2-1-2009

# Support Vector Selection and Adaptation for Classification of Remote Sensing Images

Gulsen Taskin Kaya
*Istanbul Technical University*, gtaskink@purdue.edu

okan ersoy
ersoy@purdue.edu

# PURDUE UNIVERSITY TECHNICAL REPORT

# SUPPORT VECTOR SELECTION AND ADAPTATION FOR CLASSIFICATION OF REMOTE SENSING IMAGES

Gülşen Taşkın Kaya
Department of Computational Science and Engineering
Istanbul Technical University
Istanbul, Turkey
gtaskink@purdue.edu

Okan K. Ersoy
School of Electrical and Computer Engineering
Purdue University
W. Lafayette, IN, USA
ersoy@purdue.edu

## ABSTRACT

Classification of nonlinearly separable data by nonlinear support vector machines is often a difficult task especially due to the necessity of a choosing a convenient kernel type. In this study, we propose a new classification method called support vector selection and adaptation (SVSA) that is applicable to both linearly and nonlinearly separable data in terms of some reference vectors generated by processing of support vectors obtained from the linear SVM. The method consists of two steps called selection and adaptation. In these two steps, once the support vectors are obtained by a linear SVM, some of them are rejected and others are selected and adapted to become the reference vectors. Classification is next carried out by using the K Nearest Neighbor Method (KNN) with the reference vectors. In the first step, all support vectors are classified by KNN with respect to the training data excluding the support vectors. The misclassified support vectors are rejected, and the remaining support vectors are chosen as the reference vectors. In the second step, the reference vectors are adapted by moving them towards to or away from the decision boundaries by the Learning Vector Quantization (LVQ) method. At the end of the adaptation process, the reference vectors are finalized. During classification, the class of each input vector is detected with the minimum distance rule in which the distances are calculated from the input vector to all the reference vectors. The SVSA method was experimented with some synthetic and real data, and the experimental results showed that the SVSA is competitive with the traditional SVM.

**Keywords :** SVSA, SVM, data classification, remote sensing, linearly and nonlinearly separable data

## 1. INTRODUCTION

Support Vector Machines (SVM) have been widely used and represent a very attractive approach in classification of remote sensing data. SVM is based on determining an optimum hyperplane that separates the data into two classes with the maximum margin [1]. The hyperplane is obtained from the solution of a constrained quadratic programming (QP) problem. With linearly separable data, the support vectors exist at the margin. Classification is performed subsequently not by using the support vectors further, but by using the hyperplane dependent on the Lagrange coefficients.

In order to achieve classification of nonlinearly separable data, it is necessary to transform input data into a higher dimensional feature space by using a nonlinear kernel function, followed by linear SVM. The resulting system is called nonlinear SVM. The dot products in this feature space can be computed by using the kernel function.

However, there are some difficulties with the nonlinear SVM approach itemized as follows [2]:

- As the training data grows in size, the constraint part for solving the QP problem becomes large, is very memory expensive, and decomposition methods become necessary to decompose an application to parts and to solve the corresponding parts iteration by iteration [3].
- The kernel function represented by a $m \times m$ matrix is fully dense, causing a long CPU time to compute with $m^2$ numbers. Computational complexity depends on $m$ and is $O((m+1)^3)$ for almost any SVM.
- The choice of kernel function is quite important in order to increase classification accuracy. However, it is generally hard to decide which kernel type is optimal to be used with the given data, especially if the structure of the data is not known in advance. Model selection techniques provide the principal ways to select a proper kernel. Usually, the candidates of optimal kernels are prepared using some heuristic rules, and the one which maximizes a given criterion is chosen.
- In order to get a good classification performance with nonlinear SVM, the parameters of the kernel type chosen should also be determined. For this purpose, the cross validation algorithm can be used to estimate the best parameters for the kernel type chosen. However, this means that the nonlinear SVM needs to be run many times depending on the methods of cross validation applied, so this does also take extra time during implementation.

In order to overcome or reduce these difficulties, a new method based on support vector selection and adaptation (SVSA) is introduced and applied to classification of remote sensing images. Our aim is to achieve the classification performance of the nonlinear SVM by using only the support vectors of the linear SVM, which can be considered as the most important vectors closest to the decision boundary.

By further selecting the support vectors which are most helpful to classification and adapting the chosen support vectors to be used as reference vectors by LVQ for increased classification accuracy, a highly accurate learning system is generated for linear as well as nonlinear classification. Subsequent classification is based on labeled reference vectors.

In addition, a hybrid SVSA method is generated for classification of certain types of data. In the hybrid SVSA, both SVSA and linear SVM are used for classification depending on a given threshold value.

The paper consists of seven sections. Section 1 is introduction. The SVM method is presented in Section 2. The SVSA and the hybrid SVSA methods are covered in Sections 3 and 4, respectively. The experiments done with both synthetic data and remote sensing data are presented in Sections 5, and 6, respectively. Conclusions and discussion of future research are presented in Section 7.

## 2. SUPPORT VECTOR MACHINES

The SVM theory is a new statistical approach and has drawn much attention in recent years. It was initially developed by V. Vapnik, especially for classification of separable data [4]. It was further generalized to handle nonseparable data. A SVM classifier creates an optimum hyperplane that lies in a transformed input space with the property of maximized distance to the nearest class examples. The parameters of the solution hyperplane are derived by a constrained quadratic programming optimization problem [5].

For the optimal hyperplane $\mathbf{wx}+b=0$, $\mathbf{w}\in R^N$ and $b\in R$, the classification of the testing sample $\mathbf{x}$ in a 2-class problem is obtained by

$$f(\mathbf{x})=\mathrm{sign}(\mathbf{wx}+b)=\mathrm{sign}\left(\sum_{\mathrm{i=1}}^{\mathrm{N_S}}\alpha_i m_i \mathbf{x}_i.\mathbf{x}\right)$$

where $N_S$ is the number of support vectors, $\mathbf{x}_i$ is the $i$th support vector, $\alpha_i$ is the $i$th Lagrange multiplier, and $m_i\in\{-1,+1\}$ describes which class **x** belongs to.

In most cases, searching for a suitable hyperplane in the input space of a nonlinear classification problem is too restrictive to be of practical use. One solution to this is mapping the input space into a higher dimensional feature space and searching for the optimal hyperplane in this feature space. Let $\mathbf{z}=\varphi(\mathbf{x})$ denote the corresponding feature space vector with a mapping $\varphi$ from $R^N$ to a feature space $Z$. The mapping can be indirectly represented by a kernel $K(*,*)$ which corresponds to the inner product of the transformed input vectors in the form.

$$\mathbf{z}_i\cdot\mathbf{z}_j=\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j)=K(\mathbf{x}_i,\mathbf{x}_j)$$

Finally, the decision function becomes

$$f(x)=\mathrm{sign}\left(\sum_{i=1}^{N_S}\alpha_i m_i K(\mathbf{x}_i,\mathbf{x})+b\right)$$

Functions that satisfy Mercer's theorem [6] can be used as kernels. Typical kernel functions are the following:

Linear kernel: $K(\mathbf{x},\mathbf{y})=\mathbf{x}\cdot\mathbf{y}$

$$\text{Polynomial kernel: } K(\mathbf{x},\mathbf{y}) = (\gamma\mathbf{x}\cdot\mathbf{y}+c)^d$$

$$\text{Radial basis kernel: } K(\mathbf{x},\mathbf{y}) = \exp\left(-\gamma\cdot\| \mathbf{x}\cdot\mathbf{y} \|^2\right)$$

## 3. SUPPORT VECTOR SELECTION AND ADAPTATION (SVSA)

A new method called support vector selection and adaptation (SVSA) is proposed and applied to classification of remote sensing images. A particular case is the classification of damage after an earthquake using satellite images. We used pre and post earthquake satellite images in the city of Bam, Iran to extract the ground truth of the damaged buildings.

The SVSA method uses the support vectors obtained from linear SVM [7], eliminates some of them for not being sufficiently useful for classification, and adaptively modifies the selected support vectors which are next used as reference vectors for classification. In this way, nonlinear classification is achieved without needing a kernel.

### 3.1. Selection and Adaptation

Let $X = \{(x_1,\bar{x}_1),\ldots,(x_N,\bar{x}_N)\}$ represent the training data with $x_i \in R^p$ and the class labels $\bar{x}_i \in \{1,\ldots,M\}$. $N$, $M$ and $p$ denote the number of training samples, the number of classes and the number of features, respectively. After applying the linear SVM to the training data, the support vectors are obtained as

$$S = \left\{(s_i,\bar{s}_i) \;\middle|\; (s_i,\bar{s}_i) \in X \quad i = 1,\ldots,k\right\}$$

$$T = \left\{(t_i,\bar{t}_i) \;\middle|\; (t_i,\bar{t}_i) \in X \setminus S \quad i = 1,\ldots,N-k\right\}$$

where $k$ is the number of support vectors, $S$ is the set of support vectors with the class labels $\bar{s}$, and $T$ is the set of training data vectors with the class labels $\bar{t}$, excluding the support vectors.

In the selection stage, the support vectors in the set $S$ are classified with respect to the set $T$ by using the K-Nearest Neighbor algorithm [8]. The labels of the support vectors are obtained as:

$$\bar{s}_i^p = \left\{\bar{t}_l \;\middle|\; l = \arg\min_{1\le j\le N-k}\left\{\left\|s_i - t_j\right\|\right\} \quad i = 1,\ldots,k\right\}$$

where $\bar{s}_i^p$ is the predicted label of the $i^{th}$ support vector.

Then, the misclassified support vectors are removed from the set $S$. The remaining support vectors are called reference vectors and constitute the set $R$:

$$R = \left\{ (s_i, \bar{s}_i) \ \middle|\ (s_i, \bar{s}_i) \in S \quad \text{and} \ \bar{s}_i^p = \bar{s}_i \quad i = 1, \ldots, k \right\}$$

The aim of the selection process is to select the support vectors which best describe the classes in the training set.

The reference vectors to be used for classification are next adaptively modified based on the training data in a way to increase the distance between the neighboring reference vectors with different class labels. The main idea of adaptation is that a reference vector causing a wrong decision should be further away from the current training vector, and the nearest reference vector with the correct class should be closer to the current training vector. Adaptation is achieved by using the Learning Vector Quantization (LVQ) algorithm [9,10] as described below.

Let $x_j$ be one of the training samples with label $y_j$ [11]. Assume that $\mathbf{r_w}(t)$ is the nearest reference vector to $x_j$ with label $y_{r_w}$. If $y_j \neq y_{r_w}$ then the adaptation is applied as follows:

$$\mathbf{r_w}(\mathrm{t}+1) = \mathbf{r_w}(t) - \eta(t)(\mathbf{x_j} - \mathbf{r_w}(t))$$

On the other hand, if $\mathbf{r}_l(t)$ is the nearest reference vector to $x_j$ with label $y_{r_l}$ and $y_j = y_{r_l}$ then

$$\mathbf{r}_l(\mathrm{t}+1) = \mathbf{r}_l(t) + \eta(t)(\mathbf{x_j} - \mathbf{r}_l(t))$$

where $\eta(t)$ is a descending function of time called the learning rate. It is also adapted in time by

$$\eta(t) = \eta_0 e^{-t/\tau}$$

where $\eta_0$ is the initial value of $\eta$, and $\tau$ is a time constant.

The adapted reference vectors are used for classification of the training and testing sets. For this purpose, the K-Nearest Neighbor method is applied to classify the samples with respect to the reference vectors. The Euclidian distances from the input vector to the reference vectors are calculated, and classification is done based on the majority class of the K nearest reference vectors.

## 4. THE HYBRID SVSA

It is known that linear SVM gives the best classification accuracy for linearly separable data. According to the results obtained with some experiments done with both SVSA and SVM, it was observed that the SVSA as well as nonlinear SVM are not efficient classifiers especially with linearly separable data and very nonlinearly separable data in comparison to the linear SVM. Therefore, the hybrid SVSA was developed.

During implementation, since the results of the linear SVM are also available, by utilizing this information, the hybrid model was generated by using consensus between the results of the linear SVM and the results of the SVSA.

For this purpose, the perpendicular distance to the hyperplane obtained by the linear SVM for each data sample is calculated based on the Euclidian distance. If the distance is greater than a given threshold, the data is classified by the linear SVM; otherwise the SVSA algorithm is applied. The schema for hybrid SVSA is shown in Figure 1.

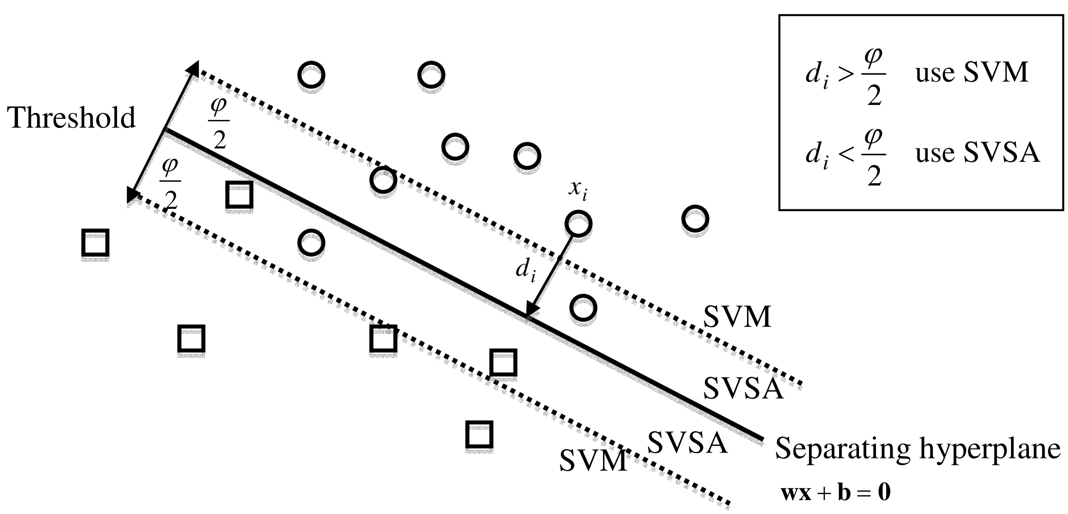



Figure 1. Processing schema for hybrid SVSA.

In the hybrid model, the SVSA method is more effective for classification of the data near the separating hyperplane, and the linear SVM is effective in the classification of the other data. We obtained comparatively good classification accuracy with the hybrid model with most data.

## 5. EXPERIMENTS WITH SYNTHETIC DATA

### 5.1. Data Representation

In our experiments, we first generated different types of synthetic data with different types of nonlinearity in order to compare the classification performance of the proposed method with the SVM. Four examples of types of data were banana shaped data and the data created by using given mean vectors and covariance matrices in a way to provide nonlinearity [12]. All the datasets are shown in Figure 2.

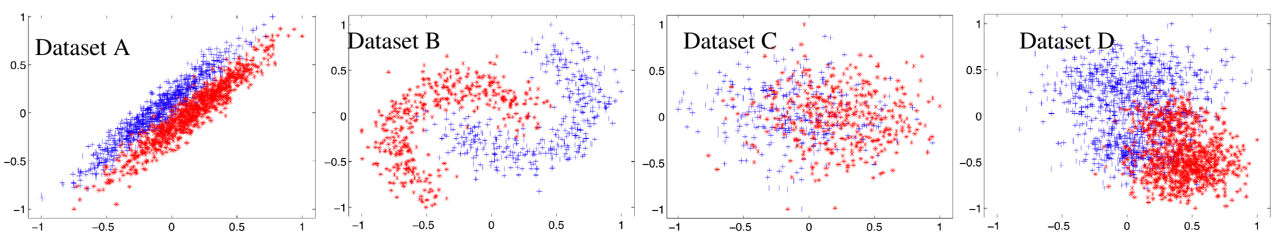



Figure 2. The synthetic datasets used in the experiments.

The process of forming the training and the testing data is as follows:

1. The data with two different classes is synthetically created.
2. The training and testing datasets are randomly chosen from the created dataset with proportions 40% and 60%, respectively.
3. Ten different datasets are generated by repeating step 2 ten times.
4. Then, each classification algorithm is run with both training and testing data over the ten datasets in order to classify them.
5. Afterwards, the classification error for each dataset is computed and averaged over the ten datasets.

## 5.2. Scaling Data

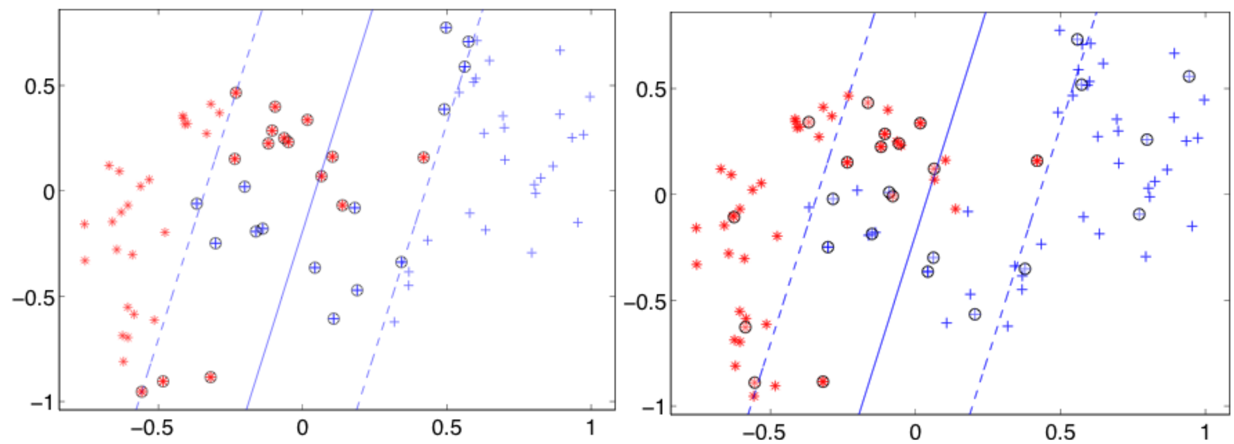
Scaling data before applying SVM is important [13]. The main advantage of scaling is to avoid features in larger numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during computations. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large feature values might cause numerical problems. In this work, each feature of a data vector was linearly scaled to the range [-1,+1] before doing experiments.

## 5.3. Parameter Tuning

For nonlinear SVM, cross-validation within the original datasets was utilized to provide a nearly unbiased estimate of the prediction error rate. The performance of classifying the datasets was evaluated using 10-fold cross-validation [14]. For this purpose, each dataset was divided into ten subsets of approximately equal size. Sequentially one subset was tested using the classifier trained on the remaining 9 subsets. Thus, each dataset instance was predicted once, and the cross-validation accuracy was the average percentage of data which was correctly classified. There are two parameters while using RBF kernels: kernel parameter $\gamma$ and penalty parameter $C$. These were also estimated by cross-validation.

## 5.4. Implementation with SVSA

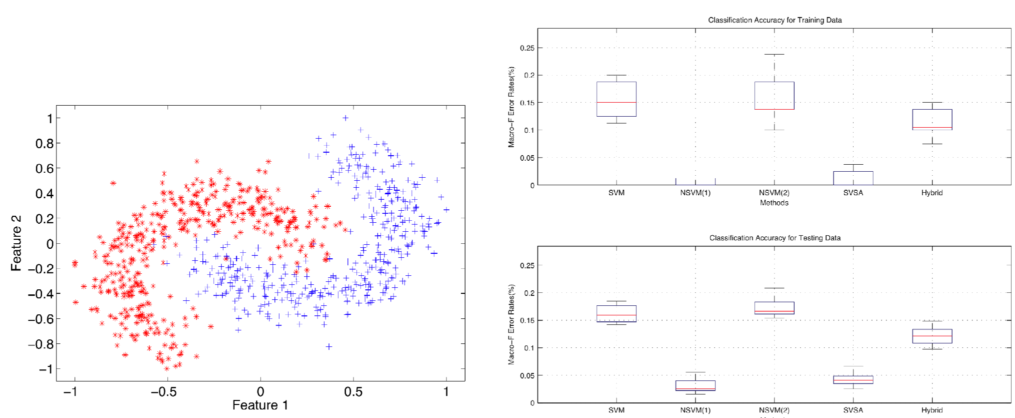
After determining support vectors as illustrated in Figure 3 (a) by the linear SVM, the selection and adaptation process is applied with respect to the training data, and adapted reference vectors are obtained as in Figure 3 (b). In the selection stage, each support vector is first classified by 1-KNN with respect to the training data excluding the support vectors, and then the misclassified support vectors are excluded from the reference vector set. During adaptation, the remaining support vectors called reference vectors are adapted based on all the training data by means of the LVQ. Figure 3 (b) shows the resulting reference vectors and how they are nonlinearly located by the SVSA with respect to the data.



(a) (b)

Figure 3. (a) The distribution of support vectors obtained by the linear SVM for Dataset B and illustrated with blue and red circles (b). The distribution of reference vectors obtained by the SVSA for Dataset B and illustrated the same way as in (a).

Afterwards, both training and testing data for dataset B is classified by using 1-KNN with the reference vectors, and classification errors of each dataset are calculated. In Figure 4, the classification performance of all the classifiers with the banana-shaped data is shown.




(a) (b)

Figure 4. (a) The banana-shaped data. (b) The classification errors with respect to all methods for both training and testing data. NSVM (1) and NSVM (2) refer to the nonlinear SVM with radial basis kernel type and polynomial kernel type, respectively.

As observed in Figure 4, the classification accuracy of nonlinear SVM depends on the choice of the kernel type. For example, NSVM (1) with radial basis kernel is better than NSVM (2) with polynomial kernel in terms of classification performance in Figure 4. However, in applications, what type of kernel is supposed be chosen is not generally known in advance. In such a case, all types of kernel should ideally be applied, and then according to their classification accuracy, the method which gives the smallest error of classification with the training dataset, especially using cross-validation, should be chosen as the method for classification of the testing data. However, this process takes much time especially since the solution of nonlinear SVM takes a long time in comparison to linear SVM. In these experiments, the classification performance of the SVSA method was competitive with nonlinear SVM as shown in Figure 4, and the SVSA does not need choosing any type of kernel.

In synthetic data experiments in which the data was not linearly separable, the distributions of the wrong decisions made by the linear SVM were usually found to be near the separating hyperplane. This means that the linear SVM caused a lot of misclassified samples with such data near the border of classification. Because of this, the classification error of the hybrid model was quite high in comparison to SVSA if the threshold value was not sufficiently large. In the experiments, the threshold value was chosen as 0.3. At this value the linear SVM contributes to classification performance more than the SVSA. The classification performance of the hybrid SVSA can be increased by choosing a bigger threshold value in such cases.

All the methods were applied with all the datasets, and the mean values of classification errors for all the datasets were obtained by averaging over the ten datasets for both training and testing data. The mean values of classification errors of all the methods are summarized in Table 1.

Table 1. The mean values of errors with the datasets.

| Dataset | | METHODS | | | | |
|---|---|---|---|---|---|---|
| | | SVM | NSVM (1) | NSVM (2) | SVSA | Hybrid |
| A | Training | 0.057 | 0.054 | 0.060 | 0.038 | 0.048 |
| | Testing | 0.061 | 0.062 | 0.064 | 0.077 | 0.060 |
| B | Training | 0.152 | 0.007 | 0.159 | 0.191 | 0.019 |
| | Testing | 0.162 | 0.030 | 0.173 | 0.041 | 0.034 |
| C | Training | 0.275 | 0.249 | 0.276 | 0.191 | 0.256 |
| | Testing | 0.280 | 0.295 | 0.280 | 0.306 | 0.279 |
| D | Training | 0.152 | 0.111 | 0.157 | 0.091 | 0.096 |
| | Testing | 0.157 | 0.127 | 0.161 | 0.137 | 0.135 |

In dataset B, the threshold value for the hybrid SVSA was chosen bigger than the one used in the experiments related to Figure 4. That is why the classification error of the hybrid SVSA in Table 1 is less than the classification error of the hybrid SVSA in Figure 4.

According to the results obtained by applying all the algorithms to all the datasets, it was observed that the classification performance of the hybrid method was better than all the other methods with linearly separable data and with data with extreme nonlinearity. If the data was not linearly separable, the SVSA was competitive with the nonlinear SVM and better than the linear SVM in terms of classification accuracy.

## 5. EXPERIMENTS WITH REMOTE SENSING DATA

### 5.1 Experiment 1: Earthquake Data

Pre- and post-earthquake Quickbird satellite images with high resolution (0.6 m) were used to identify damage patterns in the city of Bam, Iran during the 2003 earthquake. The ground truth of damaged and nondamaged buildings was generated by using the pre- and post- earthquake images from the area of interest. The SVSA and the hybrid SVSA were used for the classification of the damaged and nondamaged buildings in comparison to linear and nonlinear SVM methods.

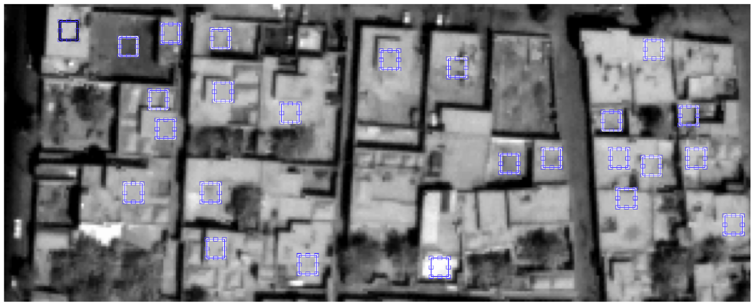


Figure 5. The panchromatic satellite image captured by Quickbird before the Bam earthquake. Each blue square region refers to a nondamaged building before the Bam earthquake.

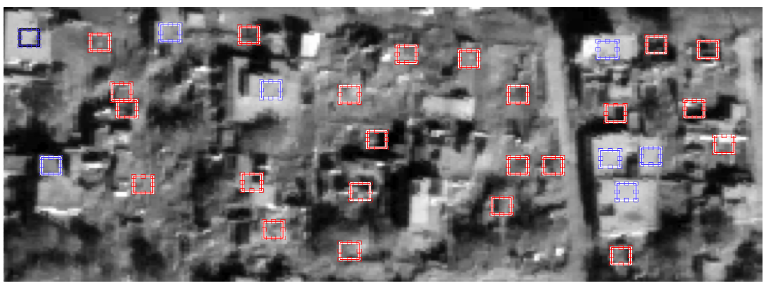


Figure 6. The panchromatic satellite image captured by Quickbird after the Bam earthquake. The red square regions refer to the damaged buildings. The blue ones have the same meaning as in the previous image.

For the evaluation of the methods, after collecting the samples for damaged and undamaged buildings from the satellite images as shown in Figures 5 and 6, training and testing data were randomly chosen from all collected samples with 40 and 60 percent proportions, respectively. The classification results obtained by applying the methods to the datasets are shown in Figure 7.

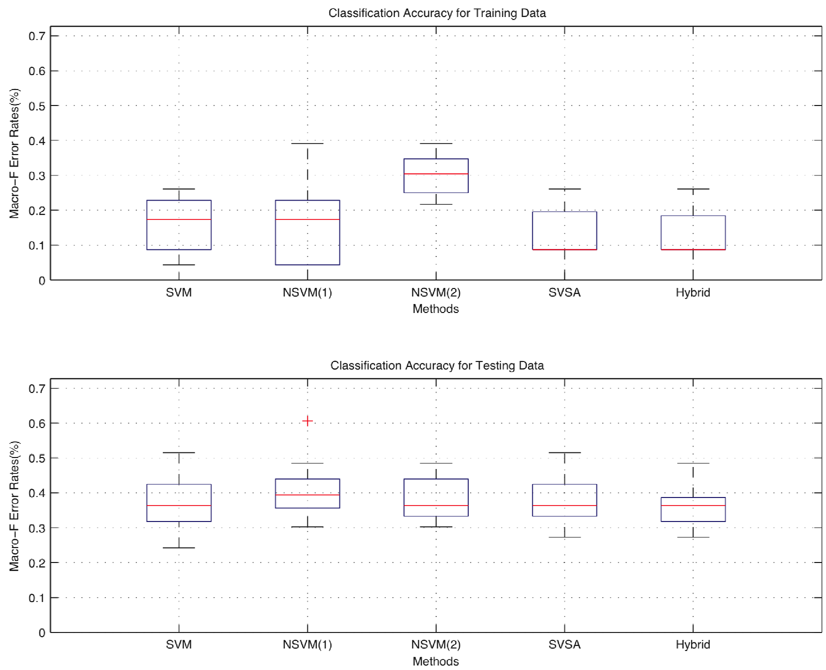



Figure 7. Macro-F Error rates with the training and testing datasets for the urban area of interest in Bam, Iran.

According to these results, the hybrid model gave best classification accuracy in comparison to all the methods applied during classification. The SVSA was also better than the nonlinear SVM in this experiment in terms of classification performance.

## 5.2 Experiment 2 : Colorado Dataset

Classification was performed with the Colorado dataset [15] consisting of the following four data sources:

1. Landsat MSS data (four spectral data chanels)
2. Elevation data (in 10-m contour intervals, one data channel).
3. Slope data ( $0-90^0$ in $1^0$ increments, one data channel).
4. Aspect data ( $1-180^0$ in $1^0$ increments, one data channel).

Each channel comprised an image of 135 rows and 131 columns, and all channels were spatially co-registered in Colorado. It has ten ground-cover classes which are listed in Table 2. One class is water; the others are forest types. It is very difficult to distinguish among the forest types using Landsat MSS data alone since the forest classes show very similar spectral response.

Table 2. Training and testing samples of the Colorado dataset.

| Class # | Information Class | Training Size | Testing Size |
|---|---|---|---|
| 1 | Water | 408 | 195 |
| 2 | Colorado Blue Spruce | 88 | 24 |
| 3 | Mountane/ Subalpine meadow | 45 | 42 |
| 4 | Aspen | 75 | 65 |
| 5 | Ponderosa Pine 1 | 105 | 139 |
| 6 | Ponderose Pine/Douglas Fir | 126 | 188 |
| 7 | Engelmann Spruce | 224 | 70 |
| 8 | Douglas Fir/White Fir | 32 | 44 |
| 9 | Douglas Fir/Ponderosa Pine/Aspen | 25 | 25 |
| 10 | Douglas Fir/White Fir/Aspen | 60 | 39 |
| | Total | 1188 | 831 |

In this experiment, 45 experiments were done with the Colorado dataset for binary classification and, a subset of the binary classification results were selected and are shown in Figure 8.

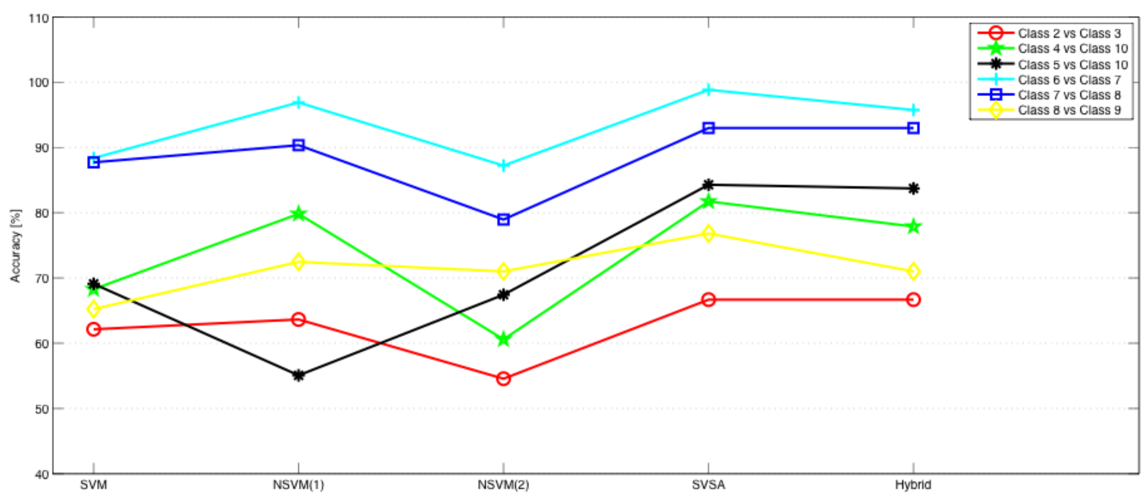



Figure 8. Colorado dataset results. Each curve corresponds to one experiment with two classes.

We obtained higher classification accuracies with the SVSA in comparison to the nonlinear SVM with the RBF kernel and all the other methods. The performance of the hybrid SVSA was also satisfactory as seen in Figure 8. The classification accuracies done with all the experiments are also summarized in Table 3.

Table 3**.** The classification accuracies of all the experiments with the Colorado dataset. $\lambda$ is the threshold value for hybrid SVSA. The best results are shown in bold.

| CLASSES | | METHODS | | | | | $\lambda$ |
|---|---|---|---|---|---|---|---|
| | | **SVM** | **NSVM (1)** | **NSVM (2)** | **SVSA** | **Hybrid SVSA** | |
| 1 vs. | 2 | **100** | 89.04 | **100** | **100** | 100 | 1 |
| | 3 | 91.98 | **94.09** | 82.7 | 92.41 | 92.41 | 1 |
| | 4 | 98.85 | **99.62** | 98.08 | 99.23 | 99.62 | 1 |
| | 5 | **99.4** | 88.02 | 98.2 | 85.93 | 94.31 | 1 |
| | 6 | **99.74** | 98.69 | 98.96 | 98.43 | 98.96 | 1 |
| | 7 | 99.62 | 89.43 | **100** | 99.62 | 99.49 | 2 |
| | 8 | **99.16** | 89.12 | 89.12 | 96.65 | 96.65 | 1 |
| | 9 | 88.64 | **99.09** | 98.64 | **99.09** | **99.09** | 1 |
| | 10 | **100** | **100** | **100** | **100** | 100 | 1 |
| 2 vs. | 3 | 62.12 | 63.64 | 54.55 | **66.67** | **66.67** | 2 |
| | 4 | 93.26 | **97.75** | 82.02 | 91.01 | 92.14 | 0.1 |
| | 5 | 58.9 | **68.71** | 57.06 | 61.35 | 61.96 | 1 |
| | 6 | 83.96 | **94.34** | 83.49 | 83.49 | 83.46 | 2 |
| | 7 | **100** | **100** | **100** | **100** | **100** | 2 |
| | 8 | 55.88 | **70.59** | 51.47 | 69.12 | 69.12 | 1.7 |
| | 9 | **97.96** | 93.88 | 91.84 | 93.88 | 95.92 | 1 |
| | 10 | 82.54 | **88.89** | 74.6 | 77.78 | 80.95 | 0.3 |
| 3 vs. | 4 | 71.03 | 63.55 | 78.5 | 91.59 | **94.39** | 2 |
| | 5 | 62.98 | 23.76 | **79.56** | 46.41 | 61.88 | 0.1 |
| | 6 | 83.48 | 28.26 | 86.52 | 80.43 | **90.43** | 1 |
| | 7 | **100** | **100** | **100** | 97.32 | 99.11 | 1.1 |
| | 8 | **67.44** | 56.98 | 60.47 | 66.28 | **67.44** | 1 |
| | 9 | 88.06 | 82.09 | **89.55** | 80.6 | 86.57 | 0.2 |
| | 10 | 76.54 | 61.73 | 74.07 | 88.89 | **90.12** | 1 |
| 4 | 5 | 60.78 | 63.73 | **64.22** | 60.29 | 60.29 | 1 |
| | 6 | 66.01 | 70.75 | **72.33** | 69.96 | 68.77 | 1 |
| | 7 | 95.56 | **97.78** | 66.67 | 94.07 | 94.07 | 1 |
| | 8 | 43.12 | **46.79** | 39.45 | 44.04 | 44.04 | 1 |
| | 9 | 36.67 | 41.11 | **71.11** | 48.89 | 47.78 | 1.5 |
| | 10 | 68.27 | 79.81 | 60.58 | **81.73** | 77.88 | 2 |
| 5 | 6 | 44.65 | 51.07 | **61.47** | 38.84 | 44.64 | 0.1 |
| | 7 | 98.56 | 93.3 | 97.61 | 98.56 | **99.04** | 1 |
| | 8 | 76.5 | 72.13 | **75.96** | 75.41 | **75.96** | 1 |
| | 9 | 74.39 | 71.34 | **77.44** | 73.17 | 73.17 | 1 |
| | 10 | 69.1 | 55.06 | 67.42 | **84.27** | 83.71 | 1.5 |
| 6 | 7 | 88.37 | 96.9 | 87.21 | **98.84** | 95.74 | 1 |
| | 8 | 70.69 | 75.43 | **81.03** | 74.14 | 75 | 1 |
| | 9 | 74.18 | 74.18 | **80.28** | 75.59 | 75.12 | 1 |
| | 10 | 68.72 | **85.9** | 75.33 | 82.82 | 82.82 | 1 |
| 7 | 8 | 87.72 | 90.35 | 78.95 | **92.98** | **92.98** | 1 |
| | 9 | 75.79 | **89.47** | 73.68 | 85.26 | 71.43 | 1 |
| | 10 | 94.5 | **97.25** | 89.91 | 93.58 | **97.25** | 1 |
| 8 | 9 | 65.22 | 72.46 | 71.01 | **76.81** | 71.01 | 1 |
| | 10 | 54.22 | 51.81 | 51.81 | **74.7** | 72.29 | 1 |
| 9 | 10 | 53.13 | **75** | 64.06 | 70.31 | 71.88 | 1 |
| ***Average*** | | ***78.39*** | ***77.62*** | ***78.60*** | ***81.34*** | ***82.13*** | |

In summary, the classification performance of the SVSA method is better than the nonlinear SVM in some cases, and is quite close to the classification performance of the nonlinear SVM in some other cases as seen in Table 3. Especially with extremely nonlinear data such as classification of class 7 against class 9 in Table 8, the SVSA method is considerably better than all the other methods.

## 7. CONCLUSIONS AND FUTURE WORK

In this study, we addressed the problem of classification of remote sensing data using the proposed support vector selection and adaptation (SVSA) method and hybrid SVSA method in comparison to linear and nonlinear SVM.

The SVSA method consists of selection of the support vectors which contribute most to the classification accuracy and adaptation of them based on the class distributions of the data. It was shown that the SVSA method gives competitive classification performance in comparison to the linear and nonlinear SVM with both synthetic data and real world data.

With linearly separable data, as well as extremely nonlinear data, it was observed the linear SVM is better than other methods in terms of classification accuracy. The hybrid model (hybrid SVSA) was developed to improve classification performance further with such data. In the hybrid SVSA, both linear SVM and SVSA are used to classify the data based on a given threshold value. It was observed that the hybrid SVSA is quite effective in classification of such data.

During implementation, it was observed that the classification performance for each class has different accuracy in all methods. This is especially related to unbalanced data in which one class is majority while the other one represents a rare event. In the future, we plan to implement resampling classification strategies for rare event detection and to generalize the methods to multiclass problems.

**ACKNOWLEDGMENTS**

The authors would like to acknowledge the Project of Disaster Information and Early Warning System funded by the Scientific and Technological Research Council of Turkey (TUBITAK), Istanbul Technical University (ITU) for providing Quickbird satellite images of Bam, Iran, and Purdue University for providing an effective environment for research and collaboration.

## REFERENCES


[1] A. Shmilovici, "The Data Mining and Knowledge Discovery Handbook", *Springer*, 2005.

[2] Yue Shihong Li Ping Hao Peiyi, "Svm Classification :Its Contents and Challenges", *Appl. Math. J. Chinese Univ. Ser. B*, Vol. 18 (3), 332-342, 2003.

[3] John Platt, "Using Analytic QP and Sparseness to Speed Training of Support Vector Machines", *MIT Press*, 557-563, 1999.

[4] V. Cherkassky and F. Mulier, "Learning From Data : Concepts, Theory and Methods", *Wiley-interscience*, 1998.

[5] J. C. Wang, J. F. Wang, K. W. He and C. Hsu, "Environmental Sound Classification Using Hybrid SVM/KNN Classifier and MPEG-7 Audio Low-Level Descriptor", *International Joint Conference on Neural Networks*, 2006.

[6] R. Courant and D. Hilbert, "Methods of Mathematical Physics", *Interscience Publishers*, 1953.

[7] C.C. Chang and C. Lin, "LIBSVM: A Library for Support Vector Machines", *http://www.csie.ntu.edu.tw/~cjlin/libsvm/*, 2001.

[8] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13(1), pp.21–27, 1967.

[9] T. Kohonen, "Learning vector quantization for pattern recognition," *Tech. Rep., TKK-F-A601, Helsinki University of Technology,* 1986.

[10] T. Kohonen, J. Kangas, J. Laaksonen, and K. Torkkola, "Lvqpak: A software package for the correct application of learning vector quantization algorithms," *Neural Networks, IJCNN., International Joint Conference*, vol. 1, pp. 725 – 730, 1992.

[11] N. G. Kasapoğlu and O. K. Ersoy, "Border Vector Detection and Adaptation for Classification of Multispectral and Hyperspectral Remote Sensing", *IEEE Transactions on Geoscience and Remote Sensing*, Vol: 45-12, pp: 3880-3892, December 2007.

[12] R.P.W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D.M.J. Tax, S. Verzakov, “A Matlab Toolbox for Pattern Recognition”*, Delft University of Technology, PRTools4.1,* 2007.

[13] Jun Cai and Yanda Li, “Classification of Nuclear Receptor Subfamilies with RBF Kernel in Support Vector Machine”, *Springer-Verlag Berlin Heidelberg,* 680–685, 2005.

[14] Bengio, Y., Grandvalet, Y, “No unbiased estimator of the variance of k-fold cross-validation”, *Journal of Machine Learning Res*, Vol. **5** pp. 1089-1105, 2004.

[15] J. A. Benediktsson, P. H. Swain, O. K. Ersoy, "Neural Network Approaches versus Statistical Methods in Classification of Multisource Remote Sensing-Data," *IEEE Transactions Geoscience and Remote Sensing,* Vol. 28, No. 4, pp. 540-552, July 1990.