machine (SVRM). If the separation hypersurface is relatively simple, Osuan's method is quite effective to reduce the number of support vectors without system degradation. However, in many large dataset classification problems, SVM usually generates a highly convoluted separation hypersurface, which is difficult to be parameterized by a small number of support vectors as Osuan's method did. Therefore, the only way for decreasing the number of support vectors is to simplify the over-convoluted separation hypersurface.

It is widely accepted that the convoluted hypersurface of SVM is critical for nonlinear separation of classes, which might overlap with each other in the original feature space. However, in certain cases, the hypersurface generated by SVM is unnecessarily over-convoluted in some local regions without increasing the generalization ability of the SVM. Fig 2 presents a toy problem to illustrate those cases. In Fig 2(a), the separation hypersurface of the SVM is relatively simple and it has 12 support vectors (denoted by blue crosses or circles). The distribution of the training samples in Fig 2(b) is almost the same as that of Fig 2(a) except an additional positive sample (denoted by the red circle). However, the separation hypersurface in Fig 2 (b) became much more convoluted, in order to satisfy this additional sample, and the trained SVM has 16 support vectors. Notably, since this additional training sample locates in an isolated region that is far from samples of the same class, it might be an outlier produced by noise or error. Therefore, the over-convoluted hypersurface, used to satisfy this sample, will decrease the generalization ability of SVM, but increase its computational cost. Obviously, this unfavorable situation should be avoided, in order to increase the classification efficiency and generalization of the SVM. In the next section, we will investigate this problem in detail, and finally prevent it by reformulating a new objective function for SVM.

2.2 Reformulation of Objective Function in SVM

It is necessary to briefly introduce the objective function of SVM, before investigating the reason for over-convoluted separation hypersurface in some classification cases. According to the statistical learning theory [1], SVM tries to generate a separation hyperplane, $\vec{w}\cdot\vec{x}+b=0$, which has the maximum generalization ability. Here, $\vec{w}$ is the normal of the hyperplane, and b is the distance from the hyperplane to the origin. Given m labeled training samples, i.e. $\{(\vec{x}_i, y_i) \mid \vec{x}_i \in R^n, y_i \in \{-1,1\}, i = 1\cdots m\}$, the training of SVM can be formulated as solving a quadratic optimal problem:

$$\begin{aligned}&\min_{\vec{w},b,\xi_i} \frac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{m}\xi_i\\ &s.t.\ y_i(\vec{w}\cdot\phi(\vec{x}_i)+b) \geq 1-\xi_i\\ &\quad \xi_i \geq 0\end{aligned} \tag{2}$$

Here, $\|\cdot\|$ is the norm of a vector, and $\phi(\cdot)$ maps samples into a higher dimensional space and can be implicitly implemented by the kernel trick [2].

In the objective function of Eq. (2), the first term $\|\vec{w}\|$ measures the inverse of the margin distance that should be minimized to obtain the minimum structural risk [2]. The second term is a penalty term consisting of a number of non-negative slack variables ξ_i, used to construct a soft margin hyperplane [12]. By using the relaxed separation constraints $y_i(\vec{w}\cdot\phi(\vec{x}_i)+b)\geq 1-\xi_i$, some training samples are allowed to locate beyond their corresponding margins, i.e. $y_i(\vec{w}\cdot\phi(\vec{x}_i)+b)<1$. The linear summation of all slack variables ξ_i is constrained by the second term in the objective function, in order to avoid the trivial solution that all ξ_i take large values.

According to Eq. (2), it is not difficult to understand the reason why the case described in Fig 2 happens. The separation hypersurface in Fig 2(b) has to be convoluted in order to satisfy that additional red positive sample; otherwise, the corresponding ξ_i of that additional sample will be very large, thus dominating the objective function. However, as the additional sample locates in an isolated region far from samples of the same class, it is an outlier that might be generated by noise or error. Therefore, the hypersurface convoluted around the additional sample is unnecessary, and it will only decrease the generalization of the trained SVM. To solve this problem, we introduce a non-linear penalty term, instead of the linear penalty term in Eq. (2) that makes the effect of outliers overwhelming over the whole objective function. The objective function of SVM is reformulated as following:

$$\begin{aligned}&\min_{\vec{w},b,\xi_i}\frac{1}{2}\|\vec{w}\|^2+C\sum_{i=1}^{m}erf\ (\xi_i;\sigma)\\&s.t.\ y_i(\vec{w}\cdot\phi(\vec{x}_i)+b)\geq 1-\xi_i\quad \xi_i\geq 0\end{aligned}\tag{3}$$

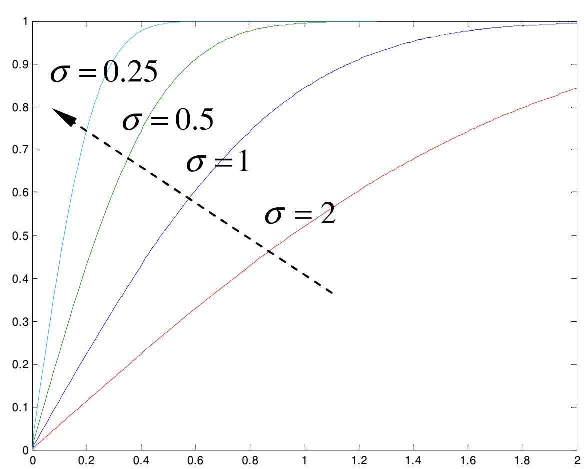


Fig. 3. A nonlinear error function for suppressing the slack variables with large value, thus adaptively penalizing outliers. The curves of different colors denote the error functions *with respect to* different parameter σ used. The dashed arrow indicates the decrease of σ with the progress of iterative training, i.e., transferring linear penalty to nonlinearly-mapped penalty.

where $erf(\xi;\sigma)$ is a nonlinear error function, defined by $erf(\xi;\sigma) = \frac{2}{\sqrt{\pi}\sigma}\int_0^{\xi} e^{-z^2/\sigma^2}dz$, to adaptively penalize outliers. As indicated by the function plot in Fig 3, the error function *erf* will suppress the slack variables when they are large. In this way, the objective function will be no longer dominated by the large slack variables, and thus the resulting separation hypersurface will not be over-convoluted around extreme outliers. On the other hand, if there are a considerable number of same-class samples clustering in an isolated region that is distant from other samples of this class, the generated hypersurface could still be convoluted to satisfy these samples, in order to decrease the total error for these samples. In this way, the generalization ability of the SVM is preserved.

2.3 Training of the Reformulated SVM

In this section, we will discuss the training algorithm for the reformulated SVM. Similarly, the Lagrangian theory is employed here to solve the reformulated constrained quadratic problem. After introducing Lagrangian multipliers α_i and η_i, we obtain the primary Lagrangian of Eq. (3) as:

$$L(\vec{w}, b, \xi_i) = \frac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{m} erf(\xi_i) - \sum_{i=1}^{m}\left[\alpha_i\left(y_i\left(\vec{w}\cdot\phi(\vec{x}_i) + b\right) - (1-\xi_i)\right) + \eta_i\xi_i\right] \quad (4)$$

According to Kuhn-Tucker condition, we can express Eq. (4) as a dual problem given next, by differentiating Eq. (4) with respect to the primary variables $\vec{w}$ and b, setting the derivatives as zero and resubmitting the relations obtained by these equations.

$$\begin{aligned} &\min_{\alpha_i,\xi_i} \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j - \sum_{i=1}^{m}\alpha_i + C\sum_{i=1}^{m}\left[\xi_i erf'(\xi_i;\sigma) - erf(\xi_i;\sigma)\right] \\ &s.t. \quad 0 \le \alpha_i \le C \cdot erf'(\xi_i;\sigma) \quad \xi_i \ge 0 \end{aligned} \quad (5)$$

where $erf'(\xi_i)$ is the derivative of the nonlinear error function, i.e., a Gaussian function with the standard deviation σ.

Notably, compared to the dual problem expressed for the standard SVM, the Lagrangian multiplier α_i in Eq. (5) is no longer constrained by a global constant C, but $C \cdot erf'(\xi_i;\sigma)$, which is adaptive to each sample according to its corresponding ξ_i. Since $erf'(\xi_i;\sigma)$ is actually a Gaussian function, the Lagrangian multiplier α_i of a training sample with large slack variable ξ_i is actually restricted by a very low upper bound. Therefore, this sample has very little contribution in constructing the separation hypersurface, even if it is selected as a support vector. Actually, the reformulated SVM offers a soft selection mechanism to adaptively determine the importance of different training samples, thus the effect of the outliers is suppressed. On the other hand, our method can be interpreted as an algorithm of adaptively and softly selecting samples for training, which is significantly different from the method proposed by Lee *et al* [14], which randomly selects a subset of training samples in order to speed up SVM.

Since the objective function in Eq. (5) has a very similar format with respect to the standard SVM, we can design an iterative framework to train the reformulated SVM, by borrowing any existing SVM training methods. *First*, α_i are optimized using a similar training method for the standard SVM, except using the adaptive constraints $0 \le \alpha_i \le C \cdot erf'(\xi_i;\sigma)$. *Then*, ξ_i can be calculated by the following equation,

$$\xi_i = \sum_{i=1}^{m} \alpha_i y_i K(\vec{x}_i, \vec{x}) + b - 1 \cdot \tag{6}$$

In the initial iterations, the parameter σ is set to be a large value, thus the error function performs as a linear slack term used in the standard SVM (c.f. Fig 3). With progress of the training, the parameter σ becomes smaller and smaller, thereby the nonlinear error function starts to suppress the large slack variables more. After the training procedure converges, we thus generate an optimal separation hypersurface for the reformulated SVM. *Finally*, Osuna's method [11], which employs the SVRM to approximate the hypersurface by a subset of support vectors, is employed to further decrease the number of support vectors and thus increase the classification efficiency of the SVM.

3 Experiments

To validate the effectiveness of our method, we applied it to the UCI Machine Learning Repository, as well as a real clinical problem, i.e., tissue classification in the prostate ultrasound images. The experimental results are presented next.

3.1 Experiments on UCI Machine Learning Repository

UCI Machine Learning Repository contains a set of data that is used by the machine learning community for the empirical analysis of machine learning algorithms. From the repository, we select the datasets "Adult" (45222 samples, 14 features), "Breast Caner" (687 samples, 10 features), "Ionosphere" (351 samples, 34 features), and "Monks" (432 samples, 6 features), to test our method.

For the datasets of "Breast Cancer", "Ionosphere" and "Monks", we randomly divided each dataset into several groups, with each group having 50 samples. In the training stage, one group is left out as testing samples, and other remaining groups are used as training samples. This leave-one-group-out cross validation is repeated. The averages on the correct classification rate and the number of support vectors from all tests are reported.

For the "Adult" dataset, we use the standard training and testing sets, which have 30162 samples and 15606 samples, respectively.

Table 1 summarizes the average number of support vectors and the average correct classification rate, obtained by the standard SVM, Osuna's method and our method, respectively. Compared to the standard SVM, our method is able to dramatically reduce the number of support vectors, without losing the generalization ability of the classifier. In some cases, the classification rate is even higher, indicating the increased generalization of our method. Compared to Osuna's method, our method is able to generate more efficient SVM with similar classification rate, except for the "Breast Cancer" dataset that has a relatively simple distribution of samples.

Table 1. Comparison on the classification performances obtained by standard SVM, Osuna's method, and our method. SV below denotes 'support vector'.

	Standard SVM		**Osuna's Method**		**Our Method**	
Problem	# SV	correct rate	# SV	correct rate	# SV	correct rate
Adult	9996	85.80%	1000	84.91%	338	85.38%
Breast	50.8	96.46%	6.8	96.92%	7.1	97.25%
Iono-sphere	161.3	93.43%	160.4	93.43%	37.5	94.28%
Monks	96.8	97.51%	53.9	97.50%	25.9	97.00%

3.2 Experiments on Tissue Classifications in Prostate Ultrasound Images

In our study of 3D prostate segmentation from ultrasound images [5], SVM is used for texture-based tissue classification. The input of SVM is a set of texture features extracted by the Gabor filter bank [16], and the output is a soft label denoting the likelihood of the voxel belonging to the prostate. In this way, prostate tissues are differentiated from the surrounding tissues. In this study, the computational cost of SVM for tissue classification is a particularly critical problem to be concerned, as the tissue classification is operated for lots of times (i.e., 10^6) in the segmentation stage and also the real-time segmentation is usually required for clinical applications. Therefore, the training method proposed in this paper is applied to speeding up the SVM for tissue classification.

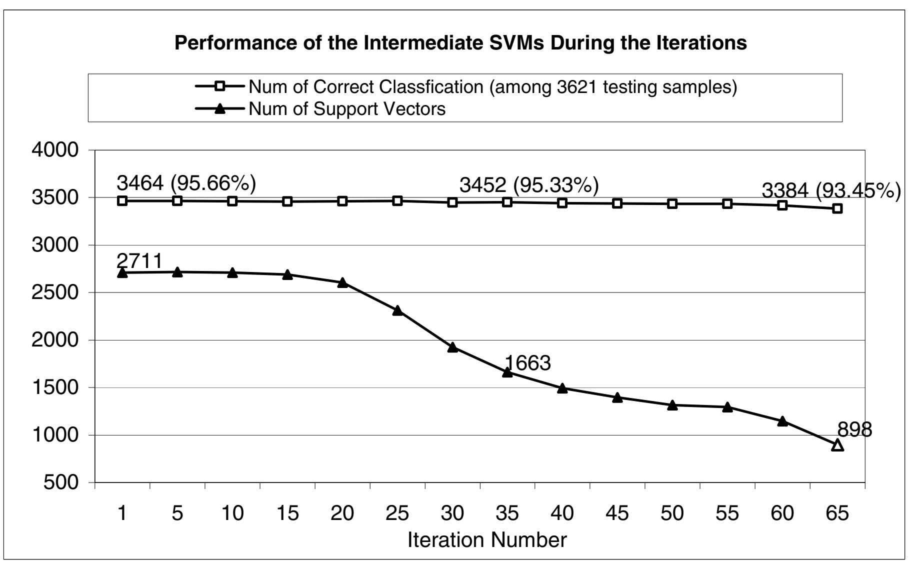
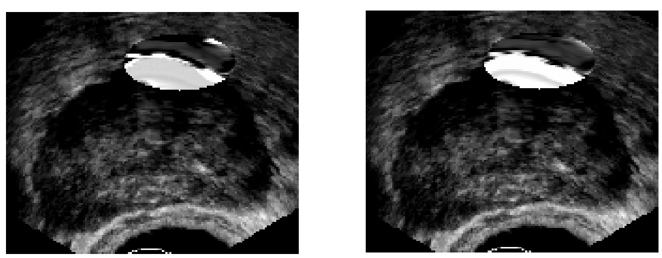


Fig. 4. Performance of intermediately trained SVMs, by different numbers of iteration

In preparing the experimental dataset, we first randomly select prostate and non-prostate samples from six manually labeled ultrasound images, in which 3621 samples from one ultrasound image are used as testing samples and 18105 samples from other five images are used as training samples. Each sample has 10 texture features, extracted by Gabor filters.

The SVM is used for tissue classification, and it is trained by the proposed iterative training method. After each iteration, we record the number of support vectors used and the correct classification of the intermediate SVM; the results from all possible numbers of iteration are provided in Fig 4. As shown in Fig 4, the number of support vectors is reduced quickly with the increase of iterations, while the classification rate keeps similar. The finally trained SVM has 898 support vectors, which is only 33.1% of those of the original SVM (2711); but its classification rate still reaches 93.45%. Compared to 95.66% classification rate achieved by the original SVM, the loss of classification rate is relatively trivial, thereby it will not affect the performance of our model-based segmentation algorithm [5]. Notably, the intermediate SVM can be selected as a classifier to satisfy different classification requirements, i.e., choosing the intermediate SVM generated in 35th iteration, with 1663 support vectors, to obtain a more accurate classification.

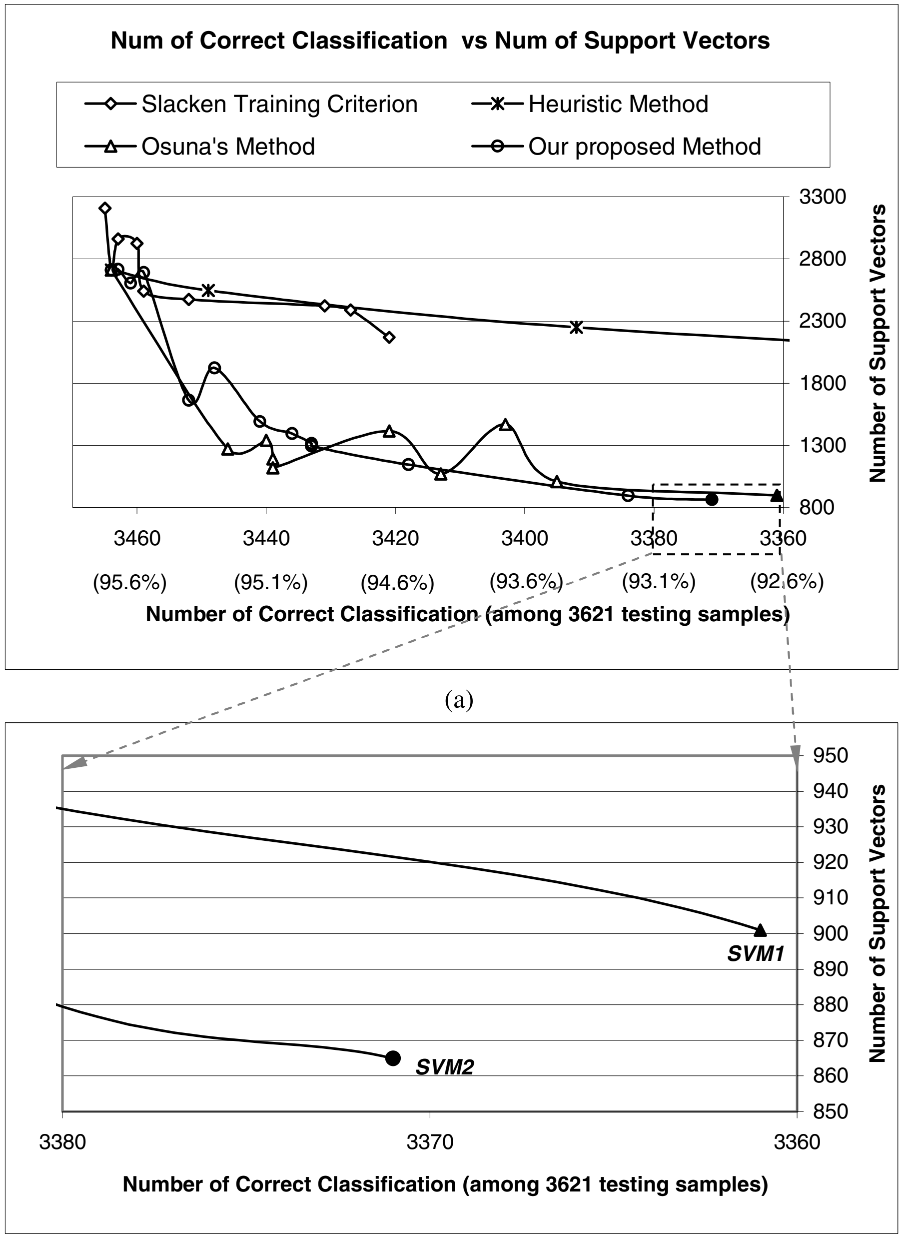


(a) (b)

Fig. 5. Comparison on tissue classification results obtained by (a) the original SVM with 2711 support vectors, and (b) our trained SVM with 898 support vectors. The tissue classification results are shown only in the regions surrounded by dashed ellipsoids.

To further validate the performance of our trained SVM in tissue classification, the SVM with 898 support vectors (denoted by the white triangle in Fig 4) is applied to a real ultrasound image for tissue classification. By comparing results in Fig 5(a) and Fig 5(b), the result of our trained SVM in Fig 5(b) is not inferior to that of the original SVM with 2711 support vectors in Fig 5(a), in terms of differentiating prostate tissues from the surrounding ones.

We further compare the performances of SVMs generated by different training methods. Four methods are implemented for comparison: (1) a method of slackening the training criterion by decreasing the linear penalty factor [2]; (2) a heuristic method, which assumes the training samples distributing in a multi-variant Gaussian way, then excludes the "outliers" distant from the respective distribution centers, and finally trains a SVM only by the remaining samples; (3) Osuna's method [11]; (4) our proposed method. The performances of these four methods are evaluated in Fig 6 (a),



(a)

(b)

Fig. 6. (a) Comparison on the performances of four training methods in increasing the classification efficiency of SVM. (b) The zoomed version of the area surrounded by the dashed rectangle in (a), for clearly illustrating the classification rate and the number of support vectors obtained by the two SVMs under comparison.

by the number of support vectors used vs the number of correct classifications achieved. In these four methods, our proposed method is most effective in reducing the number of support vectors.

The classification abilities of two SVMs, respectively trained by Osuna's method and our proposed method, are further compared. The SVM trained by Osuna's method, as denoted by the black triangle in Fig 6 (b), needs 901 support vectors and its classification rate is 92.81%. The SVM trained by our proposed method, as denoted by the black circle in Fig 6 (b), needs only 865 support vectors, while its classification rate is 93.10%, higher than that produced by Osuna's method. Moreover, our trained SVM actually has much better generalization ability than the SVM trained by Osuna's method, once checking the histograms of their classification outputs. As shown in Fig 7, the classification outputs of Osuna's SVM concentrate around 0, which means the margins between the positive and the negative samples are narrow. In contrast, most classification outputs of our trained SVM are either larger than 1.0 or smaller than -1.0. This experiment further proves that our training method is better in achieving the generalization ability of the SVM after increasing its efficiency.

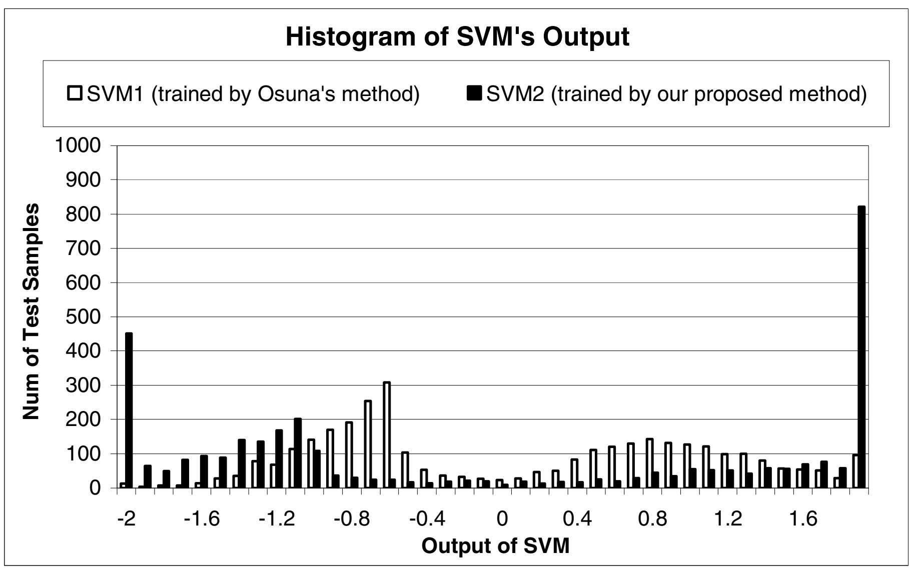


Fig. 7. Histograms of classification outputs on a testing dataset, respectively from our trained SVM (black bars) and Osuna's SVM (white bars)

4 Conclusion

In this paper, we proposed a method to increase the classification efficiency of the SVM, by reducing the number of support vectors used. We noted that the optimal separation hypersurface generated by the standard training method might be unnecessarily over-convoluted around extreme outliers, thus requesting more computational cost without increasing the generalization ability. This situation is actually resulted from the slack variables of outliers that dominate over the objective function, since all slack variables are linearly summed. To overcome this problem, we introduced a nonlinear mapping function to suppress the large slack variables of the outliers. Thus, the separation hypersurface can be simplified and the number of support vectors can be reduced, while the generalization ability of SVM is not sacrificed. To train our reformulated SVM, we modeled it as a dual problem, similar to that of the standard SVM. Therefore, we can borrow any existing SVM training algorithms to iteratively train the reformulated SVM.

Our method has been tested on the UCI machine learning repository, as well as a real clinical problem, i.e., tissue classification in prostate ultrasound images. Compared to the SVM trained by the standard training method and Osuna's method, our method is able to achieve a much more efficient SVM, without losing its generalization ability.

References

1. V.N. Vapnik, "The Natural of Statistical Learning Theory". Springer Verlag, New York, 1995.
2. C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", *Data Mining and Knowledge Discovery,* Vol. 2, pp. 121-167, 1998.
3. E. Osuna, R. Freund and F. Giosi, "Training Support Vector Machines: An Application to Face Detection", Proc. IEEE. Conf. Computer Vision and Pattern Recognition, pp. 130-136, 1997.
4. T. Joachims, "A statistical learning learning model of text classification for support vector machines", Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, New Orleans, 2001.
5. Y. Zhan, D. Shen, "Automated Segmentation of 3D US Prostate Images Using Statistical Texture-Based Matching Method", *Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2003*, Nov 16-18, Canada.
6. V.Wan and S. Renals, "Speaker Verification using Sequence Discriminant Support Vector Machines", IEEE Transactions on Speech and Audio Processing, Vol. 13, No. 2, pages 203-210, March 2005.
7. M. P. S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares Jr. and D. Haussler, "Knowledge-based analysis of microarray gene expression data by using support vector machines", Genetics, Vol. 97, Issue 1, 262-267, January 4, 2000.
8. C. Davatzikos, D. Shen, Z. Lao, Z. Xue and B. Karacali "Morphological classification of medical images using nonlinear support vector machines", *IEEE International Symposium on Biomedical Imaging (ISBI)*, Arlington, VA, April 15-18, 2004.
9. Y. Lecun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drunker, I. Guyon, U. Muller, E. Sackinger, P. Simard and V. Vapnik, "Comparison of Learning Algorithms for Handwritten Digit Recognition", International Conference on Artificial Neural Networks, pp. 53-60, 1995.
10. Y-L. Tian, L. Brown, A. Hampapur, S. Pankanti, A. W. Senior, and R. M. Bolle, "Real World Real-time Automatic Recognition of Facial Expressions", IEEE workshop on performance evaluation of tracking and surveillance, Graz, Austria, March 31, 2003.
11. E. Osuna and F. Girosi, "Reducing the run-time complexity of Support Vector Machines", *ICPR*, Brisbane, Australia, 1998.

12. B. Scholkopf and A. J. Smola, "Learning with Kernels", The MIT Press, Cambridge, 2002.
13. C.J.C Burges, "Simplified support vector decision rules", Proceedings of the 13th International Conference on Machine Learning, pp. 71-77, 1996.
14. Y.-J. Lee and O.L. Mangasarian, "RSVM: Reduced support vector machines", Proceedings of the First SIA International Conference on Data Mining, 2001.
15. S. Hettich, C.L. Blake and C.J. Merz, "Repository of machine learning databases", http://www.ics.uci.edu/~mlearn/MLRepository.html, 1998.
16. B.S. Manjunath and W.Y. Ma, "Texture Features for Browsing and Retrieval of Image Data", *IEEE Trans. on Pattern Anal. Mach. Intell.*, Vol.18, pp. 837-842, 1996.