https://doi.org/10.2352/ISSN.2470-1173.2017.7.MWSF-320

The A Priori Knowledge Based Secure Payload Estimation for Additive Model

Sai Ma, Xianfeng Zhao, Qingxiao Guan, Chengduo Zhao
State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences,
Beijing 100093, China
School of Cyber Security, University of Chinese Academy of Sciences,
Beijing 100093, China

Abstract

In this paper, we propose a practical method to estimate the payload rate for individual cover before stego embedding. The proposed method is adopted to the additive distortion model. The a priori knowledge functions employed in the method contains the relation function of steganalyzer's detection error and stego distortion ($P_E - D$), and the relation function of payload rate and distortion($D - \alpha$) of the given cover. As it is not suitable to measure the stego security with stego distortion, we adopt P_E as the security metric. With the sender's expected P_E, the role of $P_E - D$ function is to calculate the corresponding D, and then sender can solve out his expected α with $D - \alpha$ function for the cover. The $P_E - D$ function is acquired before estimating phase. During the estimating, the most time-consuming part is calculating the $D - \alpha$ function for the cover, which costs 1 time of stego embedding. Our method is an efficient solution for estimating the secure payload rate.

Introduction

Steganography is the technology of covert communication, which tries to arouse no suspicion during communication. Digital image is one of the most widely-used media in steganography. At the very beginning, the digital image steganography is designed for resisting visual detection. To effectively detect the stego image, statistical method is developed, which can figure out the statistical difference between natural image and stego image. The state-of-the-art steganalysis method is extracting the well-designed stego feature of the image and adopting the machine learning method to detect the stego image. The high dimensional rich model steganalysis feature[1][2] has an appreciable accuracy of detection. To enhance the security of stego communication, the designer of stego algorithm needs to minimize the statistical detectability of stego image.

Historically, there is a strategy named as Model Preservation or Model Based[3]. It preserves the statistical model of image in order to minimize the statistical detectability. However, the method has a disadvantage that the model is chosen, which can be attacked via higher order statistical model steganalysis as so-called "beyond the model". The prevailing stego strategy nowadays is adaptive method. It employs a special coding scheme called syndrome trellis code(STC)[4] to embed the secret message while minimizing the stego distortion. Such approach avoids modeling the cover image, which overcomes the defect of Model Preservation. The additive distortion model is a reasonable simplification which makes adaptive method practical. The cost value assigned to every cover element evaluates the statistical impact introduced by modification. In additive distortion model, the distortion of stego image is sum of cost values of changed image elements, which reflects the statistical detectability of stego image.

The fundamental models of adaptive distortion-minimizing steganography are payload limited sender (PLS) model and distortion-limited sender (DLS) model[5]. They are complementary tasks. The PLS model is to embed the fixed-length message in the cover source and at the same time minimize the stego distortion. For DLS model, its task is to communicate as more information as possible while not exceeding the given stego distortion.

On passive-warden channel, if sender focuses on the security of the communication rather than efficiency, DLS is more valuable. We name the task that estimating the maximum payload length not exceeding the sender's expected security metric as secure payload estimation (SPE) problem. It is clear that the SPE is within the framework of DLS. However, it is not convenient for sender to measure the stego security with distortion, so it is needed to define a security metric, and establish a mapping of distortion and security. A feasible choice is detection error rate of the steganalyzer(P_E), i.e., the out-of-bag error(E_{OOB}) of ensemble classifier[6]. In 2015, Zhang et al. proposed a SPE method[7] which adopts the detection error rate as the stego security metric. The method uses a texture complexity metric called CO-occurrence Matrix Entropy Difference (COMED) to evaluate the stego security of an image. The measurement of COMED is divided into several levels. In each COMED level, it is needed to fit a function of COMED and P_E. However, this method needs to storage $P_E - COMED$ function for every COMED level, which is complicated.

In this paper, we propose a practical SPE method for individual cover. The method estimates the payload rate based on the a priori knowledge. We adopt steganalytic classifier's error rate as stego security metric. We establish the relation function of security metric and stego distortion ($P_E - D$ function). The $P_E - D$ function is generated from training. With the relation function, the sender can calculate the corresponding stego distortion of his expect security metric. To estimate the secure payload for a particular cover image with the calculated distortion, the sender first conducts an stego embedding on the cover to resolve the covers relation function of distortion and payload rate($D - \alpha$ function), then computes the payload from the $D - \alpha$ function. Both $P_E - D$ function and $D - \alpha$ function are called a priori knowledge function. Our work is valuable under DLS scenario to estimate the message length for individual cover.

Preliminaries

Notations

Capital boldface symbol $\mathbf{X}$ represents matrix, and the capital normal symbol with subscript X_{ij} denotes the element of matrix $\mathbf{X}$ in the position (i, j). The boldface lowercase symbol $\mathbf{v}$ represents the vector.

The symbols $\mathbf{X}$ and $\mathbf{Y}$ denote the cover image and stego image, respectively. They will always be 8-bit grayscale spatial images in this paper, so the dynamic range of the pixel is 0 to 255. The symbols $\{\mathbf{X}\}$ and $\{\mathbf{X}\}$ denote cover image library and stego image library, respectively. The function $D(\mathbf{X},\mathbf{Y})$ represents the stego distortion caused by changing cover $\mathbf{X}$ to stego image $\mathbf{Y}$. In theoretical deduction, as the modification pattern is more focused, the stego distortion is abbreviated to $D(\mathbf{s})$, $\mathbf{s}$ denotes the modification pattern. For additive model, the cost value of change pixel X_{ij} to Y_{ij} is written as ρ_{ij}.

Additive Distortion Model

The additive distortion model is the common model in the application of adaptive steganography. It is the simplification of the real steganographic embedding procedure. The main assumption of additive model is that the statistical impact of the modification of image element is independent. So the additive distortion is defined as sum of cost values whose pixels are changed.

$$D(\mathbf{X},\mathbf{Y}) = \sum_{i,j} \rho_{ij}|X_{ij} - Y_{ij}| \tag{1}$$

PLS and DLS

Payload Limited Sender. The description of the task is as follows: with a given message $\mathbf{m}$, the sender needs to find a modification pattern $\mathbf{s}$ that has minimal distortion. To formalize the model, the optimization problem of PLS is to find a distribution $p(\mathbf{s})$ which has a minimal distortion expectation $E(D(\mathbf{s}))$ introduced by modification. The math expression is:

$$arg\min_{\mathbf{s}} \sum_{\mathbf{s}} D(\mathbf{s})p(\mathbf{s}), s.t. - \sum_{\mathbf{s}} p(\mathbf{s})\log(\mathbf{s}) = m \tag{2}$$

Distortion Limited Sender. The DLS has a fixed distortion D_ε. The sender attempts to embed the message to the greatest extent, while the distortion caused by modification does not exceed D_ε. The optimization form of the task is to determine distribution $p(\mathbf{s})$ that has a maximal entropy m with a given distortion:

$$arg\max_{\mathbf{s}} - \sum_{\mathbf{s}} p(\mathbf{s})\log(\mathbf{s}), s.t. \sum_{\mathbf{s}} D(\mathbf{s})p(\mathbf{s}) = D_\varepsilon \tag{3}$$

From expressions we can see that the two problems are dual to each other. We can use Lagrange multipliers to solve these two problems[8]. The solutions have an identical form:

$$p(\mathbf{s}) = Ae^{-\lambda D(\mathbf{s})} \tag{4}$$

which is called Gibbs distribution. A is the partition function, which $A^{-1} = \sum_{\mathbf{s}} e^{-\lambda D(\mathbf{s})}$. This form of solution is the optimal distribution of two problems. The scalar λ is the parameter determined by the constraint. For PLS, the constraint is the payload capacity $\mathbf{m}$, while for DLS, the constraint is distortion D_ε.

In additive model, the distortion can be transformed into sum of cost values. The optimal distribution (4) can be written as:

$$p(\mathbf{s}) = Ae^{-\lambda D(\mathbf{s})} = \prod_{i=1}^{n} A_i e^{-\lambda \rho(s_i)} \tag{5}$$

in the expression, $A_i^{-1} = \sum_{s_i} e^{-\lambda \rho(s_i)}$, which is the partition function of the single pixel.

Distortion Function

The distortion function is used for evaluate the impact caused by stego embedding. The distortion function which can be used in practical scenario should be additive. With additive model, the algorithm assigns a cost value to every image element based on the neighboring elements properties.

HUGO[9] is the first distortion model, which is based on the steganalytic feature SPAM[10]. Then, in 2013, Holub et al. proposed WOW[11] on spatial domain. It utilizes Daubechies-8 wavelet filter. Later, UNIWARD[12] was proposed, which is similar to WOW. In 2014, Li et al. proposed HILL[13], which makes the stego modification gathering in the complex-textured area.

In this paper, we employ S-UNIWARD to demonstrate the proposed method. To have a further understanding, here is a brief description of S-UNIWARD. The algorithm consists of two steps, which are filtering and distortion calculating.

Step 1: filtering. The filtering is to acquire the wavelet coefficients of cover image and stego image. The masks for filtering are constructed from one-dimensional high-pass and corresponding low-pass wavelet decomposition filters. We use $\mathbf{h}$ and $\mathbf{g}$ to denote high-pass filter and low-pass filter, respectively. There are three sub-band filter masks: HL filter $\mathbf{K}^{(1)}$, LH filter $\mathbf{K}^{(2)}$, and HH filter $\mathbf{K}^{(3)}$. They are constructed as following manner.

$$\mathbf{K}^{(1)} = \mathbf{h} \cdot \mathbf{g}^T, \mathbf{K}^{(2)} = \mathbf{g} \cdot \mathbf{h}^T, \mathbf{K}^{(3)} = \mathbf{h} \cdot \mathbf{h}^T \tag{6}$$

The filtering operation is as follows:

$$\mathbf{W}^{(k)}(\mathbf{X}) = \mathbf{K}^{(k)} \otimes \mathbf{X}, \mathbf{W}^{(k)}(\mathbf{Y}) = \mathbf{K}^{(k)} \otimes \mathbf{Y} \tag{7}$$

$\mathbf{W}^{(k)}$ is k-th sub-band of wavelet coefficient matrix, $k = 1,2,3$. The symbol $W_{uv}^{(k)}$ denotes the element of matrix $\mathbf{W}^{(k)}$ in the position (u, v).

Step 2: calculating. The cost value of a pixel is the relative change of the wavelet coefficients. The distortion function of $\mathbf{Y}$ is the sum of cost values.

$$D(\mathbf{X},\mathbf{Y}) = \sum_{k=1}^{3} \sum_{u=1}^{n_1} \sum_{v=1}^{n_2} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\sigma + |W_{uv}^{(k)}(\mathbf{X})|} \tag{8}$$

In the expression, σ is a constant. In the original version of S-UNIWARD, it is set to 1. To exclude the interference introduced by image size, in this paper, we adopt average distortion $\overline{D}(\mathbf{X},\mathbf{Y})$ instead of $D(\mathbf{X},\mathbf{Y})$. Note that image size is $n_1 \times n_2$:

$$\overline{D}(\mathbf{X},\mathbf{Y}) = \frac{1}{n_1 \times n_2} D(\mathbf{X},\mathbf{Y}) \tag{9}$$

Proposed Work

Stego Security Metric

To formalize the stego security, we define the security metric M_S. In the proposed method, we adopt steganalyzer's detection error rate as M_S. The common practice of evaluating the safety of stego algorithm or the power of steganalytic feature extractor is generating the stego images from standard image database, i.e., BOSSbase[14], and extracting the stego features, then training the ensemble classifier with these samples. The E_{OOB} of the classifier is an estimation of P_E. The P_E agrees with the designing intention of M_S, we define:

$$M_S = P_E \tag{10}$$

General Framework

Figure 1 is the general framework of proposed SPE algorithm. The algorithm consists of two parts: preparation part and estimation part. The preparation part is to generate the $P_E - \overline{D}$, which is based on the training with the image library. The estimation part is to estimate the secure payload rate for a particular cover.

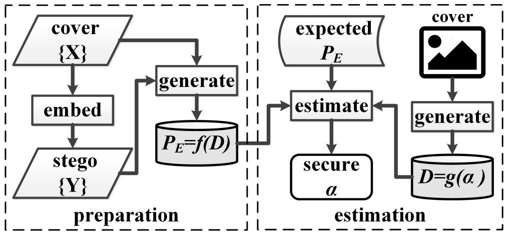

Figure 1. general framework of proposed method

A Priori Knowledge Function

The a priori knowledge of the proposed method consists of two parts, which are $P_E - \overline{D}$ function and $\overline{D} - \alpha$ function.

$P_E - \overline{D}$ Function

$P_E - \overline{D}$ function is the mapping function of security metric and stego distortion. It is universal to cover source. Distortion function is a reflection of detectability. The effect of stego payload is embodied by distortion. As mentioned above, the SPE problem is within the framework of DLS. Here, the P_E should be error rate of blind steganalysis with various payload rate.

Our method to establish the $P_E - \overline{D}$ function is cure fitting. So the main task is to generate enough valid data points $(\overline{D}, P_E)$. There are four steps, which are building image library, sampling sub library, generating training set, and fitting curve.

Step 1: building image library. In order to obtain a sufficient number of data points, we need to generate a large number of stego sample. Given an images library $\{\mathbf{X}\}$ containing L images and a group of payload rates $\{\alpha\}$ with M rates, we build a stego image library $\{\mathbf{Y}\}$ with $L \times M$ images though embedding random information via STC.

Step 2: sampling sub library. With cover image library $\{\mathbf{X}\}$ containing L images, we also need to choose L images from $\{\mathbf{Y}\}$ to form a sub image library $\{\mathbf{Y}\}_{sub}$ with M payload rates. We adopt a random sampling method that choose L elements from a set containing $L \times M$ elements. The sampling method is illustrated by Figure 2. The matrix in the Figure 2 represents the stego image library $\{\mathbf{Y}\}$ and every single grid is a stego image in the library. M is the number of payload rates and L is the number of images in $\{\mathbf{X}\}$. In the matrix, the pair (m, l) is the position of the grid, represents the stego image $\mathbf{Y}$ generated from l-th cover image in $\{\mathbf{X}\}$ with m-th payload rate in $\{\alpha\}$. The gray-colored grid indicates that the corresponding stego image is chosen. To ensure that images in $\{\mathbf{Y}\}_{sub}$ have M payload rates, we choose L/M stego images in every payload rate. The sampling order is showed in Figure 2. The order ensures that every cover image in $\{\mathbf{X}\}$ is able to form a feature pair with stego image and does not repeat at the same time. To generate more $\{\mathbf{Y}\}_{sub}$, we can randomly permute the image index l again, and re-conduct the sampling procedure. With the permutation, each time the sampled $\{\mathbf{Y}\}_{sub}$ is different from other $\{\mathbf{Y}\}_{sub}$. The purpose to generate multiple $\{\mathbf{Y}\}_{sub}$ is to obtain more data points for fitting the $P_E - \overline{D}$ curve.

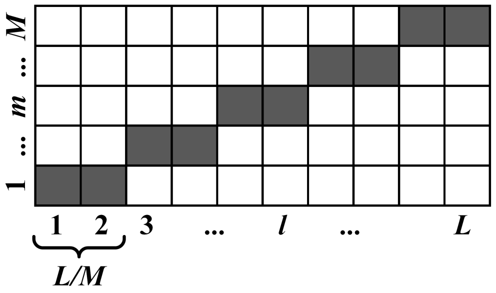

Figure 2. sampling stego images from $\{\mathbf{Y}\}$

Step 3: generating data points. Suppose that we are going to generate Q $(\overline{D}, P_E)$ data points to fit the function curve with $\{\mathbf{X}\}$ and $\{\mathbf{Y}\}_{sub}$ at one time, we need to accumulate corresponding Q training sets. We note that every instance in the training set is a steganalysis features pair $(\mathbf{v}^+, \mathbf{v}^-)$, in which the positive feature $\mathbf{v}^+$ is the feature of a stego image and negative feature $\mathbf{v}^-$ is the feature of the stego image's cover source.

In each training set, the distortion values of corresponding stego images are distributed within a interval. The mid-value of the interval is $\overline{D}_{mid}$. To build a training set with a mid-value distortion $\overline{D}_{mid}$, we first set an interval $[\overline{D}_{down}, \overline{D}_{up})$, then find out every training instance whose stego image distortion value is within the interval. In $[\overline{D}_{down}, \overline{D}_{up})$, $\overline{D}_{down} = \overline{D}_{mid} - \Delta\overline{D}$ and $\overline{D}_{up} = \overline{D}_{mid} + \Delta\overline{D}$. $\Delta\overline{D} = (\overline{D}_{max} - \overline{D}_{min})/2Q$, in which $\overline{D}_{max}$ is the maximum distortion value in $\{\mathbf{Y}\}_{sub}$ and $\overline{D}_{min}$ is the minimum value in $\{\mathbf{Y}\}_{sub}$. After split $[\overline{D}_{min}, \overline{D}_{max}]$ into Q sub-intervals, we can extract the stego feature of images in $\{\mathbf{Y}\}_{sub}$ and $\{\mathbf{X}\}$, then sort these features into Q training sets according to the distortion.

Assume that we sample R sub libraries from $\{\mathbf{Y}\}$, and for each $\{\mathbf{Y}\}_{sub}$ we can generate Q points $(\overline{D}, P_E)$, we will obtain $R \times Q$ identical data points totally.

Step 4: fitting curve. After generating data points, we can fit the $P_E - \overline{D}$ function's curve $P_D = f(\overline{D})$. Note that the valid interval for estimation is $[\overline{D}_{min}, \overline{D}_{max}]$, the function f should have following two properties:

property 1: monotonicity.

$$\forall x_1 < x_2 \in [\overline{D}_{min}, \overline{D}_{max}], f(x_1) > f(x_2) \tag{11}$$

property 2: slow-attenuation.

$$\lim_{x\to+\infty} f(x) = 0 \tag{12}$$

The detection error or the security metric should decrease as the stego distortion increasing. Property (11) ensures that f is available for estimation. Property (12) ensures that f has a large interval for estimation, while the $\overline{D}$ of various kind of image may distributed in a large range. In this paper we recommend sigmoid function or gaussian function as f's model.

$\overline{D}-\alpha$ *Function*

The $\overline{D}-\alpha$ function is another a priori knowledge adopted in proposed method. The function reflects the property of individual cover. With optimal embedding, the $\overline{D}-\alpha$ function is the performance limitation of practical embedding method, which is also called rate-distortion bound.

In the proposed work, to estimate an α for cover $\mathbf{X}$ under a expected P_E, we need to calculate the $\overline{D}-\alpha$ function of the cover $\mathbf{X}$. As digital image is a kind of complex high-dimensional signal, it is hard to calculate the closed-form expression of $D-\alpha$ function. Alternatively, like P_E-D function, we can fit the function with data points $(\alpha,\overline{D})$. To obtain the data points, we can choose some payload rates α and embed the random information in the cover, then calculate the $\overline{D}$ introduced by the embedding. As the proposed method is a practical method, we adopt STC instead of optimal embedding.

We find that the power model is a good model to fit the $\overline{D}-\alpha$ function:

$$\overline{D} = k \times \alpha^{b} \tag{13}$$

In the expression, k,b is the parameter to be determined. The value of power index b is less than 2. To simplify the fitting procedure, we set $b=1$. With the linear model, we only need to embed once, and need not fit the function:

$$\overline{D} = k \times \alpha \tag{14}$$

Here we need to solve the parameter k. With a data point $(\alpha',\overline{D}')$, $k = \overline{D}' \setminus \alpha'$.

Algorithm

As illustrated in Figure 1, the procedure of proposed work can be divided into two parts, which are preparation part and estimation part. The mission of preparation part is to establish the $P_E-\overline{D}$ function. The estimation part is to estimate the secure payload rate $\hat{\alpha}$ for a given cover $\mathbf{X}$ according to sender's expecting security metric $\hat{P_E}$. We note that the stego embedding algorithm is $Emb(\mathbf{X},\alpha)$, stego feature extracting algorithm is $Ext(\mathbf{X})$, and classifier training algorithm is $Trn(\{(\mathbf{v}^{+},\mathbf{v}^{-})\})$

Preparation Part

The Algorithm 1, 2, 3, 4 are the preparation part, they are the corresponding step 1, 2, 3, 4 for generating $P_E-\overline{D}$ function.

Estimation Part

To estimate the cover $\mathbf{X}$'s secure payload rate, we need to calculate the valid estimation interval $(\hat{P}_{Emin},\hat{P}_{Emax}]$. The sender chooses an expected $\hat{P_E}$ within the interval. Algorithm 5 describes the estimation procedure.

Algorithm 1 Building image library

```
Require: $\{\mathbf{X}_l\}$: cover library
    $M$: number of payload rates
    $\Delta\alpha$: increment of payload rate
Ensure: $\{\mathbf{Y}_l^m\}$: stego library
for $l = 1$ to $L$ do
    for $m = 1$ to $M$ do
        $\mathbf{Y}_k^l = Emb(\mathbf{X}_l, m\Delta\alpha)$;
    end for
end for
return $\{\mathbf{Y}_l^m\}$;
```

Algorithm 2 Sampling sub libraries

```
Require: $\{\mathbf{Y}\}$: stego library
    $R$: number of sampling rounds
Ensure: $R$ sub libraries of $\{\mathbf{Y}\}$
for $r = 1$ to $R$ do
    Randomly sampling $\{\mathbf{Y}\}_{sub}^{r}$ as illustrated in Figure 2;
end for
return $\{\mathbf{Y}\}_{sub}^{1}, \{\mathbf{Y}\}_{sub}^{2}, ..., \{\mathbf{Y}\}_{sub}^{R}$;
```

Algorithm 3 Generating data points

```
Require: $\{\mathbf{Y}\}_{sub}^{1}, \{\mathbf{Y}\}_{sub}^{2}, ..., \{\mathbf{Y}\}_{sub}^{R}$
    $\{\mathbf{X}\}$: cover library
    $Q$: number of points will be generated in one sub library
Ensure: $R \times Q$ data points of $(\overline{D}, P_E)$
for $r = 1$ to $R$ do
    find out $\overline{D}_{min}$ and $\overline{D}_{max}$ of $\{\mathbf{Y}\}_{sub}^{r}$;
    $\Delta\overline{D} = (\overline{D}_{max} - \overline{D}_{min})/2Q$;
    for $q = 1$ to $Q$ do
        $\overline{D}_{mid}^{q} = \overline{D}_{min} + (2q-1)\Delta\overline{D}$;
    end for
    allocate $Q$ independent memory spaces to storage training
    sets $\{(\mathbf{v}^{+}, \mathbf{v}^{-})\}_q, q = 1, 2, ..., Q$;
    for $l = 1$ to $L$ do
        $\mathbf{v}^{+} = Ext(\mathbf{Y}_l)$, $\mathbf{Y}_l$ is in the $\{\mathbf{Y}\}_{sub}^{r}$;
        $\mathbf{v}^{-} = Ext(\mathbf{X}_l)$, $\mathbf{X}_l$ is corresponding cover of $\mathbf{Y}_l$ in $\{\mathbf{X}\}$;
        if $\overline{D}(\mathbf{Y}_l) \in [\overline{D}_{min} + (q-1)\Delta\overline{D}/2, \overline{D}_{min} + q\Delta\overline{D}/2)$ then
            put $(\mathbf{v}^{+}, \mathbf{v}^{-})$ in the $q$-th sub set $\{(\mathbf{v}^{+}, \mathbf{v}^{-})\}_q$;
        end if
    end for
    for $q = 1$ to $Q$ do
        $P_E^q = Trn(\{(\mathbf{v}^{+}, \mathbf{v}^{-})\}_q)$;
        form the data point $(\overline{D}_{mid}^{q}, P_E^q)$;
    end for
end for
return the generated data points $(\overline{D}_{mid}, P_E)$;
```

Algorithm 4 Fitting curve

```
Require: data points $(\overline{D}, P_E)$ generated in 3
Ensure: $P_E - \overline{D}$ function $P_E = f(\overline{D})$
choose a suitable model to fit the curve $f$;
return $P_E = f(\overline{D})$;
```

```
Algorithm 5 Estimate the secure payload rate
Require: cover $\mathbf{X}$
    function $P_E = f(\overline{D})$
    $\alpha_{max}$: the largest $\alpha$ in $\{\mathbf{Y}\}$
Ensure: estimated payload rate $\hat{\alpha}$
  $\overline{D}_{max} = Emb(\mathbf{X}, \alpha_{max})$;
  $\hat{P}_{E\,min} = f(\overline{D}_{max})$;
  $\hat{P}_{E\,max} = f(0)$;
  inform the sender inputs his expected $\hat{P_E}$ which is within the
    interval $(\hat{P}_{E\,min}, \hat{P}_{E\,max}]$;
  $\hat{\alpha} = f^{-1}(\hat{P_E}) \times \alpha_{max}/\overline{D}_{max}$;
  return $\hat{\alpha}$;
```

Experiment

Generating $P_E - \overline{D}$ function

We randomly choose 8000 images from BOSSbase to build the cover library $\{\mathbf{X}\}$, and remaining 2000 images are for testing later. We adopt S-UNIWARD and ternary STC to generate the stego images, SRM and ensemble classifier for steganalysis. We set $\Delta\alpha$ to 0.05, M to 10, R to 10, and Q to 20.

To make the element numbers of training sets uniformly distributed, we use $\sqrt{\overline{D}}$ instead of $\overline{D}$. To ensure every training set has enough samples for training, we set a threshold T, and the set whose number of element is not larger than T will be merged into other set. For the merged new set, its $\sqrt{\overline{D}_{mid}}$ will be re-computed. Here $T = 100$.

The curve of generated $P_E - \overline{D}$ function is showed in Figure 3, its expression is:

$$P_E = \frac{37.82}{76.76 + e^{7.698 \times \sqrt{\overline{D}}}} \tag{15}$$

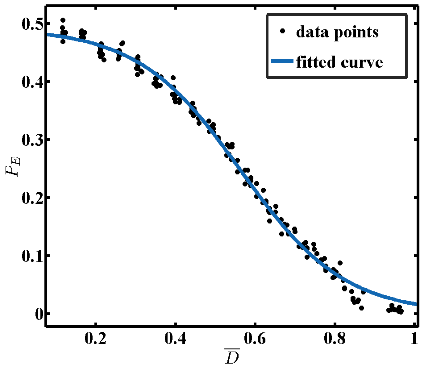

Figure 3. data points and fitted $P_E - \overline{D}$ curve

Fitting $\overline{D} - \alpha$ function

In this section, we will illustrate the $\overline{D} - \alpha$ function. We will show the profile of the function with points and the linear simplified form of the function. We choose 6 images in BOSSbase as the example, whose indexes are: 9000, 9010, 9020, 9030, 9040, 9050. We select 100 payload rates uniformly in the interval $[0.005, 0.5]$. To generate the points, we embed the random information in the covers with ternary STC. The points reflect the practical situation of stego embedding. The Figure 4 shows the profiles of $\overline{D} - \alpha$ function of 6 images. We fit the $\overline{D} - \alpha$ functions of 6 images with power model, the parameters are showed in Table 1.

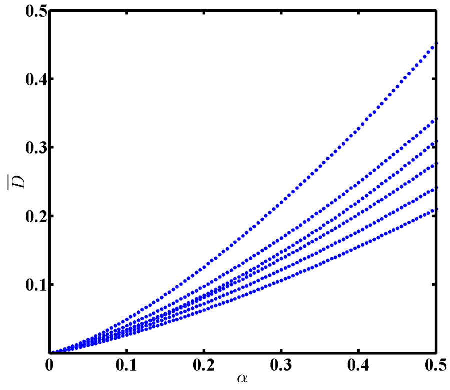

Figure 4. $(\alpha, \overline{D})$ points of 6 images

Parameters of 6 $\overline{D} - \alpha$ functions

image index	k	b
9000	0.5920	1.308
9010	1.180	1.391
9020	0.5167	1.308
9030	0.8106	1.408
9040	0.6906	1.332
9050	0.8698	1.360

As we can see, power indexes are within the interval $[1.3, 1.5]$. To simplify the estimation, we use linear model instead of power model, Figure 5 shows the comparison of the linear form and the power form. Such simplification causes some ”coding loss”, which will be discussed in the following section. Without the linear model, the sender needs to embed with STC two times at least for solving two parameters and run the nonlinear regression for fitting the function.

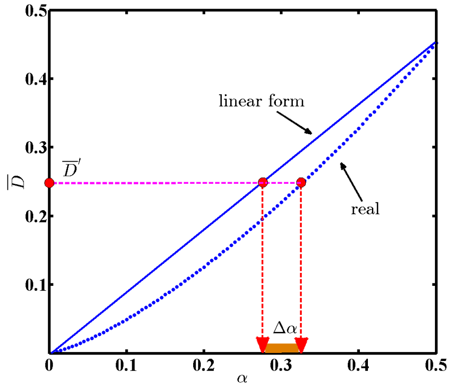

Figure 5. comparison of linear model and real function

Performance Of Proposed Work

As mentioned in Introduction, The SPE problem is within the framework of DLS model. In [4], Filler et al. proposed "coding loss" to evaluate the performance of DLS stego algorithm. The definition of coding loss is as follows:

$$coding\ loss = \frac{|\mathbf{m}|_{max} - |\mathbf{m}|}{|\mathbf{m}|_{max}} \times 100\% \tag{16}$$

In the expression, $|\mathbf{m}|$ is the message length of practical embedding(STC) with a given distortion, while $|\mathbf{m}|_{max}$ is the maximum message length within the given distortion. $|\mathbf{m}|_{max}$ is the performance bound of DLS, which can be calculated by optimal simulator, but can not be reached practically by state-of-the-art STC embedding. The coding loss is the relative difference between practical method and performance bound. To calculate the DLS performance bound, we implement a DLS embedding simulator based on the PLS embedding simulator.

To test the performance proposed SPE method, we calculate the coding loss under 9 P_Es which are within the available estimating interval of generated $P_E - \overline{D}$ function. Covers for testing are reserved 2000 images in BOSSbase. Each point illustrated in Figure 6 is the average coding loss of 2000 covers under a P_E.

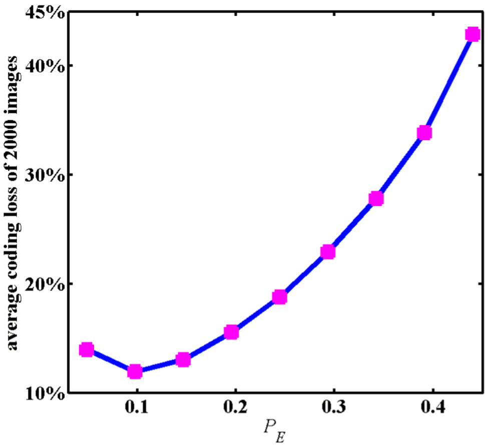

Figure 6. *average coding loss of 2000 images*

The Figure 6 shows that the coding loss goes up with the increasing of P_E. The reason is that we simplify the $\overline{D} - \alpha$ function with linear function. As illustrated in Figure 5, given a distortion $\overline{D}'$, the estimated payload rate α with linear function has a difference $\Delta\alpha$ to the real $\overline{D} - \alpha$ function. At the expense of some stego capacity, the linear $\overline{D} - \alpha$ form shorten the running time of estimation. To balance the efficiency and performance of proposed method, the simplification of $\overline{D} - \alpha$ function is acceptable.

Because $P_E - \overline{D}$ function is calculated via machine learning method, and $\overline{D} - \alpha$ function is simplified, the result is not a precise value. We note that the estimated payload rate is a reference to the sender. Before embedding the message in the cover, sender can adjust his payload rate according to the estimated value. To have a better performance, the sender can choose more powerful steganalyzer beyond the SRM, for example, maxSRM[2].

Conclusion

In this paper, we proposed a practical SPE method based on a priori knowledge functions. The named a priori knowledge contains $P_E - \overline{D}$ function, and the $\overline{D} - \alpha$ function. The $P_E - \overline{D}$ function, which is relation function of steganalysis detection error and stego distortion, is the common knowledge for covers and not related to the particular cover. To estimate the secure payload rate for a cover, the sender needs to calculate the $\overline{D} - \alpha$ function of the cover. The most time-consuming procedure is running STC, its purpose is to acquire an actual data point of $(\alpha, \overline{D})$ for fitting the $\overline{D} - \alpha$ function. The sender first get the stego distortion corresponding to his expecting security from $P_E - \overline{D}$, then calculates the $\overline{D} - \alpha$ function of cover image, and solves out the payload rate via the $\overline{D} - \alpha$ function. If the sender needs to estimate a suitable stego capacity for the cover before communication, the proposed work is a valuable approach.

Acknowledgments

This work was supported by the NSFC under U1636102 and U1536105, National Key Technology R&D Program under 2014BAH41B01 and 2016YFB0801003, and Strategic Priority Research Program of CAS under XDA06030600.

References

[1] Jessica Fridrich, Jan Kodovsky, Rich Models for Steganalysis of Digital Images, IEEE Trans. Inf. Forensics Secur., 7,3 (2012).

[2] Tomáš Denemark, Vahid Sedighi, Vojtěch Holub, Rémi Cogranne, Jessica Fridrich, Selection-Channel-Aware Rich Model for Steganalysis of Digital Images, Proc. WIFS, pg. 48. (2014).

[3] Phil Pallee, Model-based Steganography, Proc. IWDW, pg. 154. (2003).

[4] Tomáš Filler, Jan Judas, Jessica Fridrich, Minimizing Additive Distortion in Steganography Using Syndrome-trellis Codes, IEEE Trans. Inf. Forensics Secur., 6,3 (2011).

[5] Tomáš Filler, Jessica Fridrich, Gibbs Construction in Steganography, IEEE Trans. Inf. Forensics Secur., 5,4 (2010).

[6] Jan Kodovsky, Jessica Fridrich, Vojtěch Holub, Ensemble Classifiers for Steganalysis of Digital media, IEEE Trans. Inf. Forensics Secur., 7,2 (2012).

[7] Lingyu Zhang, Diao Chen, Yun Cao, Xianfeng Zhao, A Practical Method to Determine Achievable Rates for Secure Steganography, Proc. HPCC-CSS-ICESS, pg. 1274. (2015).

[8] Jessica Fridrich, Tomáš Filler, Practical Methods for Minimizing Embedding Impact in Steganography, Proc. SPIE, pg. 650502. (2007).

[9] Tomáš Pevný, Tomáš Filler, Patrick Bas, Using High-dimensional Image Models to Perform Highly Undetectable Steganography, Proc. I-H, pg. 161. (2010).

[10] Tomáš Pevný, Patrick Bas, Jessica Fridrich, Steganalysis by Subtractive Pixel Adjacency Matrix, IEEE Trans. Inf. Forensics Secur., 5,2 (2010).

[11] Vojtěch Holub, Jessica Fridrich, Designing Steganographic Distortion Using Directional Filters, Proc. WIFS, pg. 234. (2012).

[12] Vojtěch Holub, Jessica Fridrich, Tomáš Denemark, Digital Image Steganography Using Universal Distortion, Proc. IH&MMSec, pg. 59. (2013).

[13] Bin Li, Ming Wang, Jiwu Huang, Xiaolong Li, A New Cost Function for Spatial Image Steganography, Proc. ICIP, pg. 4206. (2014).

[14] Patrick Bas, Tomáš Filler, Tomáš Pevný, "Break Our Steganographic System" : The Ins and Outs of Organizing BOSS, Proc. IH, pg. 59. (2011).