A NEW COST FUNCTION FOR SPATIAL IMAGE STEGANOGRAPHY

Bin Li[1] *, Ming Wang*[1]*, Jiwu Huang*[1]*, and Xiaolong Li*[2]

[1]College of Information Engineering, Shenzhen University, Shenzhen, GD 518060, China
[2]Institute of Computer Science and Technology, Peking University, Beijing 100871, China

ABSTRACT

A well defined cost function is crucial to steganography under the scenario of minimizing embedding distortion. In this paper, we present a new cost function for spatial image steganography. The proposed cost function is designed by using a high-pass filter to locate the less predictable parts in an image, and then using two low-pass filters to make the low cost values more clustered. Experiments show that the steganographic method with the proposed cost function makes the embedding changes more concentrated in texture regions, and thus achieves a better performance on resisting the state-of-the-art steganalysis over prior works, including HUGO, WOW, and S-UNIWARD.

Index Terms— Additive distortion, cost function, steganography, steganalysis

1. INTRODUCTION

Steganography aims to hide information in stego media without drawing suspicion from steganalysis [1][2]. In order to minimize statistical detectability, steganography can be formulated as a source coding problem that minimizes embedding distortion [3][4]. Although the distortion is non-additive in nature, additive distortion function is often used because of its simplicity and also due to the fact that non-additive distortion could be approximately expressed in an additive form [5]. In an additive setting for images, distortion is evaluated by summing up *costs* which quantify the effect of making modification to individual pixels. Hence, a better defined cost function would result in a more secure steganographic method.

The cost function of HUGO (highly undetectable stego) [6] computes the weighted sum of difference between feature vectors extracted from a cover image and its stego version in SPAM (subtractive pixel adjacency matrix) [7] feature space. In this way, a pixel after modification which makes the feature vector more deviated will have a higher cost. The embedding changes of HUGO will be made within texture regions and along edges. WOW (wavelet obtained weights) [8] assigns high costs to where the image contents are more predictable by directional filters. Thus clean edges are avoided for modification. And hence it has improved security performance over HUGO under the detection by the powerful steganalysis which employs SRM (spatial rich models) [9]. S-UNIWORD (spatial universal wavelet relative distortion) [10] has a slightly modified cost function and thus has a similar performance compared to WOW. Its cost function can be extended to JPEG images.

The state-of-the-art methods, WOW and S-UNIWORD, have exploited more pixels in texture areas for hiding data. However, some pixels in texture areas, which may be suitable for carrying data, are assigned with high costs. Intuitively, embedding changes made in such pixels should be more secure than in pixels located in smooth areas. In this sense, the cost function can be further improved. Our work develops a new cost function, ensuring all pixels within textural regions have relatively low costs. The new proposed cost function is realized by using a high-pass filter and two low-pass filters, making more embedding changes concentrated in textural areas. Experimental results show that the steganographic method using the proposed cost function performs better than HUGO, WOW, and S-UNIWARD, in resisting steganalysis with SRM [9] and steganalysis with MCNUQ (mapping co-occurrence using non-uniform quantization) [11].

The rest of this paper is organized as follows. After introducing notations, we review the preliminaries on minimizing additive distortion and the cost function of WOW in Section 2. In Section 3 we present a new cost function which is improved on WOW. Results of comparative experiments are given in Section 4 to demonstrate the effectiveness of the proposed cost function. Conclusions are drawn in Section 5.

2. PRELIMINARIES AND PRIOR WORK

2.1. Notations

Throughout the paper, the symbols $\mathbf{X} = (x_{i,j})^{n_1 \times n_2}$, $\mathbf{Y} = (y_{i,j})^{n_1 \times n_2}$ are respectively used to denote an 8-bit gray-scale cover image (of size $n_1 \times n_2$) and its stego version. We restrict our design for the ternary embedding case, where $y_{i,j} \in \mathcal{I}_{i,j} = \{\min(x_{i,j} - 1, 0), x_{i,j}, \max(x_{i,j} + 1, 255)\}$. We use

Thanks to NSFC (61103174, 61100169, 61332012, U1135001), Fundamental Research Program of Shenzhen City (JCYJ20120613113535357, JCYJ20120613115442060), and Foundation for Distinguished Young Talents in Higher Education of Guangdong (2012LYM0117) for funding.

$|\mathbf{H}|$ to denote the matrix containing the absolute values of the elements of a matrix $\mathbf{H}$, and $\frac{1}{\mathbf{H}}$ the matrix containing the reciprocals of each elements of $\mathbf{H}$. The notation $\otimes$ and $\odot$ respectively stand for the mirror-padded convolution and mirror-padded correlation.

2.2. Minimizing Additive Distortion

It is assumed that the impacts of the embedding changes are mutually independent under an additive distortion scenario. To evaluate the embedding impact of modifying a pixel $x_{i,j}$ to $y_{i,j}$, a quantity called *cost*, denoted by $\rho_{i,j}(\mathbf{X}, y_{i,j})$, is used. To simplify the design, we suppose $\rho_{i,j}(\mathbf{X}, x_{i,j}) = 0$ and $\rho_{i,j}(\mathbf{X}, x_{i,j} - 1) = \rho_{i,j}(\mathbf{X}, x_{i,j} + 1) = \varrho_{i,j} \in [0, +\infty)$. The overall distortion to the image is as follows [4][8]:

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \varrho_{i,j} |x_{i,j} - y_{i,j}|. \tag{1}$$

Let m be the total amount of secret payload that would be conveyed. Denote $\pi(y_{i,j})$ as the probability of modifying $x_{i,j}$ to $y_{i,j}$. To minimize the embedding distortion (1), one can simulate *optimal embedding* [4],[5] by assigning

$$\pi(y_{i,j}) = \frac{\exp(-\lambda \rho_{i,j}(\mathbf{X}, y_{i,j}))}{\sum\limits_{y_{i,j} \in \mathcal{I}_{i,j}} \exp(-\lambda \rho_{i,j}(\mathbf{X}, y_{i,j}))}, \tag{2}$$

where λ is a parameter used to satisfy the payload constraint

$$m = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{y_{i,j} \in \mathcal{I}_{i,j}} \pi(y_{i,j}) \log \frac{1}{\pi(y_{i,j})}. \tag{3}$$

Some practical codes are available to approach the performance of optimal embedding. For example, STC (syndrome-trellis code) [12] is commonly used.

2.3. Cost Function in WOW

The cost function of WOW [8] is designed with the help of a group of directional filters, which are denoted by $\mathbf{D}^{(k)}$ $(k = 1, \cdots, n)$. Define a quantity called *embedding suitability* and denote it by $\xi_{i,j}^{(k)}$. It is computed as the *weighted* absolute values of the *filter residual differences* between a cover image and the image after changing only one pixel $x_{i,j}$ to $y_{i,j}$. Since the absolute values of the filter residuals are selected as weights, and the filter residual differences have the same form as a rotated directional filter, the embedding suitabilities can be computed by

$$\boldsymbol{\xi}^{(k)} \triangleq (\xi_{i,j}^{(k)})^{n_1 \times n_2} = |\mathbf{X} \otimes \mathbf{D}^{(k)}| \odot |\mathbf{D}^{(k)}|. \tag{4}$$

The cost value is obtained by aggregating all the reciprocals of embedding suitabilities, i.e.,

$$\boldsymbol{\varrho} \triangleq (\varrho_{i,j})^{n_1 \times n_2} = \sum_{k=1}^{n} \frac{1}{\boldsymbol{\xi}^{(k)}}. \tag{5}$$

The underlying assumption is that if the filter residual is small in one of the directions, the corresponding pixel is predictable, and thus should be assigned a high cost. The Daubechies 8-tap (DB-8) wavelet filters are used as $\mathbf{D}^{(k)}$ in (4) and can achieve a good performance as reported in [8].

3. A NEW COST FUNCTION

3.1. An Improved and Generalized Cost Function

We show a cover image in Fig. 1 (a). Using the cost function of WOW, we observe from Fig. 1 (b) that there are some pixels with high cost values (and thus low change probabilities) inside a texture region. This is because the pixels may be predictable in one of the directions. However, we should further differentiate the pixels with high cost values in texture regions and smooth regions. That is, a pixel in a texture region, even being predictable in one of the directions, should be assigned a lower cost value (and thus having a higher probability of modification) than a pixel in a smooth region which is predictable in all directions. To this end, we can employ a low-pass filter to the result of (5) in order to spread the low costs of textural pixels to their neighbourhood. This is called *spreading rule* [13]. In fact, when the cost value of a pixel is weighted by its neighboring cost values, the mutual dependencies among cost values are taken into consideration.

Furthermore, we take two additional measures to make the cost function be more generalizable by using a new designed embedding suitability other than (4). Consider the two terms involved in the correlation operation in (4). First, the directional filter $\mathbf{D}^{(k)}$ can be replaced by any high-pass filter $\mathbf{H}^{(k)}$, regardless of whether directional or non-directional, to locate the less predictable area. This measure will enrich our choice of $\mathbf{H}^{(k)}$. Second, since the elements in the second term $|\mathbf{D}^{(k)}|$ are all non-negative, the filter $|\mathbf{D}^{(k)}|$ has a low-pass nature. Hence, we can substitute it with a low-pass filter to make it more flexible for use. Besides, when the low pass-filter is central symmetric, correlation can be replaced by convolution.

Based on the above analysis, we propose an improved cost function with the following steps:

1. Compute the new designed embedding suitability by first using a high-pass filter $\mathbf{H_1}$ to the cover image to obtain filter residuals, and then using a low-pass filter $\mathbf{L_1}$ to the absolute values of the resulting residuals, i.e.,

$$\boldsymbol{\zeta}^{(k)} = |\mathbf{X} \otimes \mathbf{H}^{(k)}| \otimes \mathbf{L_1}. \tag{6}$$

2. Compute the cost value by using a low-pass filter $\mathbf{L_2}$ to the aggregation of the reciprocals of the embedding suitabilities, i.e.,

$$\boldsymbol{\varrho} = (\sum_{k=1}^{n} \frac{1}{\boldsymbol{\zeta}^{(k)}}) \otimes \mathbf{L_2}. \tag{7}$$

(a) cover image (b) using (5) with DB-8 (c) using (7) with DB-8 (d) using (5) with KB (e) using (8) with KB

Fig. 1. The probabilities of embedding changes (from (b) to (e)) for cover image (a) with payload 0.4 bit per pixel using optimal embedding. The brightness in (b) to (e) is scaled and adjusted to $[0, 1]$, where 0 is the brightest (lowest probability), and 1 is the darkest (highest probability). Note that the probabilities of embedding changes of WOW correspond to (b). There are some scattered bright elements inside the small dark regions in (b) and (d).

The new designed embedding suitability is no longer the *weighted filter residual difference*, but the *smoothed filter residual*. It can be interpreted as using a high-pass filter and than a low-pass filter to locate the less predictable regions. In fact, a single non-directional high-pass filter can also achieve the goal of finding the less predictable areas. Hence the process of obtaining cost value can be further simplified as

$$\boldsymbol{\varrho} = \frac{1}{|\mathbf{X} \otimes \mathbf{H}^{(1)}| \otimes \mathbf{L_1}} \otimes \mathbf{L_2}. \tag{8}$$

3.2. Selection on Filters

There are three filters in our proposed cost function, and they will affect the overall performance. Since the 3×3 KB (Ker-Bohme) filter [14] has a superior performance in steganalysis, we select it as our high-pass filter in (8). It has a form as

$$\mathbf{H}^{(1)} = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

which is central-symmetric and rotation invariant. Note that the authors of WOW have also tested KB filter with their cost function (5), but did not achieve satisfactory performance. As will be shown in Section 4, with the proposed cost function, we can obtain better performance by the KB filter.

We use the average filter for both $\mathbf{L_1}$ and $\mathbf{L_2}$ because of its fast implementation. The best filter size can be determined experimentally, as reported in Section 4.2. We use 3×3 for $\mathbf{L_1}$ and 15×15 for $\mathbf{L_2}$ as default. Note that when the filter size is 1×1, it means that there is no filtering operation. In such a special case, we denote the filter as $\boldsymbol{\delta}$.

3.3. Visualizing the Probabilities of Embedding Changes

In Fig. 1, we present the probabilities of embedding changes for a cover image by using different cost functions ((5), (7) or (8)) with DB-8 filters or KB filter. It can be observed from Fig. 1 (b) and (d) that there are some pixels with low change probabilities in a textural region with (5). This phenomenon is more several in the case of KB filter than DB-8 filters. As a result, the cost function (5) with KB filter has an inferior performance than that with DB-8 filters. On the contrary, when we use (7) or (8), whatever the high-pass filter(s) is (are) used, the change probabilities are concentrated in larger clustered regions, as demonstrated in Fig. 1 (c) and (e). And their performances on resisting steganalysis are much better as reported in the next section.

4. EXPERIMENTS

4.1. Setups

All experiments in this section are conducted on BOSSbase ver.1.01 database [15] with an amount of 10,000 gray-scale images with size 512×512 pixels. We use the optimal embedding simulator as default for all algorithms. And the performances are evaluated by steganalyzers using 34,671-dimensional feature set SRM [9], and 6,000-dimensional feature set MCNUQ [11], with ensemble classifiers [16]. A number of 5,000 images are randomly selected for training, and the rest 5,000 images are used for testing. We report the testing error which computes the average of the false positive rate and false negative rate by 10 times of randomly splitting the training and the testing images.

4.2. Inspecting the Roles of Three Filters

To get a better understanding on whether the proposed three filters can effectively improve the security performance, we conduct some comparative experiments, where the results are demonstrated in Table 1. The stego images are generated by (7) using the corresponding filters with payload rate 0.4 bpp (bit per pixel). We use $RA(\cdot)$ to denote a 180° rotation of a matrix after it has been taken the absolute values of its elements (referred to as RA operation). Note that when the DB-8 filters, RA(DB-8) filters, and $\boldsymbol{\delta}$ are respectively utilized as the filters $\mathbf{H}^{(k)}$, $\mathbf{L_1}$, and $\mathbf{L_2}$, the proposed cost function is

Table 1. The performance on different filters by steganalyzer with SRM. The operation $RA(\cdot)$ stands for a 180° rotation of a matrix after it has been taken the absolute values of its elements. AVER stands for average filter.

$\mathbf{H}^{(k)}$	$\mathbf{L_1}$	$\mathbf{L_2}$	Testing Error
DB-8	$\boldsymbol{\delta}$	$\boldsymbol{\delta}$	0.1800
	RA(DB-8)	$\boldsymbol{\delta}$	0.2059
	3×3 AVER	$\boldsymbol{\delta}$	0.2166
	$\boldsymbol{\delta}$	15×15 AVER	0.2210
	RA(DB-8)	15×15 AVER	0.2307
	3×3 AVER	15×15 AVER	**0.2354**
KB	$\boldsymbol{\delta}$	$\boldsymbol{\delta}$	0.0749
	RA(KB)	$\boldsymbol{\delta}$	0.1290
	3×3 AVER	$\boldsymbol{\delta}$	0.1426
	$\boldsymbol{\delta}$	15×15 AVER	0.1263
	RA(KB)	15×15 AVER	0.2469
	3×3 AVER	15×15 AVER	**0.2498**

Table 2. The performance (testing error by steganalyzer with SRM) under different sizes of low-pass filters.

$\mathbf{L_1}$ \ $\mathbf{L_2}$	11×11	13×13	15×15	17×17	19×19
3×3	0.2476	0.2475	**0.2498**	0.2468	0.2444
7×7	0.2388	0.2402	0.2394	0.2390	0.2375
11×11	0.2305	0.2332	0.2334	0.2321	0.2323

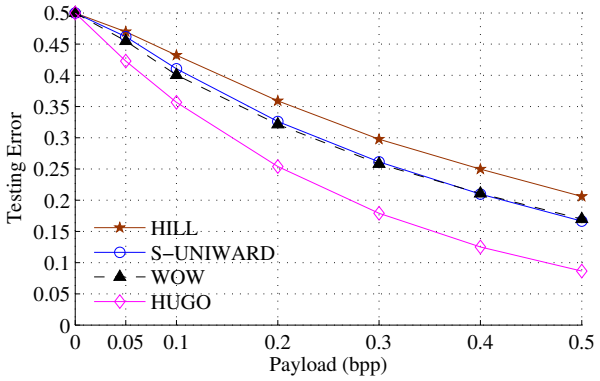
reduced to the cost function of WOW. And its performance is underlined in the table. From Table 1, we can observe that

- When $\mathbf{H}^{(k)}$ and $\mathbf{L_1}$ remain unchanged, using a filter $\mathbf{L_2}$ is always better than not using any low-pass filter.
- When $\mathbf{H}^{(k)}$ and $\mathbf{L_2}$ remain unchanged, using a filter $\mathbf{L_1}$ is always better than not using any low-pass filter.
- When at most one of $\mathbf{L_1}$ and $\mathbf{L_2}$ is used, or when neither $\mathbf{L_1}$ nor $\mathbf{L_2}$ is used, the directional filters DB-8 perform better than the high-pass filter KB. When $\mathbf{L_1}$ and $\mathbf{L_2}$ are both used, KB performs better than DB-8.

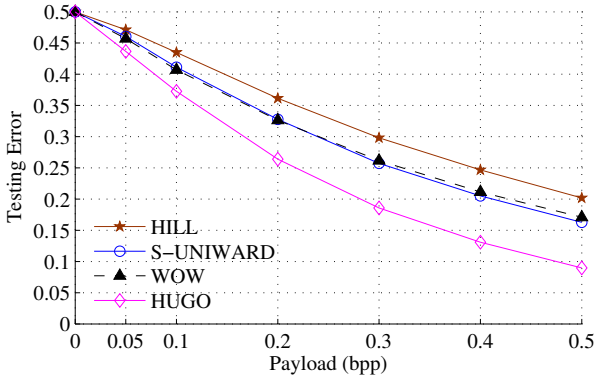
Since the two low-pass filters are critical to the new cost function, we perform an experiment to obtain the best filter size for $\mathbf{L_1}$ and $\mathbf{L_2}$, where KB filter is used as $\mathbf{H}^{(1)}$. The results on embedding payload rate of 0.4 bpp are shown in Table 2. It is suggested that a small (but larger than 1) size for $\mathbf{L_1}$ is preferred. And a medium size for $\mathbf{L_2}$ performs better.

4.3. Comparison to Prior Methods

We call the steganographic method using the proposed cost function (8) with the KB filter and the two average low-pass filters as HILL (**Hi**gh-pass, **L**ow-pass, and **L**ow-pass). We evaluate its performance under different embedding payload rate from 0.05 to 0.5 bpp. HUGO (binary embedding with parameters $\gamma = 1$, $\sigma = 1$, $T = 255$) [6], WOW [8], and S-UNIWARD [10] are used for comparison. It can be observed



(a) Steganalytic performance using SRM



(b) Steganalytic performance using MCNUQ

Fig. 2. Comparing statistical detectability

from Fig. 2 that HILL always performs the best under the two state-of-the-art steganalytic feature sets, i.e., SRM [9] and MCNUQ [11].

5. CONCLUSIONS

In this paper, we propose a new cost function which consists of a high-pass filter and two low-pass filters. With the new cost function, the situation of some pixels with high cost-values being inside an textural region is avoided, and low cost values are clustered in a larger area. The steganographic method, HILL, which utilizes the proposed cost function, achieves a better performance over prior works against advanced steganalyzers. The fact that the coefficients in the three filters are all integers, and that only convolution operations are involved in the cost function, make a fast implementation of the steganographic method possible.

Since the the low-pass filtering operation can be regarded as taking the mutual dependencies among cost values into consideration and it leads to a better security performance, we will try to generalize the cost function to steganography that minimizes a non-additive distortion function in our future study.

6. REFERENCES

[1] T. Pevný and J. Fridrich, "Benchmarking for steganography," in *Proceedings of 10th International Workshop on Information Hiding*. 2008, vol. LNCS 5284, pp. 251–267, Springer Berlin Heidelberg.

[2] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, no. 2, pp. 142–172, April 2011.

[3] J. Fridrich, "Minimizing the embedding impact in steganography," in *Proceedings of the 8th ACM workshop on Multimedia and security*. 2006, pp. 2–10, ACM.

[4] J. Fridrich and T. Filler, "Practical methods for minimizing embedding impact in steganography," in *Proceedings of Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, Proc. SPIE*, 2007, vol. 6505, p. 650502.

[5] T. Filler and J. Fridrich, "Gibbs construction in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 705–720, 2010.

[6] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proceedings of 12th International Workshop on Information Hiding*, 2010, vol. LNCS 6387, pp. 161–177.

[7] T. Pevný, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, 2010.

[8] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *Proceedings of IEEE Workshop on Information Forensic and Security*, 2012, pp. 234–239.

[9] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, 2012.

[10] V. Holub and J. Fridrich, "Digital image steganography using universal distortion," in *Proceedings of the first ACM workshop on Information hiding and multimedia security*. 2013, pp. 59–68, ACM.

[11] L. Chen, Y. Q. Shi, S. Patchara, and X. Niu, "Non-uniform quantization in breaking hugo," in *Proceedings of 12th International Workshop on Digital-Forensics and Watermarking*, 2013.

[12] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3-2, pp. 920–935, 2011.

[13] B. Li, S. Tan, M. Wang, and J. Huang, "Investigation on cost assignment in spatial image steganography," *IEEE Trans. Inf. Forensics Security*, to appear.

[14] V. Holub and J. Fridrich, "Optimizing pixel predictors for steganalysis," in *Proceedings of Media Watermarking, Security, and Forensics, SPIE*, 2012, vol. 8303, pp. 09–1–09–13.

[15] P. Bas, T. Filler, and T. Pevný, "Break our steganographic system — the ins and outs of organizing boss," in *Proceedings of 13th International Workshop on Information Hiding*. 2011, vol. LNCS 6958, pp. 59–70, Springer Berlin Heidelberg.

[16] J. Kodovský, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 432–444, 2012.