An Effective Ensemble-based Classification Algorithm for High-Dimensional Steganalysis

Fengying He
College of Mathematics and Computer Science Fuzhou University, Fuzhou, China
Email: hfy@fzu.edu.cn

Shangping Zhong and Kaizhi Chen
College of Mathematics and Computer Science Fuzhou University, Fuzhou, China
Email: {spzhong, ckz}@fzu.edu.cn

***Abstract*—Recently, ensemble learning algorithms are proposed to address the challenges of high dimensional classification for steganalysis caused by the curse of dimensionality and obtain superior performance. In this paper, we extend the state-of-the-art steganalysis tool developed by Kodovsky and Fridrich: the Kodovsky's ensemble classifier and propose a novel method, called CS-RS for high-dimensional steganalysis. Different from the Kodovsky's ensemble classifier which selects features in a completely random way, the proposed CS-RS modifies the generation method of feature subspaces. Firstly, our method employs the chi-square statistic (CS) to measure the weight of each feature in the original feature space and sorts features according to weights. Then the sorted original feature space is partitioned into two parts according to a given dividing point: high correlation part and low correlation part. Finally, the feature subset is formed by selecting features randomly in each part according to the given sampling rate. Experiments with the steganographic algorithms HUGO demonstrate that the proposed CS-RS using the FLD classifier offers training complexity comparable to the Kodovsky's classifier and significantly increases the performance of the Kodovsky's classifier in less than 1000-dimensional feature subspaces, gaining 1.2% on the optimal result. In addition, the proposed algorithm outperforms Bagging and AdaBoost and can offer accuracy comparable to L-SVM.**

***Index Terms*—high-dimensional feature; steganalysis; ensemble classifier; chi-square statistic; fld**

I. Introduction

With the increased sophistication of steganographic algorithms, steganalysis has already begun using feature spaces of increased dimensionality [1]. The most accurate spatial domain steganalysis of embedding (LSB matching) uses the 686-dimensional SPAM features [2] while a 1,234-dimensional Cross-Domain Feature (CDF) set was employed in [3] to attack YASS [4]. Moreover, the recent results of the steganalysis competition BOSS [5] indicate that there is little hope that a human-designed low-dimensional feature space effective against HUGO [6] exists and constructed a 24993-dimensional HOLMES feature proved especially effective against HUGO.

Classifying high dimensional features poses a serious challenge to steganalysis due to several reasons. The required number of labeled samples for supervised learning methods increases exponentially with the dimensionality to achieve high generalization accuracy [7]. Increase in the dimensionality makes machine learning methods computationally intractable [8]. The data points become sparse in the higher dimensions, which makes the learning task very difficult if not impossible.

Even though the support vector machine (SVM) seems to be the most popular machine learning tool used in steganalysis, SVMs are quite restrictive due to the complexity of SVM will be increased rapidly with the dimensionality of feature space growing. Traditional approach that tackles classification problems with a high-dimensional feature space employs dimensionality reduction or projection techniques like Principal Component Analysis (PCA) or KL-transformation. These project the original attribute or feature space into lower dimensions by creating new dimensions that are combinations of the original features. While such techniques help in reducing dimensionality, one drawback is that the new dimensions can be difficult to interpret, thus making it hard to relate the instances to the original dimensions [9]. A second drawback for the dimension reduction techniques like PCA is the huge computational complexity when calculate the covariance matrix of feature vectors with thousands of dimensions [10].

To address the challenges associated with the curse of dimensionality arising in staganalysis, Kodovsky [11] proposed ensemble classifier built as random forest with the FLD [12] as a base learner. In Kodovsky's ensemble classifier, each base learner is trained on randomly selected subspaces of the original feature space, the dimensionality of the random subspaces can be chosen to be much smaller than the full dimensionality, which significantly decreases the training complexity and each learner is trained on a bootstrap sample drawn from the training set rather than on the whole training set, which further increases the mutual diversity of the base learners. Experiment results demonstrated the ensemble classifier in [11] can provides performances equivalent to that of a

Technology Innovation Platform Project of Fujian Province(2009J1007); Natural Science Foundation of Fujian Province(2010J01331)

doi:10.4304/jsw.9.7.1833-1840

Support Vector Machine (SVM) [13] for steganalysis of large databases with large feature vectors.

But the features in subspaces used to train the base learners in [11] were selected randomly from the original feature space, this approach is not suitable for high dimensional feature space consisting of thousands of features, because the random sampling may result in the selected subspace contains many features which are uninformative to classification, thus affecting the classification performance of the base learners.

In this paper, we propose a new method, called CS-RS which modifies the generation method of feature subspaces in [11] and therefore enhances classification performance over high-dimensional feature space. Firstly the weight of each feature in original feature space is computed with respect to the correlation between the feature and the class label using the chi-square statistic, the weights are treated as the classification ability of the features, the higher the weight, the more informative the feature to the classification. Then the features in original feature space are sorted from high to low according to weights and the sorted original feature space is partitioned into two parts according to a given dividing point. Finally, the feature subset is formed by selecting features randomly in each part according to the given sampling rate. Such a method increases the probability for informative features to be included in each subspace, thus increases the performance of the base learners and without sacrificing diversity between them due to the random selection of features in each part. Experiments with steganographic algorithm HUGO show that, the proposed algorithm is an effective method for solving the problems caused by high-dimensional feature vectors, the detection accuracy is much better than *the Kodovsky's classifier* in low dimensional feature subspace without increasing training complexity obviously. The proposed algorithm gets the best result of 85.97% which increases the optimal value of *the Kodovsky's classifier* of 1.2%. In addition, the proposed algorithm outperforms Bagging and AdaBoost and can offer accuracy comparable and often even better to L-SVM.

The rest of this paper is organized as follows. In SectionⅡ, we recall the important concepts of *the Kodovsky's classifier*. In SectionⅢ, we describe the CS-RS algorithm in detail. In SectionⅣ, we implement experiments and present an empirical analysis of the proposed algorithm. Finally, we give the conclusion and some suggestions for future work in the last Section.

II. The Kodovsky's Ensemble Classifier

The ensemble classifier proposed in [11] consists of *L* independently trained base learners, designed to keep low complexity and overall simplicity. To facilitate the description, we use the same notation as in [11].Denote *d* as the dimensionality of the high dimensional feature space, N^{trn} and N^{tst} the number of training and testing samples from each class, $x_m \in R^d$, *m=1,…,* N^{trn} as the feature vector of dimension *d* computed from the training set, $y_k \in R^d$, *k=1,…,* N^{tst} as the feature vector of dimension *d* computed from the testing set and *L* the number of base learners. The set of all training and testing samples will be denote $x^{trn} = \{x_m\}_{m=1}^{N^{trn}}$ and $y^{tst} = \{y_k\}_{k=1}^{N^{tst}}$.

For $D \subset \{1,...d\}$, $x^{(D)}$ is a |*D*|-dimensional feature vector consisting only of those features from *x* whose indices are in *D*, preserving their original order.

The Kodovsky's classifier is to learn separately individual base learners. Each base learner, denoted B_l with $l \in \{1,...,L\}$ is a mapping $R^{d_{sub}} \rightarrow \{0,1\}$, '0' standing for cover and '1' for stego, trained on X_l= $\{x_m^{(D_l)}\}_{m \in N_l^b}$,where $D_l \subset \{1,...d\}$,| D_l|=d_{sub}<<*d,* is the *lth* random subspace and N_l^b ,| N_l^b |=N^{trn} is the *lth* bootstrap sample of the set of indices *{1,…,* N^{trn}*}*.For all test examples $y \in y^{tst}$, the final prediction $B(y)$ is obtained by a majority vote:

$$B(y) = \begin{cases} 1 & if \sum_{l=1}^{L} B_l(y) > L/2 \\ 0 & otherwise \end{cases}$$

The Kodovsky's ensemble classifier is of low computational complexity and achieves performance as good as or even better than the SVM. But the features in D_l are selected randomly from the whole high-dimensional feature space, which may result in D_l contains many features uninformative to classification, thus affecting the classification performance of the base learners. We should improve it to achieve higher performance.

III. The CS-RS Algorithm

Theory and experiments show that [14-17], increasing the performance of the base learners and the diversity between them helps to improve the generalization performance of the ensemble classifier. Therefore, how to improve the performance of base classifiers, while enhancing the diversity between them is the main goal of this study. The main idea of the proposed CS-RS is: 1) firstly, the chi-square statistic is adopted to compute the weight of each feature in original feature space. The weights are treated as the classification ability of the features, the higher the weight, the more informative the feature to the classification; 2) the features in original feature space are sorted from high to low according to weights and the sorted original feature space is partitioned into two parts according to a given dividing point: high correlation part and low correlation part. Then each part is assigned a sampling rate, the feature subset is formed by selecting features randomly in each part according to the given sampling rate, by this way, we can guarantee good performance of base learners, while ensuring diversity; 3) train base classifiers on feature subsets using bootstrap sample and combine their individual decisions to form the final class predictor.

A. Computation of Feature Weight using Chi-Square Statistic

The chi-square test [18] is a kind of hypothesis test method used widely. It is used to analysis the correlation of two variables. The basic idea of the chi-square test is to calculate the difference between the expected frequency and the actual frequency, denoted as the chi-square statistic. The larger the value of the chi-square statistic, the stronger the correlation of the two variables is.

Let Y be the class with 2 distinct class labels y_j ($j=1,2$). For the purposes of our discussion we consider the lth-dimensional feature F with P distinct values in training set D of dimension d, $l \in \{1,...,d\}$. We denote the distinct features by f_i for $i=1,\ldots,p$. Considering all combinations of the distinct features of F and the labels of Y, we can obtain a contingency table[19] of F against Y as shown in Table Ⅰ. In Table Ⅰ, the symbol val_{ij} stands for the number of the subset of D satisfying the condition that $F=f_i$ and $Y=y_j$.

TABLE Ⅰ.
CONTINGENCY TABLE OF FEATURE F AND CLASS Y

	$Y=y_1$	$Y=y_2$	*Total*
$F=f_1$	val_{11}	val_{12}	$val_{1.}$
…	…	…	…
$F=f_i$	val_{i1}	val_{i2}	$val_{i.}$
…	…	…	…
$F=f_p$	val_{p1}	val_{p2}	$val_{p.}$
Total	$val_{.1}$	$val_{.2}$	val

As shown in Table Ⅰ, the far right column contains the marginal totals for feature F:

$$val_{i.} = \sum_{j=1}^{2} val_{ij} \qquad i = 1,...,p \tag{1}$$

The bottom row is the marginal totals for class Y:

$$val_{.j} = \sum_{i=1}^{p} val_{ij} \qquad j = 1,2 \tag{2}$$

The grand total (the total number of samples) is in the bottom right corner:

$$val = \sum_{j=1}^{2}\sum_{i=1}^{p} val_{ij} \tag{3}$$

The weight of the feature F is computed as:

$$corr_{cs}(F,Y) = \sum_{i=1}^{p}\sum_{j=1}^{2} \frac{(val_{ij} - t_{ij})^2}{t_{ij}} \tag{4}$$

Where val_{ij} is the actual frequency from the contingency table and t_{ij} is the expected frequency computed as:

$$t_{ij} = \frac{val_{i.} \times val_{.j}}{val} \tag{5}$$

Supposed training set D has d-dimensional features $F_1, F_2, \ldots, F_d$, to ensure the sum of the feature weights is equal to 1, we compute the normalized weight of each feature as:

$$w_i = \frac{\sqrt{corr_{cs}(F_i,Y)}}{\sum_{i=1}^{d}\sqrt{corr_{cs}(F_i,Y)}} \qquad (1 \le i \le d) \tag{6}$$

The feature with larger w_i has a stronger power in predicting the classes of samples.

B. Improved Algorithm

We modify *Kodovsky's algorithm* [11] using the new sampling selection method to form feature subspaces.

Denote d as the dimensionality of the high-dimensional feature space, d_{sub} for the dimensionality of the feature subspace on which each base learner operate, L the number of base learners and N^{trn} the number of training samples from each class. The improved algorithm is called as CS-RS algorithm, described as follows:

Algorithm 1: CS-RS Algorithm

```
1. Input:
 -F: the feature space $\{F_1,\ldots,F_d\}$,
 -$d_{sub}$: the size of subspaces,
 -L: the number of base learners,
 -dp: the dividing point,
 -p: the sampling rate,
 -x:the test sample
2. For i=1 to d do
   Calculate the weight of the ith-dimensional feature $w_i$ using formula (6).
3. End for
4. Sort the original feature space according to the weight of each feature.
5. Divide the sorted feature space $F'$ into two parts according to the dividing point dp: [ $F_1'$ , $F_{dp}'$ ] and [ $F_{dp+1}'$ , $F_d'$ ].
6. For i=1 to L do
   (1) Draw a bootstrap sample $N_i^b$ , | $N_i^b$ |=$N^{trn}$ by uniform sampling with replacement from the training set $\{1,\ldots, N^{trn}\}$;
   (2) Select features randomly in each part according to the given sampling rate p to form subset $D_i$, $|D_i|= d_{sub} << d$;
   (3) Train a base learner $B_i$ on features $X_i= \{x_m^{(D_i)}\}_{m \in N_i^b}$ ;
   (4) Project x onto the feature subset $X_i$ and obtain the testing sample subset $x_i$ ;
   (5) Predict $x_i$ using $B_i$ and obtain the prediction $B_i(x_i)$ , $B_i(x_i) \in \{0,1\}$.
7. End for
8. Obtain the final prediction of x by majority voting:
```

$$B(x) = \begin{cases} 1 & when \sum_{i=1}^{L} B_i(x_i) > L/2 \\ 0 & when \sum_{i=1}^{L} B_i(x_i) < L/2 \\ random & otherwise \end{cases}$$

The above algorithm has the following characteristics:

(1)The diversity between base learners is obtained by training them on bootstrap samples and the randomization in the feature subspace generation.

(2)The classification accuracy of base classifiers is improved, because the probability for informative features to be included in each subspace is increases. It is suggested that the feature subset can contain more features with stronger classification ability to improve performance.

(3)The flexibility of forming the feature subspace can be effectively enhanced by setting different sampling rate in corresponding parts to adjust dynamically the distribution of features of different parts in the random subspace for each base learner.

IV. Experiment Results and Analysis

A. Experimental Setting

We demonstrate the power of the proposed CS-RS algorithm by applying it to the recently proposed adaptive spatial-domain steganographic algorithm called HUGO. All experiments were carried out on the database BossBase v1.00 (http://www.agents.cz/boss/BOSSFinal/).This training database is made of 10000 512×512 greyscale cover images in the pgm format, and the same 10000 images embedding a message at 0.4 bpp with HUGO algorithm with default parameters. 9000 covers and the associated 9000 stegos are selected randomly as testing set and another 2000 images as a training set. On each image, we extract 12753-dimensional SRM feature set [20] for the experiments.

The program code are implemented using C/C ++language, the test platform is WIN7 operating system, Intel Xeon E5300 2.60GHz, 8GB Memory.

B. Analysis of Experimental Results

In order to show the advantages of the new generation method of feature subset, we compare our proposed method and *the Kodovsky's method*. The dimensionality of the subspace can significantly influence the performance of the ensemble classifier. To evaluate the performance of our proposed method in a fair and reasonable way, we ran both algorithms against different sizes of feature subspaces. The dimensionality of the subspace is 100、150、200、250、300、350、400、450、500、600、700、800、900、1000、2000、5000,respectively. Due to the computational resource limitation, for a given subspace size we set the number of base learners L=71.

For our CS-RS algorithm, we set the dividing point dp=5000, the sampling rate p={0.5,0.5}、{0.6,0.4}、{0.7,0.3}、{0.8,0.2}, respectively. To facilitate the description, our method is described as CS-RS_x.y, Where x represents the sampling rate in the first part and y represents the sampling rate in the second part. For example, the CS-RS_6.4 represents the sampling rate in the first part is 60% and the sampling rate in the second part is 40%.

For each algorithm, there are totally 16 sets of experiments carried out in this paper. To make the results statistically reliable, each set of experiment has been repeated for 10 times independently to take the average value of detection accuracy、true positive rate and true negative rate as the final results, as shown in TableⅡ. The best results are marked in bold. *Tp* represents the true positive rate, *Tn* represents the true negative rate and *T* represents the detection accuracy. From the results shown in TableⅡ, we observe that:

(1) In the feature subspaces with less than 1000 dimensionality, the proposed CS-RS algorithm obtains an average detection accuracy of 83.92% which increases the average detection accuracy of *the Kodovsky's ensemble classifier* of 2.6%. Compared with *the Kodovsky's algorithm*, the CS-RS algorithm can get the optimal value in lower dimensional feature subspace, the detection accuracy of the CS-RS is more than 85% with d_{sub}=300 while *the Kodovsky's algorithm* gets the optimal value of 84.98% with d_{sub}=500. We also can see, the CS-RS algorithm gets the highest detection accuracy of 85.97% which increases the optimal value of *the Kodovsky's ensemble classifier* of 1.2% and gets the average detection accuracy of all subspaces of 82.84% which increases the average detection accuracy of *the Kodovsky's ensemble classifier* of all subspaces of 2.2%. But when d_{sub}＞1000, with the increase of dimensionality, the advantages of our algorithm is weakened, even though the performance still gets a small improvement with an average detection accuracy=77.26%. When d_{sub}=*5000*, the performance of two algorithms is very similar. The reason for that is our algorithm greatly increases the possibility of informative features to be selected, which significantly improves the classification ability of base learners in low dimensionality feature subspace and the diversity between them is increased due to the randomization in the feature subspace generation, thus, the performance of our algorithm is greatly improved. Then with the increase of dimensionality, the classification ability of base learners is still improved, but the diversity of base learners is declined, therefore, the performance of our ensemble is weakened.

(2) For the proposed algorithm, increase appropriately the proportion of the features in the first part can extend the subspace's classification ability to improve the fusion decision. But if select too much features with higher classification ability, the accuracy of ensemble classifier will decline. This further indicates that we should make a trade-off between diversity and accuracy of the base classifier. From TableⅡ, it can be seen when p is {0.7, 0.3}, the detection accuracy is the best on the whole, which provides the basis for feature selection in the practical application.

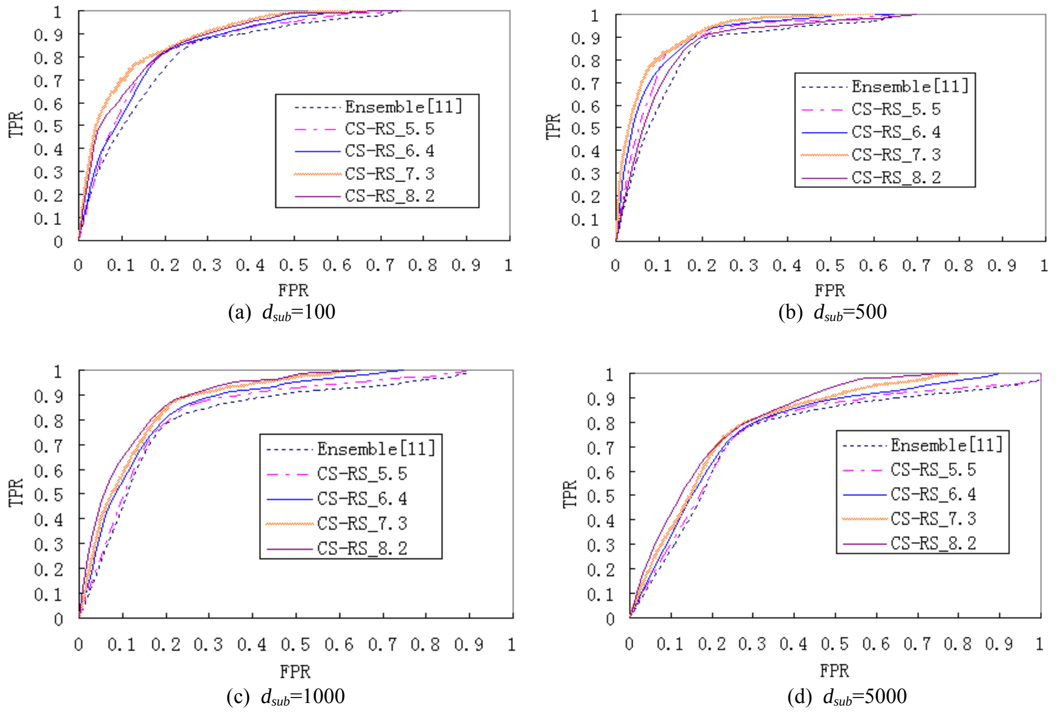
(3) As is already apparent from TableⅡ, the detection accuracy of both algorithms first increases and then decreases with the increase of dimensionality of feature subset, which is mainly because of the following two reasons. First, the redundancy between features will increase with the growing dimension sizes of subsets, the individual base learners become more dependent and thus

the ability of the ensemble classifier to form non-linear boundaries decreases. Second, the base classifiers start to suffer from overtraining as the subspace dimensionality increases while the training size remains the same.

In order to visualize the performance of the algorithms, the Receiver Operating Characteristic (ROC) curve is used when comparing both methods. Fig. 1 plotted the curves in different dimensions of the subspaces, which are 100, 500, 1000 and 5000 respectively. The great advantages of our proposed algorithm in detection accuracy can be obviously seen from Fig.1.

TABLE II
DETECTION ACCURACY OF METHODS AGAINST HUGO

d_{sub}	CS-RS_5.5			CS-RS_6.4			CS-RS_7.3			CS-RS_8.2			Ensemble[11]		
	Tp	*Tn*	*T*	*Tp*	*Tn*	*T*	*Tp*	*Tn*	*T*	*Tp*	*Tn*	*T*	*Tp*	*Tn*	*T*
100	80.54	81.46	81.00	81.31	80.69	81.00	81.95	80.53	**81.24**	81.73	80.50	81.11	83.87	75.35	79.61
150	84.45	80.05	82.25	82.02	80.88	81.45	84.09	80.71	**82.40**	83.18	80.82	82.00	80.88	80.35	80.61
200	85.12	80.76	82.94	87.69	79.93	**83.81**	87.41	79.83	83.62	87.52	79.56	83.54	84.41	80.28	82.35
250	83.97	82.85	83.41	86.78	81.46	84.12	87.31	81.59	**84.45**	86.18	81.65	83.91	84.66	78.78	81.72
300	86.73	84.05	85.39	86.45	84.71	**85.58**	88.08	82.70	85.39	87.09	83.58	85.33	84.08	81.64	82.86
350	85.93	85.05	85.49	86.91	84.77	85.84	87.12	84.82	**85.97**	85.62	84.97	85.30	83.98	80.41	82.19
400	84.46	86.12	85.29	87.15	83.55	**85.35**	85.98	84.30	85.14	87.39	82.79	85.09	84.08	83.74	83.91
450	83.61	80.71	82.16	82.33	81.57	81.95	86.35	78.53	**82.44**	85.97	85.64	80.80	83.55	81.15	82.35
500	84.22	86.96	85.59	88.14	83.18	**85.66**	82.65	88.67	**85.66**	86.12	84.84	85.48	87.41	82.56	84.98
600	82.51	87.11	84.81	86.28	83.98	85.13	87.16	83.40	**85.28**	86.43	82.94	84.68	83.34	81.95	82.64
700	85.57	85.09	85.33	86.52	84.60	85.56	85.99	85.23	85.61	86.09	85.65	**85.82**	84.55	78.08	82.31
800	84.25	81.91	83.08	83.34	83.18	83.26	85.18	83.12	**84.15**	83.98	84.16	84.07	79.65	78.75	79.12
900	81.69	79.91	80.80	81.67	80.75	81.21	80.99	82.53	81.76	83.46	80.54	**82.00**	82.02	77.84	79.93
1000	77.12	81.34	79.23	79.57	80.73	80.15	83.61	80.39	82.00	82.78	81.69	**82.23**	81.42	76.88	79.15
2000	81.41	76.59	79.00	78.76	78.36	78.56	82.11	78.57	**80.34**	82.15	78.52	80.33	78.52	76.95	77.73
5000	74.59	74.67	74.63	75.92	73.52	74.72	73.51	77.15	75.33	77.26	74.32	**75.79**	75.67	73.45	74.56



(a) d_{sub}=100 (b) d_{sub}=500

(c) d_{sub}=1000 (d) d_{sub}=5000

Figure 1. ROC curve of different algorithms in different dimensions of the subspace

We also compare the training time of the CS-RS algorithm and *the Kodovsky's algorithm* for different d_{sub}. The experiment is repeated 20 times, and we calculate the average value of the training time as the final result. The training time of the CS-RS algorithm is the average value of the CS-RS algorithm with different sampling rate. The comparison is reported in TableIII. This experiment reveals that although our algorithm needs to compute the feature weights, there is little difference between the training times of both algorithms. The CS-RS algorithm can offer training complexity comparable to *the Kodovsky's classifier*.

TABLE III
TRAINING COMPLEXITY OF BOTH ALGORITHMS

d_{sub}	Ensemble[11]	CS-RS
100	0.380min	0.457 min
150	0.397 min	0.532 min
200	0.428 min	0.587 min
250	0.465 min	0.617 min
300	0.490 min	0.648 min
350	0.515 min	0.670 min
400	0.545 min	0.695 min
450	0.588 min	0.732 min
500	0.622 min	0.753 min
600	0.648 min	0.802 min
700	0.682 min	0.855 min
800	0.723 min	0.873 min
900	0.802 min	0.930 min
1000	0.893 min	1.003 min
2000	1.730 min	1.942 min
5000	5.247 min	5.590 min

Next we investigated the impact of the number of base learners *L* on our algorithm and compared the results with those of *the Kodovsky's algorithm*. Fig.2 shows the detection accuracy on the testing set of our algorithm as a function of the number of fused base learners *L*, for 5000 as the dividing point and {0.8,0.2} as the sampling rate. Fig.3 shows the detection accuracy on the testing set of *the Kodovsky's algorithm* as a function of the number of fused base learners *L*. The performance of the two algorithms for different subspace (d_{sub}=200, 500 and 1000, respectively) is shown in Fig.2 and Fig.3. Obviously, the classification accuracy of both algorithms quickly saturates with *L*. The classification accuracy is tending towards stability and does not change seriously when *L*>71. For the fastest performance, one should choose the smallest *L* that gives satisfactory performance. We suggest the number of base learners *L* =71.

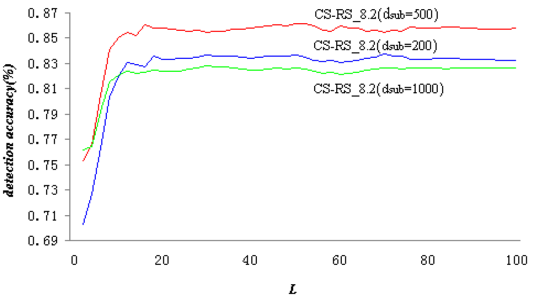


Figure 2. Detection accuracy saturates with the number of base learners *L*. Target algorithm:CS-RS with sampling rate {0.8,0.2}

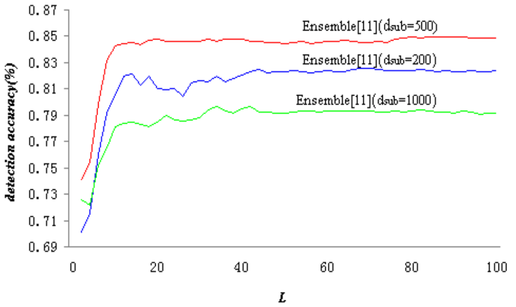


Figure 3. Detection accuracy saturates with the number of base learners *L*. Target algorithm: Ensemble [11]

The features in first part (high correlation part) have stronger classification ability. The dividing point *dp* adjust the proportion of the features in first part selected in random subspace. Obviously, increasing appropriately the proportion of the features in first part selected in the random subspace can improve the classification ability of the ensemble. But when the value of *dp* increases to a certain extent, the accuracy of ensemble classifier will decline because of the weaken diversity. Fig.4 gives the curves of the detection accuracy with respect to the values of parameter *dp* in the case of CS-RS with the number of base learners *L* = 71, subspace dimensionality d_{sub} = 300 and sampling rate *p*= {0.5, 0.5}, {0.7, 0.3} respectively. It can be seen that the detection accuracy varies with different *dp*. The detection accuracy first increases and then tends to be stable, and finally decreases with the value of *dp* increasing from small to large. When *dp*= 10000, the performance of the CS-RS algorithm and *the Kodovsky's algorithm* is vary similar. It can be seen that the highest detection accuracy always occurs when *dp*∈ {3000,5000}. For gaining good steganalysis performance in various application scenarios, we suggest *dp*= 5000 as the default value.

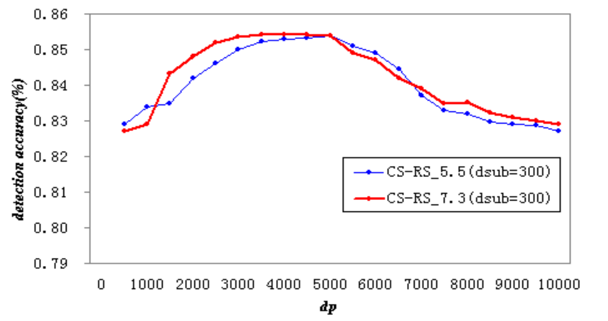


Figure 4. Detection accuracy varies with different dividing point *dp*

According to the theory of ensemble classifier [14], the individual base learners have to be sufficiently diverse in the sense that they should make different errors on unseen data. The proportion mechanism to select features at each part according the sampling rate is to increase the diversity between them. To verify the effectiveness of the sampling rate used in our algorithm, we compare our algorithm, *the Kodovsky's algorithm* and *the CS-RS (part1) algorithm* which only selects features from the first part to form feature subspace for different d_{sub}. The comparison is shown in Fig.5. From it we can see: (1) The performance of the CS-RS algorithm with different sampling rate is better than *the CS-RS (part1) algorithm* in all subspaces except d_{sub}=450 and 800. (2) *The CS-RS (part1) algorithm* gets better performance than *the Kodovsky's algorithm* in 10 out of 16 subspaces with different dimensionality and provides performances equivalent to *the Kodovsky's algorithm* in 3 subspaces. In two subspaces(d_{sub}=900 and 5000) *the CS-RS (part1) algorithm* gets the worst results while our CS-RS algorithm gets optimal results in all subspaces with different dimensionality. The results indicate using the sampling rate in our algorithm can improve the diversity between base learners.

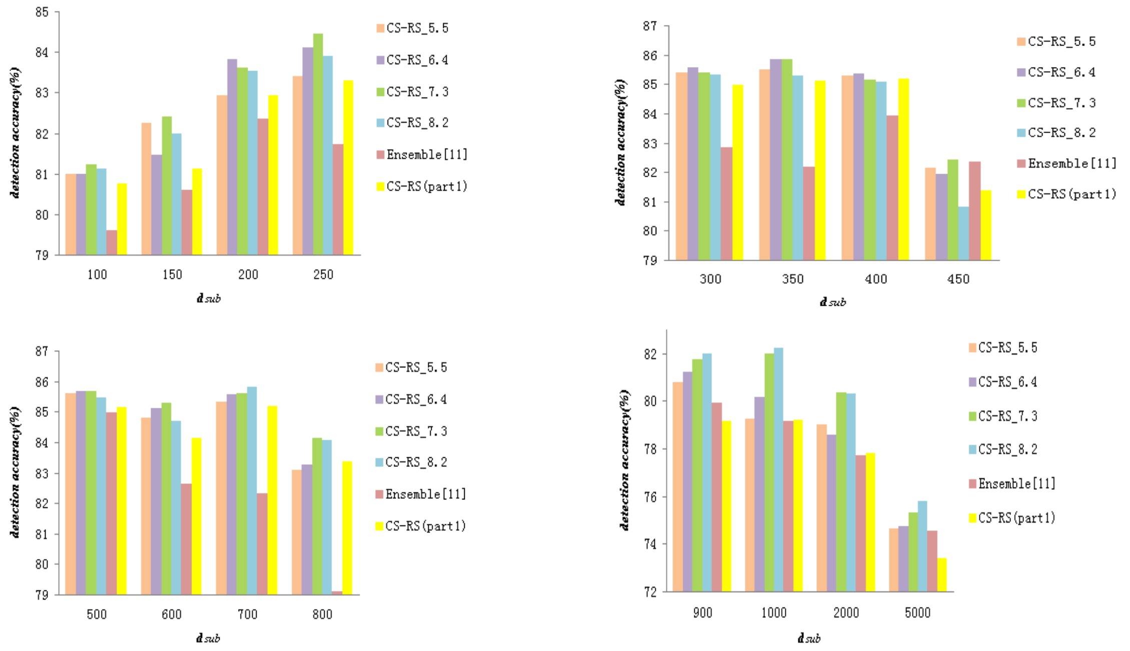


Figure 5. Comparison of the detection accuracy of different algorithms for different feature subspace

To assess whether our algorithm can be competitive in terms of accuracy and cpu time with L-SVM and other popular ensemble algorithms, namely Bagging and AdaBoost when a high-dimensional feature space is used for steganalysis, the performance of our algorithm with d_{sub}=5000, *dp*=5000 and the number of base learners L = 71 is contrasted with Bagging, AdaBoost and L-SVM with 5000-dimensional feature set extracted from SRM in Table Ⅳ. We use FLD as base learner, the parameters of Bagging and AdaBoost are kept at their default values in WEKA, the training parameters of L-SVM are obtained using five-fold cross-validation to search over the grid of the cost parameter $C \in \{10^{a}\}$, $\alpha \in \{-4,...,3\}$,which is recommended in [11].The results in Table Ⅳ indicate, the proposed CS-RS outperforms Bagging and AdaBoost, increasing the performance of Bagging and AdaBoost more than 10% with much lower training complexity. The performance of our algorithm is similar to or even better than L-SVM, but the time required for training of our algorithm is a litter higher than L-SVM due to computing the weights of features.

TABLE Ⅳ
COMPARISON OF DIFFERENT ALGORITHMS IN TERMS OF TRAINING COMPLEXITY AND PERFORMANCE

Classifier	Detection accuracy	Training time
Bagging	66.42%	60.27 min
AdaBoost	63.64%	63.15 min
L-SVM	74.38%	4.40 min
CS-RS_5.5(d_{sub}=5000)	74.34%	5.24 min
CS-RS_6.4(d_{sub}=5000)	74.72%	5.93 min
CS-RS_7.3(d_{sub}=5000)	75.33%	5.58 min
CS-RS_8.2(d_{sub}=5000)	75.79%	5.61 min

V. CONCLUSIONS

The current trend in steganalysis is to train classifiers on feature space with increasing higher dimensionality to obtain more accurate and robust detectors. We presented a new CS-RS algorithm that incorporates the new subspace generation method. In the proposed algorithm, the feature subset can contain more informative features, while ensure the randomization of features by the randomly selection, thus, strengthen the accuracy of the base learners, without sacrificing diversity between them. To be specific, the proposed method obtained a feature space sorted by weights of features and partitioned it into two parts, then formed feature subset by selecting features randomly from each part according to a given sampling rate. Experimental results show that the proposed scheme can effectively defeat the HUGO steganographic methods and its performance is better than that of existing steganalysis approaches. The future study will focus on how to implement the new algorithm in parallel in a distributed environment, significantly reducing the time for creating an ensemble model from large data.

ACKNOWLEDGMENT

This study was supported by the Technology Innovation Platform Project of Fujian Province under Grant No.2009J1007, the Natural Science Foundation of Fujian Province under Grant No. 2010J01331.

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

REFERENCES

[1] J.Kodovský, J.Fridrich, "Steganalysis in high dimensions: Fusing classifiers built on random subspaces," In IS&T/SPIE Electronic Imaging. International Society for Optics and Photonics,USA: California, 2011, pp. 78800L-78800L.

[2] T.Pevny, P.Bas, J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," information Forensics and Security, IEEE Transactions, vol.5, pp.215-224, March 2010.

[3] J.Kodovský, T.Pevný, J.Fridrich, "Modern steganalysis can detect YASS," In IS&T/SPIE Electronic Imaging,vol.7541, pp.754102-754102, January 2010.

[4] A.Sarkar,K.Solanki,B.S.Manjunath, "Further study on YASS: Steganography based on randomized embedding to resist blind steganalysis," Electronic Imaging 2008,vol.6819, pp.16-31, January 2008.

[5] P.Bas, T.Filler, T.Pevný, "" Break Our Steganographic System": The Ins and Outs of Organizing BOSS," Information Hiding,Berlin: Heidelberg, 2011,pp.59-70.

[6] T.Pevný, T.Filler, P.Bas, "Using high-dimensional image models to perform highly undetectable steganography," Information Hiding, Berlin: Heidelberg, 2010, pp.161-177.

[7] O.Maimon, L.Rokach, "Improving supervised learning by feature decomposition," Foundations of Information and Knowledge Systems, vol.2284, pp.178-196,February 2002.

[8] K.Fukunga, Introduction to statistical pattern recognition. MacMillan Publishing Company, San Diego,1990.

[9] L.Molina, L.Belanche, A.Nebot, "Feature selection algorithms: A survey and exp. evaluation," in Proc. ICDM, 2002, pp. 306-313.

[10] D.Fradkin, D.Madigan, "Experiments with random projections for machine learning," Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, New York: ACM, 2003, pp. 517-522.

[11] J.Kodovsky, J.Fridrich , V.Holub, "Ensemble classifiers for steganalysis of digital media," Information Forensics and Security, IEEE Transactions on,vol.7,pp. 432-444, June 2012.

[12] R.O.Duda, P.E.Hart, D.G. Stork, Pattern classification, John Wiley & Sons, 2012.

[13] C.Chih-Chung and L.Chih-Jen, "LIBSVM: A library for support vector machines," ACM Transactions on Intelligent Systems and Technology, vol. 2, pp.27:1–27:27,April 2011.

[14] R.Polikar, Ensemble learning, Ensemble Machine Learning,US: Springer,2012.

[15] H.Tao, X.Ma, M.Qiao, "Subspace Selective Ensemble Algorithm Based on Feature Clustering," Journal of Computers,vol.8,pp.509-516, February 2013.

[16] D.Zou, Z.He, Y.Yan, "Using an Ensemble of Support Vector Machine Classifiers to Predict Protein Supersecondary Structural Motifs," Journal of Computers,vol.6,pp.2053-2059, October 2011.

[17] J.Liu, G.Xu, D.Xiao, "A Semi-supervised Ensemble Approach for Mining Data Streams," Journal of Computers,vol.8,pp.2873-2879, November 2013.

[18] J.Illian, A.Penttinen,H.Stoyan,Statistical analysis and modelling of spatial point patterns,John Wiley & Sons, 2008.

[19] K.Pearson, On the theory of contingency and its relation to association and normal correlation,UK: Cambridge University Press,1904.

[20] J.Fridrich, J.Kodovsky, "Rich models for steganalysis of digital images," Information Forensics and Security,vol.7,pp.868-882, January 2012.

Fengying He She is currently the Lecturer in the College of Mathematics and Computer Science, Fuzhou University of China. She was born in1979. Her current research interests include machine learning, pattern recognition and multimedia security.

Shangping Zhong He received his PhD in Computer Science and Technology from Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005. He is currently a professor in the College of Mathematics and Computer Science, Fuzhou University, China. He was born in 1969. His current research interests include machine learning, pattern recognition and multimedia security.

Kaizhi Chen He is currently a Lecture in the College of Mathematics and Computer Science, Fuzhou University, China. He was born in 1983.His current research interests include machine learning, pattern recognition and multimedia security.