Optimized LSB Matching Steganography Based on Fisher Information

Yi-feng Sun
Zhengzhou Information Science and Technology Institute, Zhengzhou , China
Email: yfsun001@163.com

Dan-mei Niu
Electronic Information Engineering College, Henan University of Science and Technology, Luoyang, China
Email: niudanmei@163.com

Guang-ming Tang and Zhan-zhan Gao
Zhengzhou Information Science and Technology Institute, Zhengzhou , China
Email: tgm1983@sina.com gaozhandyx@126.com

***Abstract*—This paper proposes an optimized LSB matching steganography based on Fisher Information. The embedding algorithm is designed to solve the optimization problem, in which Fisher information is the objective function and embedding transferring probabilities are variables to be optimized. Fisher information is the quadratic function of the embedding transferring probabilities, and the coefficients of quadratic term are determined by the joint probability distribution of cover elements. By modeling the groups of elements in a cover image as Gaussian mixture model, the joint probability distribution of cover elements for each cover image is obtained by estimating the parameters of Gaussian mixture distribution. For each sub-Gaussian distribution in Gaussian mixture distribution, the quadratic term coefficients of Fisher information are calculated, and the optimized embedding transferring probabilities are solved by quadratic programming. By maximum posteriori probability principle, cover pixels are classified as the categories corresponding to sub-Gaussian distributions. At last, in order to embed message bits, pixels chose to add or subtract one according to the optimized transferring probabilities of the category. The experiments show that the security performance of this new algorithm is better than the existing LSB matching.**

***Index Terms*—steganography, security, information hiding, quadratic programming, Fisher information**

I. INTRODUCTION

The aim of steganography is to hide secret message imperceptibly into a cover, so that the presence of hidden data cannot be diagnosed. But steganogrpahy faces the threaten of steganalysis. Steganalysis aims to expose the presence of hidden data. How to improve the security of steganography is an important problem.

In Least Significant Bit (LSB) replacement algorithm, the LSBs of cover elements are replaced with message bits. Some structural asymmetry (never decreasing even pixels and increasing odd pixels when hiding message bit) is introduced. It is easy to detect the existence of hidden message. A trivial modification of LSB replacement is LSB matching, which randomly increases or decreases pixel values by one to match the LSBs with the message bits. LSB matching is much harder to detect than LSB replacement algorithm.

The embedding in LSB matching is similar to adding an independent and identically distributed (IID) noise sequence independent of cover. It is well known that values of neighboring pixels in natural images are not independent. Further more, there are complex dependences in the noise component of neighboring pixels [1]. These dependences are violated by LSB matching. Many steganalysis methods utilized this fact [1] [2]. The LSB matching also needs to be improved.

There are two kinds of approaches to improve steganography. The first approach is to preserve a chosen cover model in steganographic methods, such as model-based (MB) steganography[3][4], OutGuess[5], the statistical restoration based steganographic algorithms [6][7], which preserve the first and second order statistics. Another strategy is to minimize a heuristically-defined embedding distortion. The steganography using tri-way pixel-value differencing [8] reduces the quality distortion of stego-image. Matrix embedding methods [9][10][11] minimize the change number of cover elements by linear codes, and the change number is predefined as embedding distortion. Many other syndrome codes steganographic methods, such as steganography using wet paper codes[12], steganographic algorithms using syndrome trellis codes [13][14][15][16], minimize the more complex embedding distortion. According to experiments, the above steganographic methods have the better performance.

In this paper, we propose an optimized LSB matching steganography based on Fisher information. It can

Manuscript received January 1, 2012; revised June 1, 2012; accepted July 1, 2012.
National Nature Science Foundation of China (Grant No.60970141, 60903220) corresponding author: Yi-feng Sun, yfsun001@163.com.

doi:10.4304/jmm.7.4.295-302

demonstrate the security improvement in the sense of Kullback Leibler (KL) divergence. Firstly, we introduce the relation between KL divergence and Fisher information, and explain why Fisher information can be used to improve steganographic security. Then we present an embedding optimization framework based on Fisher information. In the framework, an embedding algorithm is designed to solve the optimization problem whose objective is minimizing the Fisher Information. The optimized LSB matching algorithm is an instance of the framework. We assume that the groups of pixels in the cover image submit Gaussian mixture distributions. After obtaining the Gaussian mixture distribution of pixel group for each cover image, the optimized embedding transferring probabilities are solved by quadratic programming for each sub-Gaussian distribution. Cover pixels are classified as the categories corresponding to sub-Gaussian distributions. The embedding is operated by the optimized embedding transferring probabilities. The experimental results show that the optimized LSB matching steganography is better than the existing LSB matching in the aspect of security performance.

II. STEGANOGRAPHIC SECURITY AND FISHER INFORMATION

KL divergence between cover and stego distributions can be used to measure the stganographic security [17]. Let the distribution of cover objects be P_0 , and the distribution of stego objects be P_λ , where the parameter λ denotes the relative payload, and $0 \le \lambda \le 1$. The KL divergence between P_0 and P_λ is denoted as $D(P_0 \| P_\lambda)$ .The smaller the $D(P_0 \| P_\lambda)$ is, the more secure the steganography will be. If $D(P_0 \| P_\lambda)$ is equal to 0, stego-system is perfectly secure.

The physical interpretation of cover objects is not discussed in [17]. We think that each statistic from a cover can represent the cover object. Here the elements sequence of cover, such as the sequence of the pixel value, or the sequence of Discrete Cosine Transform (DCT) coefficients, is selected as the presentation of the cover object. The elements sequence is denote by $\mathbf{C} = (C_1, C_2, \cdots, C_N)$, where N is the number of the elements that can be used to embed secret information in a multimedia cover. In the following, we use $P_0^{(N)}$ and $P_\lambda^{(N)}$ instead of P_0 and P_λ . It emphasizes that $P_0^{(N)}$ and $P_\lambda^{(N)}$ is N-dimensional joint probability distribution.

Assuming that $P_0^{(N)}$ is known and embedding function is fixed, KL divergence $D(P_0^{(N)} \| P_\lambda^{(N)})$ is the function of λ . KL divergence can also be denoted as $d_N(\lambda)$. The second derivative of $d_N(\lambda)$ to λ (when $\lambda = 0$) is known as steganographic Fisher Information (FI) [18]. It is denoted by $d_N''(0)$. When $\lambda \to 0$,

$$D(P_0^{(N)} \| P_\lambda^{(N)}) \approx \frac{1}{2} d_N''(0) \cdot \lambda^2 . \quad (1)$$

When relative payload λ is fixed, Fisher Information is smaller, and KL divergence will be smaller. Thus Fisher Information can be used for evaluating steganographic security.

Assuming that embedding operations are mutually independent (MI) [19], Fisher Information can be calculated. The probability of cover element being x and stego element being y is denoted by the conditional probability $P(S_k = y \mid C_k = x) = b_{xy}, x \in \boldsymbol{\chi}, y \in \boldsymbol{\chi}$, where $\boldsymbol{\chi}$ is the set of cover and stego elements. The matrix $\mathbf{B} = (b_{xy})_{x \in \boldsymbol{\chi}, y \in \boldsymbol{\chi}}$, also named as embedding matrix, corresponds to an embedding method. The probability b_{xy} is also called as the embedding transferring probability. For embedding matrix $\mathbf{B}$, Fisher information is calculated [20]:

$$d_N''(0) = \sum_{y \in \chi^N} \frac{A(\mathbf{y})^2}{P_0^{(N)}(\mathbf{C} = \mathbf{y})} - N^2 , \quad (2)$$

$$A(\mathbf{y}) = \sum_{i=1}^{n} \sum_{u \in \chi} [P_0^{(N)}(\mathbf{C} = \mathbf{y}[u / y_i]) \cdot b_{uy_i}] . \quad (3)$$

Here $\mathbf{y} = (y_1, y_2, \cdots, y_N) \in \boldsymbol{\chi}^N$ denotes the value of $\mathbf{C} = (C_1, C_2, \cdots, C_N)$, where $\boldsymbol{\chi}^N = \boldsymbol{\chi} \times \boldsymbol{\chi} \times \cdots \times \boldsymbol{\chi}$. Here $\mathbf{y}[u/y_i]$ denotes the sequence $(y_1, \cdots, y_{i-1}, u, y_i, \cdots, y_N)$. It means that u replaces the item y_i in $\mathbf{y}$.

The computational complexity of Fisher information is decided by *N*. We call $d_N''(0)$ *N*-dimensional Fisher Information. We often simplify the model of the covers in order to calculate KL divergence. For example, the elements of cover objects are IID. But it ignores the correlation among the elements of cover objects. Hence, we suppose that the cover is composed of IID element groups. The elements in a group are correlated, and the number of elements in a group is *n*. If a group is composed of two elements, we can take the impact of second-order correlation into account. If it includes more elements, we can analyze the influence of "higher order" correlation. For cover $\mathbf{C} = (C_1, C_2, \cdots, C_N)$, starting from the first element C_1 , every *n* adjacent elements are divided into a group, whose *n*-dimensional joint distribution is denoted by $P_0^{(n)}$. If $P_0^{(n)}$ replaces $P_0^{(N)}$, and *n* replaces *N* in (2) and (3), we can obtain the calculation formula about *n*-dimensional Fisher information corresponding to the element groups. The *n*-dimensional Fisher information is denoted by $d_n''(0)$. In the following, we focus on decreasing the value of $d_n''(0)$ to improve the security of embedding method.

III. THE EMBEDDING OPTIMIZATION FRAMEWORK BASED ON FISHER INFORMATION

In order to improve the steganographic security, we need to find the embedding algorithm whose **B** matrix corresponds to the small Fisher information. In general,

the embedding matrix $\mathbf{B}=(b_{xy})_{x\in\chi,y\in\chi}$ is usually constant for each cover. For example, in LSB matching algorithm, if the message bit does not match the LSB of the cover pixel, one is randomly either added or subtracted from the value of the cover pixel. That is to say $b_{xx+1}=b_{xx-1}=0.25$ and $b_{xx}=0.5$ for each cover. Here we think the probability b_{xy} is variable, and we adjust b_{xy} to minimize $d_n''(0)$ for each cover, assuming that $P_0^{(n)}$ can be obtained according to one cover. So $\mathbf{B}=(b_{xy})_{x\in\chi,y\in\chi}$ can be adapted according to the cover statistical properties, and we can get optimized embedding algorithm.

Here $d_n''(0)$ is determined by the *n*-dimensional joint probability distribution $P_0^{(n)}$ and embedding matrix $\mathbf{B}=(b_{xy})_{x\in\chi,y\in\chi}$. If $b_{xy}, x\in\chi, y\in\chi$ is viewed as variables, Fisher information is the function of b_{xy}. To improve the steganographic security, we need solve the optimization problem in which the Fisher information $d_n''(0)$ is the objective function. Here *n* is constant, and the optimization problem is

$$\mathbf{B}_{\text{optimized}}=\arg\min_{\mathbf{B}=(b_{xy})}\sum_{\mathbf{y}\in\chi^n}\frac{A(\mathbf{y})^2}{P_0^{(n)}(\mathbf{C}=\mathbf{y})}, \tag{4}$$

with the constraints

$$\begin{cases}\sum_{y\in\chi}b_{xy}=1, x\in\chi\\ b_{xy}\ge 0\end{cases}. \tag{5}$$

We expand the formula as follows:

$$\begin{aligned}
&\sum_{\mathbf{y}\in\chi^n}\frac{A(\mathbf{y})^2}{P_0^{(n)}(\mathbf{C}=\mathbf{y})}\\
&=\sum_{u\in\chi}\sum_{v\in\chi}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{\mathbf{y}\in\chi^n}\frac{P_0^{(n)}(\mathbf{C}=\mathbf{y}[u/y_i])P_0^{(n)}(\mathbf{C}=\mathbf{y}[v/y_j])}{P_0^{n}(\mathbf{C}=\mathbf{y})}b_{uy_i}b_{vy_j}\\
&=\sum_{u\in\chi}\sum_{v\in\chi}\sum_{a\in\chi}\sum_{\substack{b\in\chi\\ b\ne a}}\left(\sum_{i=1}^{n}\sum_{\substack{j=1,\\ j\ne i}}^{n}\sum_{\substack{y_h\in\chi\\ (h=1,\cdots,n)\\ (h\ne i,j)}}\frac{P_0^{(n)}(\mathbf{C}=\mathbf{y}[u/y_i,b/y_j])P_0^{(n)}(\mathbf{C}=\mathbf{y}[a/y_i,v/y_j])}{P_0^{(n)}(\mathbf{C}=\mathbf{y}[a/y_i,b/y_j])}\right)b_{ua}b_{vb}\\
&\quad+\sum_{u\in\chi}\sum_{v\in\chi}\sum_{a\in\chi}\left\{\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n}\sum_{\substack{y_h\in\chi\\ (h=1,\cdots,n)\\ (h\ne i,j)}}\frac{P_0^{(n)}(\mathbf{C}=\mathbf{y}[u/y_i,a/y_j])P_0^{(n)}(\mathbf{C}=\mathbf{y}[a/y_i,v/y_j])}{P_0^{(n)}(\mathbf{C}=\mathbf{y}[a/y_i,a/y_j])}+\sum_{i=1}^{n}\sum_{\substack{y_h\in\chi\\ (h=1,\cdots,n)\\ (h\ne i)}}\frac{P_0^{(n)}(\mathbf{C}=\mathbf{y}[u/y_i])P_0^{(n)}(\mathbf{C}=\mathbf{y}[v/y_i])}{P_0^{(n)}(\mathbf{C}=\mathbf{y}[a/y_i])}\right\}b_{ua}b_{va}
\end{aligned} \tag{6}$$

Here $\mathbf{y}[a/y_i,b/y_j]$ denotes the sequence

$$(y_1,\cdots,y_{i-1},a,y_{i+1},\cdots y_{j-1},b,y_{j+1},\cdots,y_n).$$

From (6), we can find that Fisher information $d_n''(0)$ is a quadratic function about the probability b_{xy}. Searching b_{xy} which minimizes Fisher information is a quadratic programming problem.

IV. Optimized LSB matchIng embedding Ealgorithm In Spatial Domain

In this paper, we optimize the LSB matching algorithm in image spatial domain. To get the optimized b_{xy}, we must know the *n*-dimensional joint distribution $P_0^{(n)}$ of the pixel group made of adjacent pixels. The distribution of pixel group in spatial domain is complex. But Gaussian mixture model (GMM) can describe many complex distributions approximately [21]. We take GMM as the distribution model of the *n*-dimensional pixel group in spatial domain.

GMM is the sum of weighted sub-Gaussian distributions. It can fit very complex distribution by adjusting the parameters of GMM. The small local areas of an image often correspond to a particular object or belong to the same background. Therefore, we can think that the pixel groups in a local area have the same distribution characteristics. Sub-Gaussian distributions in GMM can reflect these local characteristics. The pixel groups in an image are instances of these sub-Gaussian distributions in GMM.

For GMM composed of K sub-Gaussian distributions, pixel groups can be divided into K categories, and each sub-Gaussian distribution associates to a separate category. Different pixel groups belonging to different categories have different distribution characteristics. The embedding matrix $\mathbf{B}$ should be optimized according to the local distribution properties of the cover image. Denote the optimized embedding matrix corresponding to k sub-Gaussian distribution as $\mathbf{B}_{\text{optimized}}^{k}$, where $k=1,2,\cdots,K$. For a pixel which is selected to embedding message bit, if it belongs to the category of k sub-Gaussian distribution, we should modify it according to $\mathbf{B}_{\text{optimized}}^{k}$.

The whole framework of the optimized embedding algorithm is shown in Figure 1. Firstly, we estimate the GMM parameters for the pixel groups in a cover image. Secondly, we obtain the optimized embedding matrix for each sub-Gaussian distribution in GMM. Then we should judge the sub-Gaussian distribution category of the pixels which is selected to embedding message bit. At last, if the LSBs of those pixels don't equal to message bits, the

pixels should add or subtract 1 according to the transferring probabilities indicated by the optimized embedding matrix corresponding to each category.

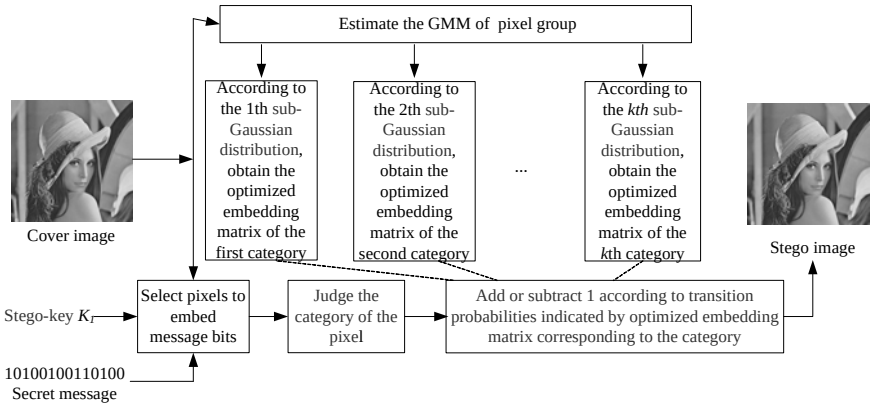

Figure 1. The framework of the optimized LSB matching algorithm

A. Estimating GMM for each cover image

For pixel groups $\mathbf{y}=(y_1,y_2,\cdots,y_n)$ in a cover image, we assume that they submit to the following Gaussian mixture distribution

$$P_0(\mathbf{y})=\sum_{k=1}^{K}\pi_k P_{0,k}(\mathbf{y}), \tag{7}$$

$$P_{0,k}(\mathbf{y})=\frac{1}{(2\pi)^{n/2}}\left|R_k\right|^{-1/2}\exp\left\{-\frac{1}{2}(\mathbf{y}-m_k)R_k^{-1}(\mathbf{y}-m_k)^{\mathrm{T}}\right\} \tag{8}$$

where K is the amount of sub-Gaussian distributions in GMM. Here m_k and R_k represent the mean vector and covariance matrix of sub-Gaussian distribution respectively, and π_k denote the weights of the sub-Gaussian distributions, where $k=1,2,\cdots,K$. In the following, we discuss how to estimate the parameters $\{m_k,R_k,\pi_k\}_{k=1,2,\cdots,K}$ for each cover image.

The pixel group is made up of two adjacent pixels in order to reduce computational complexity in our scheme. But in a cover image, the horizontal adjacent pixels, the vertically adjacent pixels, and the adjacent pixels in main diagonal direction or in vice diagonal direction reflect different second-order statistical properties. In order to embody these second-order properties in GMM, we get pixel group samples as follows: Assume that an image $\mathbf{C}$ has $N_1\times N_2$ pixels. Firstly, from the pixel in the first line and the first column of the image, every two horizontal adjacent pixels $(C_{i,j},C_{i,j+1})$ constitute a pixel group sample, where $i=1,\cdots,N_1$ and $j=1,\cdots,N_2-1$, and there are a total of $N_1\times(N_2-1)$ samples; secondly, every two vertical adjacent pixels $(C_{i,j},C_{i+1,j})$ constitute a sample, and the number of these samples is $(N_1-1)\times N_2$; Then, every two adjacent pixels in the main diagonal direction $(C_{i,j},C_{i+1,j+1})$ constitute a sample, where $i=1,\cdots,N_1-1$ and $j=1,\cdots,N_2-1$, there are $(N_1-1)(N_2-1)$ samples in total; at last, every two adjacent pixels in the vice diagonal direction $(C_{i+1,j},C_{i,j+1})$ also constitute $(N_1-1)(N_2-1)$ samples. With the above $4(N_1-1)(N_2-1)+(N_1-1)+(N_2-1)$ samples, we use the MDL algorithm of Purdue University [21] to estimate the parameters of GMM. The MDL algorithm is an extension of Expectation Maximum (EM) algorithm. Please refer to the literature [22].

Note that GMM is the continuous distribution. We need the discrete distribution in optimization problem (4). In order to reduce computation complexity, the values of the sub-Gaussian probability density $p_{0,k}(\mathbf{y})$ replace the discrete values directly.

B. Obtaining the optimized LSB matching embedding matrix for each sub-Gaussian distribution

Here we embed information into spatial pixel groups in grayscale image, then $\chi=\{0,1,\cdots,255\}$. The embedding matrix of LSB matching is adjusted as follows: Use x, $x=1,2,\cdots,254$, to represent the value of the selected cover pixel. If the secret message bit is different with the LSB of x, and x is not equal to 0 or 255, we add 1 with the probability $b_{x,x+1}$, or subtract 1 with the probability $b_{x,x-1}$. The probabilities $b_{x,x+1}$ and $b_{x,x-1}$ are the variables in optimization problem. There are 254×2 unknown variables in all. We suppose that the secret message bits have been encrypted. They are pseudo-random bit streams. We think that half of pixels' LSBs are equal to the message bits. So $b_{x,x}=0.5$, and the new LSB matching embedding matrix becomes

$$\mathbf{B}=\begin{pmatrix} 0.5 & 0.5 & 0 & \cdots & 0 & 0 & 0 \\ b_{1,0} & 0.5 & b_{1,2} & \cdots & 0 & 0 & 0 \\ 0 & b_{2,1} & 0.5 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0.5 & b_{253,254} & 0 \\ 0 & 0 & 0 & \cdots & b_{254,253} & 0.5 & b_{254,255} \\ 0 & 0 & 0 & \cdots & 0 & 0.5 & 0.5 \end{pmatrix}. \tag{9}$$

For each sub-Gaussian distribution of GMM, we need to obtain the corresponding optimized LSB matching embedding matrix, denoted as $\mathbf{B}^k_{\text{optimized}}$, $k=1,\cdots,K$. Here the pixel group is made up of two adjacent pixels, and the optimization problem is simplified as

$$\mathbf{B}^k_{\text{optimized}} = \arg\min_{\mathbf{B}=(b_{xy})} f_k(\mathbf{B}), \tag{10}$$

$$\begin{aligned} f_k(\mathbf{B}) &= \sum_{u\in\chi}\sum_{v\in\chi}\sum_{a\in\chi}\sum_{\substack{b\in\chi\\ b\neq a}}\left(\sum_{i=1}^{2}\sum_{\substack{j=1,\\ j\neq i}}^{2}\frac{P_{0,k}(y_i=u,y_j=b])P_{0,k}(y_i=a,y_j=v])}{P_{0,k}(y_i=a,y_j=b)}\right)b_{ua}b_{vb} \\ &+\sum_{u\in\chi}\sum_{v\in\chi}\sum_{a\in\chi}\left\{\sum_{i=1}^{2}\sum_{\substack{j=1\\ j\neq i}}^{2}\frac{P_{0,k}(y_i=u,y_j=a])P_{0,k}(y_i=a,y_j=v])}{P_{0,k}(y_i=a,y_j=a)}\right. \\ &\left.+\sum_{i=1}^{2}\sum_{y_j\in\chi}\frac{P_{0,k}(y_i=u,y_j])P_{0,k}(y_i=v,y_j])}{P_{0,k}(y_i=a,y_j)}\right\}b_{ua}b_{va}. \end{aligned} \tag{11}$$

The constraints are

$$\begin{cases} b_{x,x+1}+b_{x,x-1}=0.5,\ b_{x,x+1}\geq 0,\ b_{x,x-1}\geq 0, x\in\boldsymbol{\chi}, x\neq 0, x\neq 255 \\ b_{x,x}=0.5,\ x\in\boldsymbol{\chi} \\ b_{0,1}=0.5, b_{255,254}=0.5 \\ \text{others } b_{x,y}=0 \end{cases}. \tag{12}$$

This optimization problem is a quadratic programming. The standard form of quadratic programming is $\min_{\boldsymbol{x}}\frac{1}{2}\boldsymbol{x}\mathbf{H}\boldsymbol{x}^{\mathrm{T}}$, where $\boldsymbol{x}=\{x_1,x_2,\cdots,x_L\}$ is a row vector, and $\mathbf{H}$ represents the quadratic coefficient matrix. By arranging matrix $\mathbf{B}$ into a row vector, we can get the standard form. The elements of $\mathbf{H}$ are the corresponding coefficients in (6).

The complexity degree of quadratic programming depends on the element number of $\boldsymbol{x}=\{x_1,x_2,\cdots,x_L\}$. The element number of $\mathbf{H}$ is the square of the element number of $\boldsymbol{x}$. If computer memory can not store matrix $\mathbf{H}$, we need to iteratively calculate. It is slow.

Here, though the amount of elements in matrix $\mathbf{B}$ is 256×256, many matrix elements are 0 in our algorithm because $b_{xy}=0$ when $|x-y|\geq 2$. If b_{xy} is 0, its corresponding quadratic coefficient has no effect on the quadratic function. There are 766×766 nonzero elements in $\mathbf{H}$, which need to be stored in the computer memory. They are $b_{x,x-1}$, $b_{x,x+1}$, where $x=1,2,\cdots,254$, $b_{x,x}=0.5$, where $x=0,1,\cdots 255$, $b_{0,1}=0.5$ and $b_{255,254}=0.5$. PC can store them. So we use quadratic programming function in Matlab software to solve the optimized transferring probabilities $b_{x,x-1}$ and $b_{x,x+1}$.

C. Judging the category of Pixels

In the existing LSB matching algorithm, if the message bit does not match the LSB of the cover pixel, one is randomly either added or subtracted from the value of the cover pixel. In our optimized LSB matching algorithm, if the message bit does not match the LSB of the cover pixel, we need judging the category to which the pixel belongs at first. The embedding matrix of the kth category is $\mathbf{B}^k_{\text{optimized}}$. We add or subtract 1 by the probability $b_{x,x-1}$ or $b_{x,x+1}$ in $\mathbf{B}^k_{\text{optimized}}$.

Now we explain the method of judging the category the pixels $C_{i,j}$ belonging to, where i, j are the row and column induce of the pixel. Firstly, we constitute pixel group by $C_{i,j}$ and its adjacent pixels. For example, constructing the pixel group $(C_{i,j},C_{i,j+1})$ with the right horizontal adjacent pixel $C_{i,j+1}$. Note that if $C_{i,j}$ is at the right edge of the image, then we can use the left horizontal adjacent pixel $C_{i,j-1}$ to constitute the pixel group. Then based on the value of the pixel group $\mathbf{y}=(C_{i,j},C_{i,j+1})$ or $\mathbf{y}=(C_{i,j-1},C_{i,j})$, the posterior probabilities of each sub-Gaussian distribution is calculated as follows:

$$P(k\mid\mathbf{y})=\frac{P_{0,k}(\mathbf{C}=\mathbf{y})\pi_k}{\sum_{k=1}^{K}P_{0,k}(\mathbf{C}=\mathbf{y})\pi_k},\ k=1,2,\cdots,K. \tag{13}$$

The pixel $C_{i,j}$ is categorized according to maximum posteriori probability principle

$$k_{\text{optimal}}=\arg\max_{k=1,2,\cdots,K}P_{0,k}(\mathbf{C}=\mathbf{y})\pi_k, \tag{14}$$

where k_{optimal} denotes the category of pixels $C_{i,j}$ to be determined.

D. Embedding and Extraction Algorithm

The embedding process of the optimized LSB matching steganography is summarized as follows:

Input: cover image $\{C_{i,j}\}$ which has $N=N_1\times N_2$ pixels, M bits encrypted secret message, stego-key k_1.

Output: stego image $\{S_{i,j}\}$.

Step(i): Model the two-dimensional probability distribution of two spatial adjacent pixels as GMM, and estimate the parameters of GMM according to the pixel group samples from cover image $\{C_{i,j}\}$.

Step(ii): For each sub-Gaussian distribution $P_{0,k}(\mathbf{y})$ of GMM, work out the matrix $\mathbf{H}$ by calculating the coefficients of nonzero $b_{x,x-1}$ and $b_{x,x+1}$ in objective function $f_k(\mathbf{B})$ according to (11), then solve the quadratic programming problem (10), and get the optimized embedding matrix $\mathbf{B}^k_{\text{optimized}}$ corresponding to each sub-Gaussian distribution.

Step(iii): Calculate the rate of pixels used for embedding, which is denoted as $\lambda=M/N$. The rate is equal to the relative payload in general sense. According to λ and the stego-key k_1, select the set of image pixels $\{C_{i,j}\}$ to embed message bits;

Step(iv): For each selected pixel $C_{i,j}$, if its LSB is equal to the message bit, do not change it, otherwise, turn to the next step;

Step(v): Judge whether the value of $C_{i,j}$ is 0 or 255, if so, replace the LSB of $C_{i,j}$ with the message bit, otherwise, turn to the next step;

Step(vi): Constitute the pixel group **y** with the right horizontal adjacent $C_{i,j+1}$ or the left horizontal adjacent pixel $C_{i,j-1}$, and then determine the category k_{optimal} of the pixel group according to (14);

Step(vii): Obtain $b^{k_{\text{optimal}}}_{C_{i,j},C_{i,j}+1}$, then produce a pseudo-random number $n_{i,j}$ within the range (0,1), if $n_{i,j} \in (0, 2b^{k_{\text{optimal}}}_{C_{i,j},C_{i,j}+1})$, then $S_{i,j} = C_{i,j} + 1$; otherwise, $S_{i,j} = C_{i,j} - 1$. The reason why selecting the threshold $2b^{k_{\text{optimal}}}_{C_{i,j},C_{i,j}+1}$ is that $b^{k_{\text{optimal}}}_{C_{i,j},C_{i,j}+1} + b^{k_{\text{optimal}}}_{C_{i,j},C_{i,j}-1} = 0.5$.

The extraction embedding process of the optimized LSB matching steganography is summarized as follows:

Input: stego-image $\{S_{i,j}\}$ which has $N = N_1 \times N_2$ pixels, the rate λ of pixels used for embedding, and stego-key k_1.

Output: secret message.

Step(i): According to λ and the stego-key k_1, select the set of image pixels $\{C_{i,j}\}$ to embed message;

Step(ii): Select the LSB in each selected pixel $C_{i,j}$, and combine them into the secret message.

In summary, for a cover image, we should calculate K optimized embedding matrices corresponding to each sub-Gaussian distribution. Then the message bits are embedded by adding or subtracting 1 according to the probability indicated by the embedding matrix corresponding to the category the pixel belonging to.

V. EXPERIMENT AND ANALYSIS

In this section, we present some experimental results to demonstrate the effectiveness of our proposed optimized LSB matching steganography compared with the existing LSB matching methods. A common way of testing steganographic schemes is to report the detection metric of some steganalysis methods empirically estimated from a database of cover and stego images where each stego image carriers a fixed relative payload[16].

Here the cover image database consists of 1000 images which were downloaded from USDA NRCS Photo Gallery [23]. The images are of very high resolution TIF files (mostly 2100×1500) and appear to be scanned from a variety of film sources. For testing, the images were resampled to 614×418 and converted to grayscale (The tool used is Advanced Batch Converter 3.8.20, and the selected interpolation filter is bilinear).

Firstly, the cover images were used to generate 3 groups of 1000 stego images of the existing LSB matching with the relative payload $\lambda = 1$, $\lambda = 0.75$ and $\lambda = 1$, respectively. Then, another 3 groups of 1000 stego images of the optimized LSB matching were generated with the relative payload $\lambda = 1$, $\lambda = 0.75$ and $\lambda = 1$, respectively. The 2000 stego images of two kind of steganography with the same relative payload and the corresponding cover images were used to build a training set and a test set. The training and the test sets were built randomly, both containing 50% cover and 50% stego images. The training sets are used to train some steganalysis detector. The security of two kind steganography with the same relative payload is compared by the detection performance on test sets.

The steganalysis detector makes two types of errors - either detect the cover image as stego (false alarm, or false positive) or recognize the stego image as cover (missed detection, or false negative). The corresponding probabilities are denoted P_{FA} and P_{MD}. The receiver operating characteristic (ROC) curve is obtained by plotting 1-P_{MD}(P_{FA}) as a function of P_{FA}. The ROC curve can be reduced into a scalar detection measure called the mimimum error probability:

$$P_E = \min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD}(P_{FA})) \tag{15}$$

Both the ROC curve and the mimimum error probability P_E can be used as the detection performance metrics of the steganalysis method for some steganography with the fixed relative payload.

A. ROC curves metric

In the experiments, the first steganalytic detector [2] adopts the Difference Characteristic Function Moments as the features, and Fisher linear discriminator (FLD) as the classifier. The detection performance for the existing LSB matching is excellent [2].

We adopt ROC curves to evaluate the security performances of two steganography methods. The ROC curves of detector show how the detection probability (true positive rate, the fraction of the stego images that are correctly classified) and the false alarm probability (false positive rate, the fraction of the cover images that are misclassified as stego images) vary as detection threshold is varied. The lower the curve is, the more difficult the detector is. It means that the corresponding steganography is more secure.

Figure 2 shows the ROC curves of the existing LSB matching and the optimized LSB matching with the relative payload $\lambda = 1$. Figure 3 shows the ROC curves with $\lambda = 0.75$. Figure 4 shows the ROC curves with $\lambda = 0.5$. From the figures, the ROC curves of the optimized LSB matching are higher than those of the existing LSB matching. Thus the optimized LSB matching is more secure than the existing LSB matching. Further more, the distinctions between the ROC curves of two steganography become small with the decrease of the relative payload. It means that the improvement of security performance is more obvious when λ is larger.

B. Minimum error probability metric

We also use the steganalysis detector in the literature[1] to compare the security of two steganography methods.

The steganalysis detector is famous in the realm of steganalysis and steganography. We use the second-order SPAM features with T=3 and first-order SPAM features with T=4 respectively. There are 686 dimensions in the second-order features, and 162 dimensions in the first-order features. With regards to machine learning, we use soft-margin SVMs with a Gaussian kernel of width γ. Soft-margin SVMs penalize the error on the training set through a hyper-parameter C. In this section, the minimum error probability in (15) is used to evaluate the security. The steganography, with larger P_E, is more secure than that with smaller P_E.

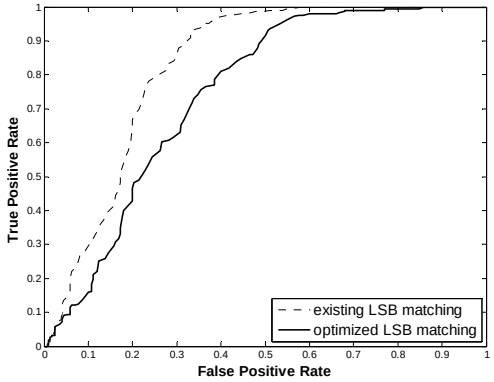

Figure 2. The ROC curves for two kind LSB matching steganography when the relative payload is 1

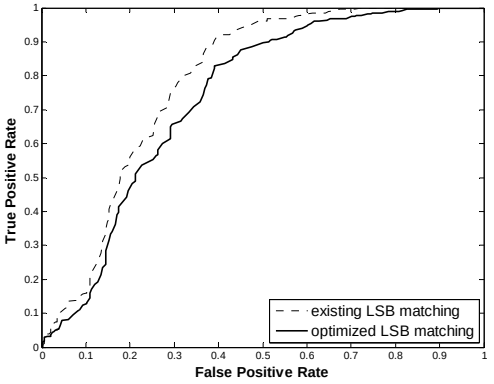

Figure 3. The ROC curves for two kind LSB matching steganography when the relative payload is 0.75

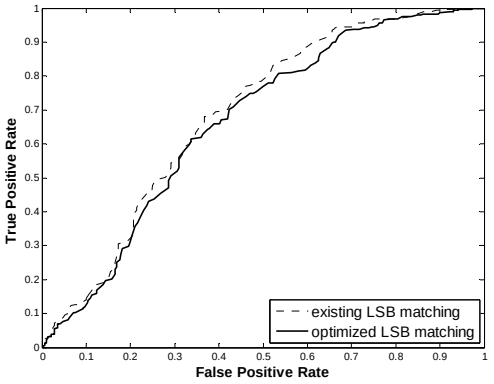

Figure 4. The ROC curves for two kind LSB matching steganography when the relative payload is 0.5

In the experiment, the parameters $C = 300$ and $\gamma = 5$. Note that we didn't using a grid-search with five-fold cross-validation on the training set to obtain the best C and γ as in [1]. The reason is that we only need to find out which steganography has larger P_E. We needn't the best detector with the best C and γ. The grid-search is time consuming. As a result, the values of P_E in our experiments is larger than that in the literature [1]. But it is trivial to compare the security of two steganographic methods.

Table I demonstrates the security performances of the existing LSB matching steganography and the optimized LSB matching steganography. From table I, with each relative payload, the minimum error probability P_E of the optimized LSB matching is larger than that of the existing LSB matching. So the optimized LSB matching is more secure than the existing one. On the other hand, the distinction of the minimum errors of two steganography decreases when the relative payload become small. Thus the larger the λ is, the more the security improvement of the optimized LSB matching is. The results are consistent with the results in V-A section.

TABLE I. MINIMUM ERRORS OF SPAM STEAGANALYZERS

Relative payload λ	SPAM Features	existing LSB matching	Our optimized LSB matching
0.5	Second-order with T=3	0.3955	**0.4098**
	First-order with T=4	0.2962	**0.3105**
0.75	Second-order with T=3	0.3636	**0.3972**
	First-order with T=4	0.2556	**0.2893**
1	Second-order with T=3	0.3359	**0.3808**
	First-order with T=4	0.2263	**0.2643**

VI. SUMMARY

We present an optimized LSB matching algorithm. The probabilities of adding or subtracting one for embedding, which is also named as the embedding transferring probabilities, are determined by solving an optimization problem. We demonstrate that Fisher information is the quadratic function of the embedding transferring probabilities. Assuming that the groups of pixels in the cover image can be modeled as GMM, we obtain Gaussian mixture distribution of pixel group for each cover image. Based on it, the optimized embedding transferring probabilities are solved by quadratic programming for each sub-Gaussian distribution. The experiments show that the security performance of this new algorithm is better than the existing LSB matching.

Furthermore, the principle of improving the security in our optimized LSB matching algorithm is different with that of stegnography using syndrome coding. The

optimized embedding method here can be combined with them. The reason is that many steganographic algorithms based on syndrome coding [12][13] don't define the specific modification operation (adding or subtracting 1) on cover elements, and our optimized LSB matching algorithm can be utilized to decide which operation to be selected. The combination of two kinds of algorithms may further improve the security.

REFERENCES

[1] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," IEEE Transactions on Information Forensics and Security, vol. 5, no. 2, 2010, pp. 215–224.

[2] Y. Sun, F. Liu, and B. Liu, "Steganalysis based on difference image," in IWDW 08, LNCS 5450. Heidelberg: Springer-Verlag, 2009, pp. 184–198.

[3] P. Sallee, "Model-based steganography," in Proceedings of IWDW 2003, LNCS 2939, 2004, pp.154–167.

[4] P. Sallee, "Model-based methods for steganography and steganalysis," International Journal of Image and Graphics, vol. 5, no. 1, 2005, pp. 167–189.

[5] N. Provos, "Defending against statistical steganalysis," in Proceedings of 10th USENIX Security Symposium. Washington. DC, 2001, pp. 323–335.

[6] K. Solanki, K. Sullivan, and U. Madhow, etc, "Provably secure steganography: Achieving zero K-L divergence using statistical restoration," in Proceedings of ICIP'06. Piscataway: IEEE Signal Processing Society, 2007, pp. 125–128.

[7] A. Sarkar, K. Solanki, and U. Madhow, etc, "Secure steganography: statistical restoration of the second order dependencies for improved security," in Proceedings of ICASSP'07. Piscataway: IEEE Signal Processing Society, 2007, pp. II277–II280.

[8] K. Changa, C. Changa, and P. S. Huangb, etc, "A novel image steganographic method using tri-way pixel-value differencing," Journal of Multimedia, vol.3, no.2, 2008, pp.37-44.

[9] A. Westfeld, "F5--A steganographic algorithm high capacity despite better steganalysis," in Proceedings of IH 2001, LNCS 2137, 2001, pp. 289–302.

[10] Y. Kim, Z. Duric, and D. Richards, "Modified matrix encoding technique for minimal distortion steganography," in Proceedings of IH 06, LNCS 4437, 2006, pp. 314–327.

[11] J. Fridrich, and D. Soukal, "Matrix embedding for large payloads]," IEEE Transactions on Information Forensics and Security, vol. 1, no. 3, 2006, pp. 390–394.

[12] J. Fridrich, M. Golijan, and D. Soukal, etc, "Wet paper codes with improved imbedding efficiency," IEEE Transactions on Information Forensics and Security, vol. 1, no. 1, 2006, pp. 102–110.

[13] T. Filler, J. Judas, and J. Fridrich, "Minimizing embedding impact in steganography using Trellis-Coded quantization," Proceedings of Media Forensics and Security III, SPIE 7451, 2010. pp. 715405-1~715405-14.

[14] T. Filler, J. Judas, and J. Fridrich, "Gibbs construction in steganography," IEEE Transactions on Information Forensics and Security, vol. 5, no.4, 2010, pp.705-720.

[15] T. Pevny, T. Filler, and B. Patrick, "Using high-dimensional image models to perform highly undetectable steganography," in Proceedings of IH'10, LNCS 6387, 2010, pp.161-177.

[16] T. Filler, and J. Fridrich, "Design of adaptive steganographic schemes for digital images," in Proceedings of Media Watermarking, Security and Forensics III, SPIE 7880, 2011, pp. 78800F-1~78800F-13.

[17] C. Cachin, "An information-theoretic model for steganography," in proceedings of IH'98, LNCS 1525. Heidlberg: Springer-Verlag, 1998, pp. 306–318.

[18] T. Filler, and J. Fridrich, "Fisher Information determines capacity of secure steganography," in proceedings of IH'09, LNCS 5806. Heidlberg: Springer-Verlag, 2009, pp. 31–47.

[19] T. Filler, A. D. Ker, and J. Fridrich, "The square root law of steganographic capacity for Markov covers," in Proceedings of Media Forensics and Security 09, SPIE 7254, Bllingham: SPIE press, 2009, pp. 08-1–08-11.

[20] A. D. Ker, "Estimating the information theoretic optimal stego noise," in Proceedings of IWDW'09, LNCS 5703. Heidlberg: Springer-Verlag, 2009, pp. 184–198.

[21] G. McLachlan, and P. David, "Finite mixture models," New York: Wiley, 2001.

[22] C. A. Bouman, "CLUSTER: An unsupervised algorithm for modeling Gaussian mixtures," http://www.ece.purdue.edu/~bouman.

[23] http://photogallery.nrcs.usda.gov.

Yi-feng Sun was born in Xinxiang, China in 1976. He received the B.S., M.S. and PH.D. degrees in computer science from Zhengzhou information science and technology institute, Zhengzhou, China, in 1999, 2002, 2010.

He is presently an instructor with the department of Information Security, Zhengzhou information science and technology institute, Zhengzhou, China. His research interests are in image steganography, image steganalysis, computer vision.

Dan-mei Niu was born in Luoyang, China in 1979. She received the B.S. degree in 2001, and received the M.S. degree from Henan university of science and technology, Luoyang, China in 2006.

She is presently an instructor in electronic information engineering college, Henan university of science and technology, Luoyang, China. Her research interests are in information security, digital rights management.

Guang-ming Tang was born in Wuhan, China in 1963. She received the B.S., M.S. and PH.D. degrees in information security from Zhengzhou information science and technology institute, Zhengzhou, China, in 1983, 1990, and 2008, respectively.

She is presently a professor with the department of Information Security, Zhengzhou information science and technology institute, Zhengzhou, China. Her fields of professional interest are information hiding, watermarking and software reliability. She has published 51 research articles and 3 books in these areas.

Ms. Tang is a recipient of Provincial Prize for Progress in Science and Technology.

Zhan-zhan Gao was born in Shijiazhuang, China in 1988. He received the B.S. degree in electronic science and technology from Zhengzhou information science and technology institute, Zhengzhou, China, in 2011. He is pursuing the M.S degree in information security at Zhengzhou information science and technology institute. His research interests include information hiding and information security.