Hindawi Publishing Corporation
EURASIP Journal on Information Security
Volume 2009, Article ID 382310, 8 pages
doi:10.1155/2009/382310

Research Article

Peak-Shaped-Based Steganographic Technique for JPEG Images

Lorenzo Rossi, Fabio Garzia, and Roberto Cusani

INFOCOM Department, "Sapienza" Università di Roma, Via Eudossiana 18, 00184 Rome, Italy

Correspondence should be addressed to Lorenzo Rossi, lorenzo.rossi@uniroma1.it

Received 1 August 2008; Revised 16 October 2008; Accepted 29 January 2009

Recommended by Andreas Westfeld

A novel model-based steganographic technique for JPEG images is proposed where the model, derived from heuristic assumptions about the shape of the DCT frequency histograms, is dependent on a stegokey. The secret message is embedded in DCT domain through an accurate selection of the potentially modifiable coefficients, taking into account their visual and statistical relevancy. A novel block measure, named discrepancy, is introduced in order to select suitable areas for embedding. The visual impact of the steganographic technique is evaluated through PSNR measures. State-of-the-art steganalytical test is also performed to offer a comparison with the original model-based techniques.

1. Introduction

Steganography is the art of hidden communication. Its aim is in fact to hide the presence of communication between two parties. Current steganographic techniques conceal secret messages in innocuous-looking data as images, audio files, and video files.

Following the approach in [1], the actual message to be transmitted is called embedded message, while the innocuous-looking message, in which the other will be enclosed, is the cover message (cover image in case of images). This embedding process creates a new message, called stego message (stego image in case of images), with the same visual and statistical appearance of the cover message but containing the embedded message.

Modern steganographic techniques follow Kerckhoffs' principle: the technique used to hide the embedded message is known to the opponent, and the security of the stegosystem lies only in the choice of a hidden information shared between the sender and the receiver, called stegokey [1].

Because of their large diffusion among the Internet, JPEG images are very attractive as cover messages. As a consequence, many steganographic techniques have been designed for JPEG, most of them embedding the message in DCT domain by modifying the least significant bits (LSBs) of the quantized DCT coefficients. One of the first JPEG steganographic techniques following this approach is Jsteg [2]. Outguess [3] is not only similar to Jsteg, but also preserves DCT global histogram by additional bit-flipping. F5 [2] performs the embedding by decreasing the absolute value of DCT coefficients, thus preserving the DCT histograms peak-shape. Unfortunately, all the techniques are detected with known statistical methods [4].

Model-based steganography (MB) [5] introduces a different methodology, where the message is embedded in the cover according to a model representing cover message statistics. In [5], two image steganographic techniques (MB1 and MB2) are illustrated: MB1 models DCT AC histograms by the generalized Cauchy distribution and embeds the message in the cover image through an entropy decoder driven by the model. MB2 also preserves blockiness [6]. In [7], an ad hoc steganalytical test is developed to detect MB1.

The aim of this work is to improve the performance of the mentioned Model-based techniques by considering a better model and a more accurate selection of the modifiable coefficients. The peak-shaped-based (PSB) technique, here illustrated, applies F5 heuristic principles in a Model-based methodology. It is known that both MB1 and MB2 modeling of every DCT AC frequency leaves a fingerprint which allows to detect the presence of the embedded message. In fact, MB1 is detected via a model calculation followed by a goodness-of-fit test [7]. On the other hand, PSB modeling does not characterize strictly DCT AC histograms, but only models in a broad sense the histograms shape. Many cover images

already present similar properties, thus making much more difficult the fingerprint discovering. Moreover, PSB model depends on the stegokey: a simple analysis of the stego image is not sufficient to perform an exact model calculation, regardless of the possible attacker. Futhermore, PSB accurately selects the modifiable coefficients by exploiting the quantization matrix and introducing a novel parameter, named discrepancy, measuring how much a given image portion is suitable for embedding the hidden message.

This paper is organized as follows. Model-based methodology is introduced in Section 2, together with embedding and extraction algorithms. In Section 3 the PSB technique is described and its superior performance over original Model-based techniques is demonstrated. Conclusions are drawn in Section 4.

2. PSB Steganography

The steganographic technique introduced in this work, named peak-shaped-based steganography (PSB), is developed following the Model-based steganography principles exposed in Sallee's work [5] of which for sake of completeness, a brief outline is given in Section 2.1. Next, PSB is illustrated and the embedding and the extraction algorithm are described.

2.1. Principles of Model-based Steganography. Model-based steganography was first introduced in 2003 [5]. The aim of Model-based steganography is in characterizing some statistical properties of the cover message in order to embed the secret message without altering these properties. The outline of Model-based steganography is described in the following.

A cover message, represented as a random variable X, is split into two parts, X_a, that remain unaltered during the embedding, and X_b, that is modified to carry the embedded message. X_a is selected so as to preserve the relevant characteristics of the cover, whereas X_b can be modified without altering the perceptual and statistical characteristics of the cover message. By modeling the cover message class X according to a probability distribution $\widehat{P}_X(x)$ it is possible to calculate the conditioned probability distribution $\widehat{P}_{X_b|X_a}(x_b|x_a)$.

The embedded message is assumed to be a uniform random stream of bits, which is in fact the same distribution shown by encrypted messages. The embedding outline is shown in Figure 1. The cover message x is split into x_a and x_b, then the embedded message is processed by an entropy decoder according to the conditioned probability distribution $\widehat{P}_{X_b|Xa}(x_b|x_a)$. The output of the decoder is denoted by x'_b and replaces x_b to form together with x_a the stego message x'.

The extraction outline is shown in Figure 2. Its structure is very similar to the embedding scheme: the main difference consists in the replacement of the entropy decoder by an entropy encoder. The stego message x' is separated in x_a and x'_b. The conditioned probability distribution $\widehat{P}_{X_b|X_a}(x'_b|x_a)$ is calculated, then the entropy encoder process x'_b according to the model distribution. The encoder output is the embedded message.

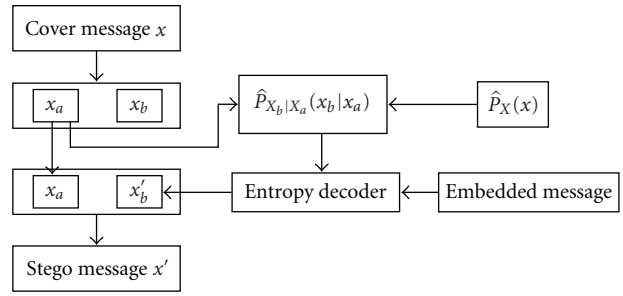

Figure 1: Model-based embedding scheme.

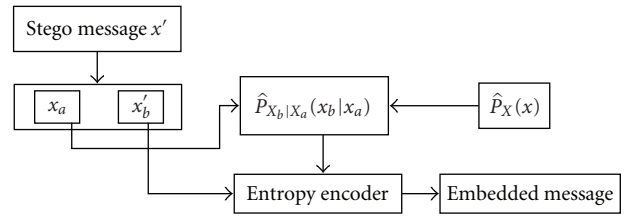

Figure 2: Model-based extraction scheme.

From now on, since the main focus of this work is on hiding information in images, the cover messages will be denoted as cover images.

2.2. Selection of the Modifiable Coefficient Set. The JPEG compression codes the images by dividing them in blocks, calculating DCT coefficients for every block, and then performing a coefficient quantization. Thus, the quantization makes it impossible to get the original image after the compression. This is an issue for steganography, since hiding the message in the spatial domain should take into account this information loss. Instead, embedding the message in DCT domain permits to avoid this issue. Hence, X_b is selected as a subset of quantized DCT coefficients. The modifiable coefficients are accurately selected in order to preserve the visual and the statistical characteristics of the cover image. The selection consists in three steps: in the first step a preliminary coefficient exclusion is performed, in the second step the maximum number of modifiable coefficients per block is calculated, and then in the final step the modifiable coefficients are selected.

2.2.1. Preliminary Coefficients Exclusion. At first, some of the coefficients are excluded from embedding because of their visual or statistical relevance. This set includes

(i) DC coefficients;
(ii) zero-valued coefficients;
(iii) highly quantized DCT frequencies;
(iv) unitary coefficients.

DC coefficients are excluded from embedding because of their visual relevance, since they represent the mean

luminance value of a block. Zero-valued coefficients are also excluded, since they occur in featureless areas of the image where changes are most likely to create visible artefacts. All the highly quantized DCT frequencies (whose quantization coefficients are greater of a threshold $T = 15$) are discarded during the embedding because small changes in these coefficients result in large alterations in the respective dequantized coefficients. Moreover, unitary coefficients $(-1, +1)$ are also excluded from embedding; experimental results illustrated in Section 3.2.3 show that modifying unitary coefficients increases detectability.

The residual coefficient set is denoted by $\hat{x}_b$. Moreover, for every block m, let P_m denote the number of remaining coefficients in the block.

2.2.2. Coefficient Modification. Every DCT coefficient, according to its value, is represented by a group and an offset. Denoting b the DCT coefficient, its group $g(b)$ is calculated through the following expression:

$$g(b) = \text{sign}(b) \cdot \left(\left\lfloor \frac{|b|}{2} \right\rfloor \right), \quad |b| > 1. \tag{1}$$

Thus all the groups are disjoint and have two elements which differ only in one unitary value, for example, $\{2, 3\}, \{6, 7\}, \{-4, -5\}$, and so forth.

The coefficient offset $O(b)$ is defined by the following expression:

$$O(b) = |\, b - 2 \cdot (g(b)) \,| + 1, \quad |b| > 1, \tag{2}$$

thus offsets can be only 1 or 2. PSB embeds the message by changing modifiable coefficient offsets, thus only unitary increments/decrements are possible, for example, a coefficient whose value is 3, after embedding could be only 2 or 3 (its group is $\{2, 3\}$). Offsets are modified according to the model.

2.2.3. Discrepancy. Some areas of the image could not be suitable to embed the message (e.g., a periodic texture, a sharp area, and so forth where changes could be more detectable), but a first-order statistic modeling is not able to discriminate such areas. A new measure is introduced, named discrepancy, to derive the embedding suitability of an area. The discrepancy is calculated at block layer and expresses how much a block is similar to adjacent blocks. In PSB, discrepancy is used to determine the maximum number of modifiable coefficients within a block.

Block B_0 discrepancy is an approximation of the mean value of the $L1$-distance, calculated in DCT domain, between block B_0 and block B_j, $j = 1, \ldots, 4$, where B_j is one of the blocks shown in Figure 3:

$$S_0 = \frac{\sum_{j=1}^{4} \sum_{i=1}^{64} q^i \, |\, \hat{b}_0^j - \hat{b}_i^j \,|}{4}, \tag{3}$$

being S_0 the discrepancy, q^i the quantization coefficient of the ith DCT frequency. $\hat{b}_i^j$ assumes the following expression:

$$\hat{b}_i^j = \begin{cases} b_i^j & \text{if } b_i^j \notin \hat{x}_b, \\ 2 \cdot g(b_i^j) & \text{if } b_i^j \in \hat{x}_b, \end{cases} \tag{4}$$

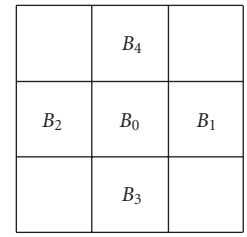

Figure 3: Block neighborhood.

where b_i^j is ith quantized DCT coefficient of jth block. Since the embedding modifies the exact $L1$-distances from the blocks, and the sender, and the receiver must calculate the same discrepancy in order to extract the embedded message, discrepancy is not calculated as the exact mean, thus the approximation (3) is required. If the block B_0 is on image border the discrepancy is calculated taking into account only existing blocks.

Since discrepancy is larger when blocks are different, a block is suitable for embedding when it has a large discrepancy. Numerical simulations show that the discrepancy calculated in random pixel images is 4284 on average. Assuming that steganography works better on random pixel images, PSB divides the interval $[0, 4284)$ in 63 subintervals labeled from 0 to 62: $[0, 68)$ is 0, $[68, 136)$ is $1, \ldots,$ $[4216, 4284)$ is 62, and $[4284, \infty)$ is 63. Let M_m denote the label from block m then

$$M_m = \begin{cases} \left\lfloor \dfrac{S_m}{68} \right\rfloor & \text{if } S_m < 4284, \\ 63, & \text{elsewhere}, \end{cases} \tag{5}$$

M_m represents the maximum number of modifiable coefficients for block m according to discrepancy, but without considering the preliminary exclusions illustrated in Section 2.2.1. Therefore the actual maximum number of modifiable coefficients for block m, N_m, is calculated through the following expression:

$$N_m = \min(M_m, P_m). \tag{6}$$

If $M_m < P_m$ the coefficients are selected from $\hat{x}_b$ by a pseudo random noise generator (PRNG) seeded by the stegokey.

The class of the remaining coefficients after the random selection is denoted by x_b. (Even if x_b should represent the class of the remaining coefficients offsets, to lighten the notation it will denote the entire coefficients.)

2.3. Message Embedding. The offsets of the coefficients belonging to x_b are replaced according to the message and the model described in the next sections.

2.3.1. Coefficient Permutation. The embedded message is scattered across the image using a PRNG seeded by the stegokey that permutes the order of the modifiable coefficients. As reported in [2], it represents a good solution to spread the embedded message in the whole image, both in spatial and in DCT domains.

2.3.2. The Peak-Shaped Model. The peak-shaped model is a first-order model characterizing DCT frequency histograms. The model is dependent on the stegokey and therefore an attacker is not able to calculate it exactly.

The model is based on two heuristic assumption derived from F5 steganography [2]:

$$h(b) > h(b+1), \quad b \geq 0,$$
$$h(b) - h(b+1) > h(b+1) - h(b+2), \quad b \geq 0, \tag{7}$$

being h the histogram of a fixed DCT AC frequency and b a positive DCT coefficient. Similar properties apply on negative coefficients. For sake of simplicity the model is described only for positive coefficients on a fixed DCT AC frequency, but equivalent steps hold also from negative coefficients and all the DCT AC frequencies.

The peak-shaped model characterizes offset probabilities for the groups by exploiting (7). Let h' denote the stego image histogram (for a fixed DCT AC frequency) and $i > 0$ a coefficient group:

$$h'(2i) = \left[h(2i) + h(2i+1)\right] \cdot P_i,$$
$$h'(2i+1) = \left[h(2i) + h(2i+1)\right] \cdot (1 - P_i), \tag{8}$$

where P_i is the first offset probability conditioned to group i. Let $H_g(i) \doteq h(2i) + h(2i+1)$, $i > 0$ denote the group histogram and assuming $H_g(i) > H_g(i+1)$, then (7) leads to

$$0.5 \leq P(i) \leq 1. \tag{9}$$

Defining $k_i \doteq P(i) - 0.5$, the stego image histogram is

$$h'(2i) = H_g(i)(k_i + 0.5),$$
$$h'(2i+1) = H_g(i)(0.5 - k_i). \tag{10}$$

From (7) and (10), the simple algebra calculations lead to

$$k_i > \frac{0.5 \cdot \left(H_g(i) - H_g(i+1)\right) - k_{(i+1)} H_g(i+1)}{3H_g(i)}, \tag{11}$$

$$k_i < \frac{0.5 \cdot \left(H_g(i) - H_g(i+1)\right) - 3k_{(i+1)} \cdot H_g(i+1)}{H_g(i)}. \tag{12}$$

By exploiting (11) and (12) it is possible to find an iterative algorithm to obtain k_i. However, (11) and (12) are not always satisfied in conjunction, but only when

$$k_{i+1} < \frac{H_g(i) - H_g(i+1)}{8H_g(i+1)}. \tag{13}$$

Finally, k_i is calculated recursively starting with the largest group i following the algorithm illustrated by the flow chart in Figure 4.

(i) For $i > 5$, $k_i = 0$ since large coefficients are not statistically relevant [8]. Moreover, Figure 5 shows the deviation of the offset distribution per group from a uniform offset distribution at the DCT frequency $(0, 1)$ averaged on an image database: groups with $i > 5$ show a little deviation from the uniform distribution. In addition, it maximizes the embedding capacity for these groups.

(ii) For $i \leq 5$ if $H_g(i) \leq H_g(i+1)$ or $H_g(i) = 0$ then $k_i = k_{i+1} + 0.05$. If (13) is not satisfied, the inferior limit expressed by (11) is assumed to be 0. k_i is derived by a PRNG (pseudo random noise generator) seeded by the stegokey according to (11) and (12).

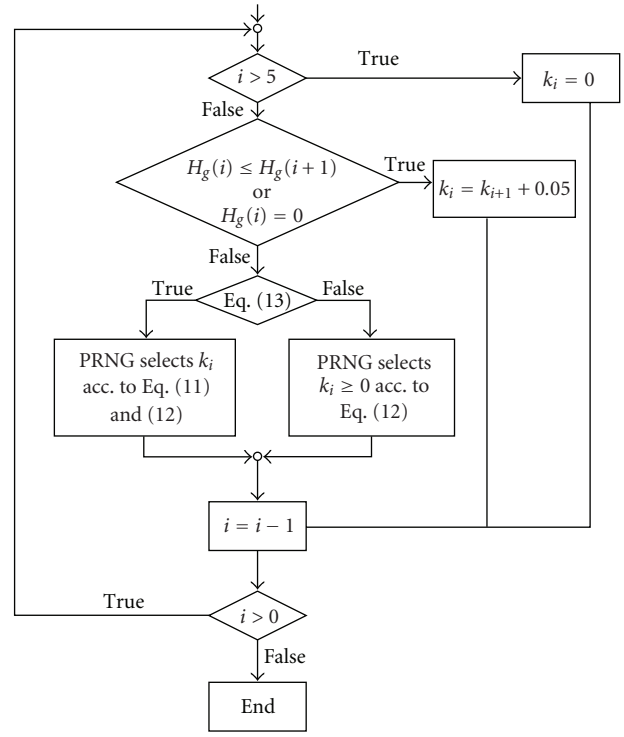

Figure 4: Peak-shaped model outline.

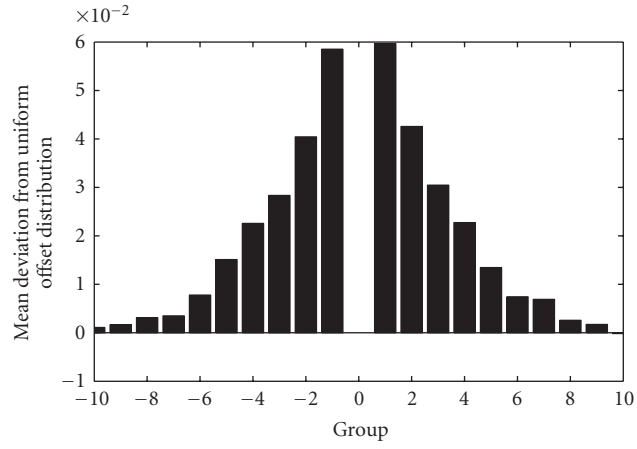

Figure 5: Offset deviation from uniform distribution at the DCT frequency $(0, 1)$.

2.4. Algorithm Summary. A summary of the embedding and the extraction algorithm is illustrated in the following.

2.4.1. Embedding Outline. The embedding algorithm follows the steps listed as follows:

(i) a header is added to the embedded message: the header is formed by two parts, one of fixed length (5 bits) and one of variable length, whose dimension is written in the fixed part. Message length is written into the variable part;

(ii) a preliminary exclusion of non-modifiable coefficients (as described in Section 2.2.1) is performed and P_m is calculated for every block m;

(iii) discrepancy is calculated according to (3), and M_m is derived for every block m;

(iv) the maximum number of modifiable coefficients per block is calculated through (6);

(v) x_b is derived by selection of the modifiable coefficients for each block using PRNG if $M_m < P_m$;

(vi) a permutation of modifiable coefficients is performed by the PRNG;

(vii) the offset probabilities are calculated for every modifiable coefficient according to the model;

(viii) the embedded message is processed by the arithmetic decoder illustrated in [5, 9] according to the order established above;

(ix) the modifiable coefficient offsets are replaced by the output of the arithmetic decoder.

2.4.2. Extraction Outline. The extraction algorithm follows the steps listed as follows:

(i) a preliminary exclusion of non-modifiable coefficients (as described in Section 2.2.1) is performed, and P_m is calculated for every block m;

(ii) discrepancy is calculated according to (3), and M_m is derived for every block m;

(iii) the maximum number of modifiable coefficients is calculated through (6);

(iv) x_b is derived by selection of the modifiable coefficients for each block using PRNG if $M_m < P_m$;

(v) a permutation of modifiable coefficients is performed by the PRNG;

(vi) the offset probabilities are calculated for every modifiable coefficient according to the model;

(vii) the message is obtained by encoding the offsets using the arithmetic encoder [5, 9];

(viii) the header is inspected so as to read the message length and to extract the message.

Table 1: PSNR test result.

PSNR	min	mean	max
MB1	34.2 dB	40 dB	43.6 dB
MB2	35.2 dB	39.9 dB	44.3 dB
PSB	34.2 dB	40.4 dB	46.6 dB

2.5. Embedding Capacity. Embedding capacity is defined as the maximum mean message length which could be embedded in an image.

A modifiable coefficient b can hold as many bits as the entropy of the binary alphabet associated to its group $g(b)$:

$$C_b = P_{g(b)} \log_2 \frac{1}{P_{g(b)}} + (1 - P_{g(b)}) \log_2 \frac{1}{(1 - P_{g(b)})}, \quad (14)$$

where $P_{g(b)}$ is the probability of the first offset conditioned to group $g(b)$. So the embedding capacity is

$$C = \sum_{b \in X_b} C_b. \quad (15)$$

3. Experimental Results

To test the validity of this technique, PSB is compared to the original Model-based steganography (MB1 and MB2, described in [5]). Two experiments are performed: in the first experiment the visual degradation in the image introduced by the steganography is evaluated by calculating PSNR; in the second experiment the state-of-the-art steganalytical test [10] is performed to compare the robustness of the techniques.

These test are carried out on an image database that contains 2000 images taken from BOWS-2 database [11]. All the images are natively in lossless format and gray-scaled. Image dimensions are 512×512 pixels. The images are converted in JPEG format with a fixed quality factor equal to 80.

3.1. PSNR Evaluation. This experiment is performed by embedding the same message for the three techniques and then evaluating PSNR. The message length is different among the images and equals to the PSB embedding capacity, which is the smallest among the techniques, because of PSB unitary coefficients exclusion. PSNR results are shown in Table 1: PSB achieves slightly higher PSNR with respect to MB1 and MB2 (0.5 dB higher); moreover PSNR is adequate to ignore the visual degradation introduced by the three techniques. The degradation introduced by MB2 blockiness compensation is negligible.

3.2. Steganalytical Test. PSB detectability is compared to MB1 and MB2 by means of the state-of-the-art steganalytical test [10].

3.2.1. Test Overview. Following [10] the evaluation is performed as follows:

(i) the image database is split in a training set (1300 images) and a testing set (700 images);

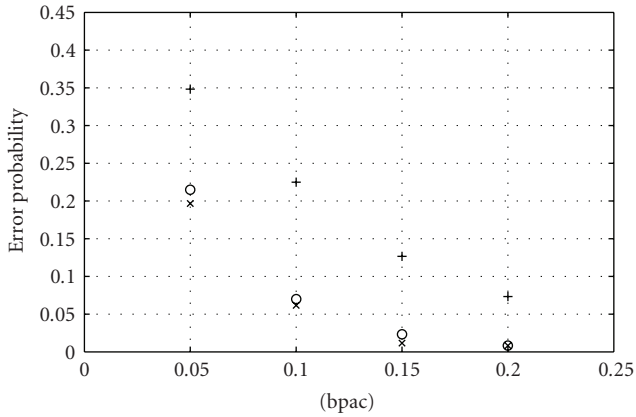

Figure 6: Experimental results at various bpac (circle: MB1, cross: MB2, plus: PSB).

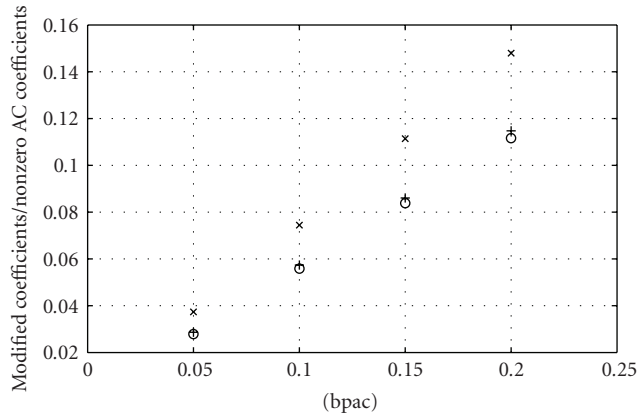

Figure 7: The embedding impact on AC coefficients (circle: MB1, cross: MB2, plus: PSB).

(ii) the embedded message is the same for all the three techniques but it differs among the images. The message length for a given image is set as a fixed percentage of the image nonzero AC DCT coefficients (bpac-bit per nonzero AC coefficient). The following [10] experiments are performed at 0.05, 0.1, 0.15, 0.2 bpac;

(iii) no header is added to the message since it is negligible for the aim of the test;

(iv) both the test images and the train images are analyzed by the steganalyzer without the embedded message (as cover images) and with the embedded message (as stego images);

(v) the support vector machine (SVM) [12, 13] is trained with the features of the training set scaled in $[-1, +1]$;

(vi) the SVM parameters C and γ are estimated by a five-fold cross-validation.

The simulation outcome is expressed by the error probability P that is the minimal total average error probability [10] on the testing set:

$$P = 0.5 \cdot (P_{\mathrm{FA}} + P_{\mathrm{MD}}), \tag{16}$$

where P_{FA} and P_{MD} are the probability of false alarm and missed detection, respectively. The aim of a steganographic technique is in achieving a high error probability.

3.2.2. PSB Steganalysis. Figure 6 shows test results: it is noticeable that PSB outperforms MB1 and MB2 at every bpac. Indeed, PSB error probability is about 0.13 higher than MB1 error probability. At 0.05 bpac PSB achieves about 0.35 error probability whereas MB1 and MB2 error probabilities are about 0.22. At higher bpac, all the techniques get lower error probabilities: at 0.2 bpac MB1 and MB2 are always detected, in fact the both get 0.008 error probability, instead of PSB error probability which is near 0.07.

Figure 7 shows the embedding impact as the mean (among the images) of the ratio between the number of modified coefficients and the total number of nonzero AC coefficients. PSB replaces a few more coefficients than MB1 but it gets lower visual degradation and larger error probability. Moreover MB2 has the major embedding impact on AC coefficients due to the addictional changes to preserve blockiness.

Table 2: Comparison between PSB+1 and the other techniques: error probability.

bpac	0.05	0.1	0.15	0.2
PSB+1	0.31	0.17	0.08	0.03
PSB	0.35	0.22	0.12	0.07
MB1	0.21	0.07	0.02	0.008

By comparing the embedding impact to the error probability and PSNR it results that the embedding impact has a minor relevance with respect to the selection of the modifiable coefficients. In fact, PSB outperforms MB1 in error probability and gets similar PSNR with a larger embedding impact. These superior performances are achieved by taking into account discrepancy and quantization matrix in order to select the modifiable coefficients set. MB2 modifies additional coefficients to preserve a superior-order statistical measure, but the additional coefficients to be replaced are not selected carefully, getting the worst performances.

3.2.3. Unitary Coefficient Exclusion. Since unitary coefficients are the most common coefficient values except for 0, their exclusion affects embedding capacity, but on the other hand modifying unitary coefficients increases detectability. In fact, Table 2 shows error probability for a modified PSB including unitary coefficient values, denoted by PSB+1 (groups and model are modified to include unitary coefficient values). PSB and MB1 are also included for sake of readability. It can be seen that unitary coefficient values exclusion increases PSB error probability by approximately 0.04 at bpac minor than 0.2, motivating their exclusion. At bpac larger than 0.2 both PSB, MB1 and PSB+1 get a zero error probability, hence it no longer makes sense the

Table 3: Comparison between PSB+1 and PSB: modified coefficients/nonzero AC coefficients.

bpac	0.05	0.1	0.15	0.2
PSB+1	0.028	0.057	0.086	0.115
PSB	0.028	0.058	0.086	0.117

Table 4: Error probability at different quality factors.

Quality factor	70		90	
bpac	0.05	0.1	0.05	0.1
PSB	0.32	0.22	0.35	0.22
MB1	0.22	0.08	0.17	0.04
MB2	0.21	0.07	0.14	0.04

embedding. Therefore, the unitary coefficient values impact on embedding capacity is negligible.

Moreover, Table 3 shows the embedding impact of PSB+1 with respect to PSB. Both the techniques achieve the same embedding impact, whereas PSB+1 gets lower error probability. This is a further confirm to the minor relevance of the embedding impact with respect to the selection of suitable coefficients.

3.2.4. Error Probability at Different Quality Factors. Usually JPEG quality factors used in storage are included in the interval $(70, 90)$ that is a good trade-off between quality and file size. Hence in the previous experiments the quality factor is set to 80. Moreover, in [8] the quality factor is set to 80, whereas in [10] and in [4] is set, respectively, to 75 and 70. Although the quality factor choice is arbitrary, the steganographic detectability could be affected by the different quantization, so some experiments are made to test PSB detectability with different quality factor. The results are illustrated in Table 4. PSB outperforms MB1 and MB2 at all the quality factors. Furthermore, PSB error probability, together with MB1 and MB2 error probability, is affected only partially by the quality factor. In fact at 0.05 bpac the error probabilities at the two quality factors differ only in 0.03, whereas at 0.1 bpac the error probabilities are the same. MB1 and MB2 show a larger difference, in particular MB2 error probabilities at 0.05 bpac differ in 0.07. Interesting enough, PSB undetectability improves at the quality factor increase, instead of MB1 and MB2 that show the opposite behavior.

4. Conclusions

A new Model-based technique, named peak-shaped-based steganography, is introduced in order to improve the original Model-based steganography. PSB novelty is in a more accurate coefficient selection, taking into account quantization and coefficient relevancy. A novel block measure, named discrepancy, is introduced to describe how much a block is suitable to embed a message. PSB model derives from heuristic hypothesis about histogram shape, moreover the model depends on the stegokey, therefore an attacker cannot calculate exactly the model. The message is scattered in the image by a PRNG seeded by the stegokey. The technique is evaluated by calculating the PSNR on an image database and performing the state-of-the-art steganalytical test described in [10]. In each test PSB outperforms the original Model-based techniques. It is also shown that the embedding impact (how many coefficients are modified during the embedding) results having minor relevance with respect to the selection of the areas in which the message is embedded.

Future work on JPEG steganography are directed toward a superior-order modeling of the DCT coefficients, by studying Markov Random Fields and the effect of image noise in DCT domain. In particular, since unitary coefficients modification affects detectability, they are actually excluded from the embedding. However, their exclusion decreases embedding capacity. The authors believe that if a more accurate model is used, unitary coefficients could be included to increase the capacity with no detectability increase.

Acknowledgment

The authors would like to thank Patrick Bas and Teddy Furon for making the BOWS-2 database available.

References

[1] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE Journal of Selected Area in Communications*, vol. 16, no. 4, pp. 474–481, 1998.

[2] A. Westfeld, "F5—a steganographic algorithm," in *Proceedings of the 4th International Workshop on Information Hiding (IH '01)*, vol. 2137 of *Lecture Notes in Computer Science*, pp. 289–302, Pittsburgh, Pa, USA, April 2001.

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

[4] J. Fridrich, T. Pevný, and J. Kodovský, "Statistically undetectable JPEG steganography: dead ends challenges, and opportunities," in *Proceedings of the 9th Workshop on Multimedia & Security (MM&Sec '07)*, pp. 3–14, Dallas, Tex, USA, September 2007.

[5] P. Sallee, "Model-based steganography," in *Proceedings of the International Workshop on Digital Watermarking (IWDW '03)*, pp. 154–167, Seoul, Korea, October 2003.

[6] J. Fridrich, M. Goljan, and D. Hogea, "Attacking the outguess," in *Proceedings of the 10th ACM Workshop on Multimedia & Security (MM&Sec '02)*, pp. 3–6, Juan-les-Pins, France, December 2002.

[7] R. Böhme and A. Westfeld, "Breaking Cauchy model-based JPEG steganography with first order statistics," in *Proceedings of the 9th European Symposium on Research in Computer Security (ESORICS '04)*, pp. 125–140, Sophia Antipolis, France, September 2004.

[8] J. Fridrich, "Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes," in *Proceedings of the 6th International Workshop on Information Hiding (IH '04)*, pp. 67–81, Toronto, Canada, May 2004.

[9] I. Witten, R. Neal, and J. Clearly, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[10] J. Fridrich and T. Pevny, "Merging Markov and DCT features for multi-class JPEG steganalysis," in *Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505 of *Proceedings SPIE*, pp. 3–4, San Jose, Calif, USA, January 2007.

[11] BOWS-2 database (clean images), http://dud.inf.tu-dresden.de/~westfeld/rsp/.

[12] C. Chang and C. Lin, LIBSVM: a library for support vector machines, Software, 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[13] C. Hsu, C. Chang, and C. Lin, "A Practical Guide to Support Vector Classification," 2007, http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.