Enhancing Steganography via Stego Post-processing by Reducing Image Residual Difference

Bolin Chen
Sun Yat-sen University
Guangzhou, China
chenbl8@mail2.sysu.edu.cn

Weiqi Luo*
Sun Yat-sen University
Guangzhou, China
luoweiqi@mail.sysu.edu.cn

Peijia Zheng
Sun Yat-sen University
Guangzhou, China
zhpj@mail.sysu.edu.cn

ABSTRACT

Most modern steganography methods focus on designing an effective cost function. To our best knowledge, there is no related works concerned about modifying stego to enhance steganography security. In this paper, therefore, we propose a novel post-processing for stego image in the spatial domain. To ensure the correct extraction of hidden message, our method restricts the modification amplitude of each pixel according to the characteristics of STCs (Syndrome-Trellis Codes). To enhance steganography security, our method traverses the stego image pixel by pixel, and modifies those pixels that can reduce the image residual difference between cover and stego under some criterion. Experimental results show that the proposed method can improve the security of current steganography especially for large payloads, e.g. larger than 0.3 bpp. In addition, the post-modification rate is rather low, for instance less than 8 ‰ pixels have been changed in the enhanced stego image for the five existing steganography methods for payload as large as 0.5 bpp.

KEYWORDS

Stego Post-processing, Steganography, Steganalysis

ACM Reference Format:
Bolin Chen, Weiqi Luo, and Peijia Zheng. 2019. Enhancing Steganography via Stego Post-processing by Reducing Image Residual Difference. In *ACM Information Hiding and Multimedia Security Workshop (IH&MMSec '19), July 3–5, 2019, Paris, France.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3335203.3335716

1 INTRODUCTION

Steganography is a science and art of concealing secret message into ordinary digital media without drawing suspicion. The study of steganography has received considerable attention during the past decade, and many effective steganography methods have been proposed until now.

The modern steganography is usually designed under the framework of distortion minimization, in which the cost function design is the key issue. For instance, HUGO [12] is the first work based on this framework, it defines the cost of each pixel using the weighted difference of the extended steganalytic feature sets SPAM [11]. WOW [6] assigns high costs to pixels that are more predictable by directional filter banks. UNIWARD [7] simplifies the cost computation of WOW and generalizes it in both spatial and DCT domain. HILL [9] employs a high-pass filter and two low-pass filters to define the cost function, which achieves a higher security performance than WOW and S-UNIWARD. MiPOD [13] computes cost according to the pixel variance estimated by multivariate Gaussian model, and achieves similar results as HILL. Please note that above methods are in an additive way, which assumes that all pixels in an image are independent. However, some non-additive methods such as CMD [10], Synch [2] and DeJoin [14] take into account the mutual impacts of pixels. They sequentially embed message into different parts of the image and dynamically update the cost to synchronize modification directions. Usually, non-additive methods achieve better security than additive ones.

Unlike steganography methods that focus on cost function design, in this paper, we try to improve the security of existing steganography in a totally different manner, that is, stego post-processing. Since most effective steganalytic feature sets such as SRM[5] are mainly derived from image residual analysis, the motivation of the proposed method is to reduce the image residual difference between cover and stego. It is expected that the smaller difference, the better security. In addition, in order to ensure that the secret message can be correctly extracted from the modified stego, we analyze the robustness of STCs [4] and restrict the post-modification amplitude for each image pixel. Based on extensive experimental results, we found that it is possible to achieve much better security performance by carefully modifying a small quantity of pixels within a stego, especially when the payload is large. To our best knowledge, this is the first work to enhance current steganography via stego post-processing.

The rest of the paper is organized as follows. Section 2 describes the STCs and its robustness against post-modification; Section 3 presents our method; Section 4 shows the experimental results and discussions. Finally, the conclusion remarks of this paper and future works are given in Section 5.

2 PRELIMINARIES

It is well known that the current steganography is constructed under the framework of distortion minimization, which usually includes two steps, that is, defining cost for every embedding unit (pixel/DCT coefficient) and embedding secret message with STCs. Since the proposed method is dependent on STCs, we will give a brief description of STCs and its robustness against post-modifications.

*Corresponding author.

IH&MMSec '19, July 3–5, 2019, Paris, France

ACM ISBN 978-1-4503-6821-6/19/06…$15.00
https://doi.org/10.1145/3335203.3335716

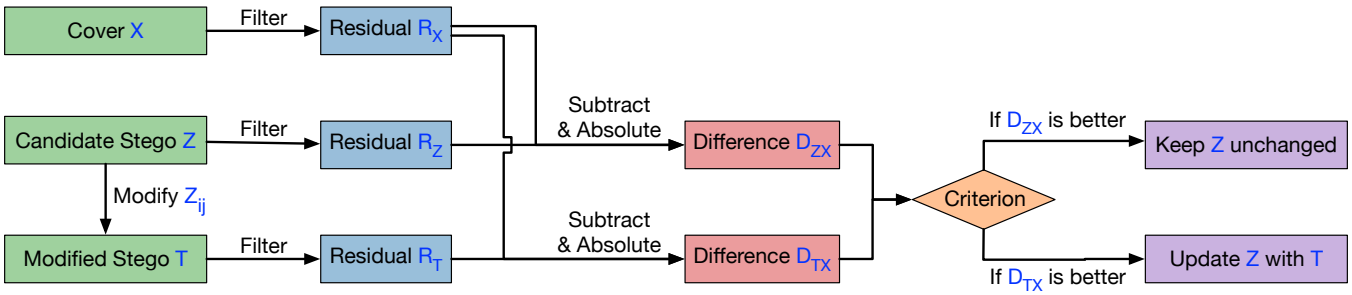

Figure 1: The proposed procedure of updating a target pixel within a stego image

2.1 Syndrome-Trellis Codes

STCs is a practical coding method, which can be used to solve the binary or non-binary embedding problem in steganography. For binary problem, the embedding and extraction of STCs can be formulated as follows:

$$Emb(X, m) = \arg \min_{P(Y) \in C(m)} D(X, Y) \tag{1}$$

$$Ext(Y) = \mathbb{H}P(Y) \tag{2}$$

where Emb is an embedding function, Ext is an extraction function; X is a cover image, Y is a stego image; m is a secret message; D is a distortion function; P is a parity function such as $P(Y) = Y \mod 2$; $\mathbb{H}$ is a parity-check matrix of a binary linear code C, $C(m) = \{z|\mathbb{H}z = m\}$ is the coset corresponding to syndrome m. Since $\mathbb{H}$ plays an important role in the embedding speed, STCs constructs $\mathbb{H}$ by placing a small submatrix along the main diagonal. In this way, equation (1) can be solved optimally by Viterbi algorithm with linear time and space complexity.

For non-binary embedding problem, STCs exploits multi-layered construction to solve it efficiently. Multi-layered STCs firstly decomposes the non-binary problem into a sequence of binary problems, and then uses the solution for binary problem as described above. It turns out that the non-binary problem can be solved optimally if each binary problem is solved optimally. Please refer to [4] for more details.

2.2 Robustness against Post-Modifications

From equation (2), we observe that if $\mathbb{H}$ is fixed, the message extraction completely relies on $P(y)$. Therefore, if there exists a modification matrix Δ such that $P(Y + \Delta) = P(Y)$, the messages extracted from $Y + \Delta$ and Y would be exactly the same, meaning that the STCs is robust against the modification Δ in this case. Generally, the parity function P of q-ary STCs returns the 1^{st} to k^{th} LSBs of the input matrix, where $k = \lceil \log_2 q \rceil$. Thus, q-ary STCs' robustness against post-modifications can be formulated as follows:

$$Ext(Y) = Ext(Y + \Delta), \quad \Delta_{ij} = 2^k \times n, n \in \mathcal{Z} \tag{3}$$

where Y and Δ are matrices of the same size $n_1 \times n_2$, T_{ij} denotes the ij^{th} element of modification matrix Δ

Taking a stego image Y obtained by ternary STCs (i.e. $q = 3$) for example, in this case, $k = \lceil \log_2 q \rceil = 2$, and $\Delta_{i,j} = 2^2 \times n = 4n$, $n \in \mathcal{Z}$. Thus, we conclude that adding a multiple of 4 to any pixel of the stego will not confuse the message extraction at all.

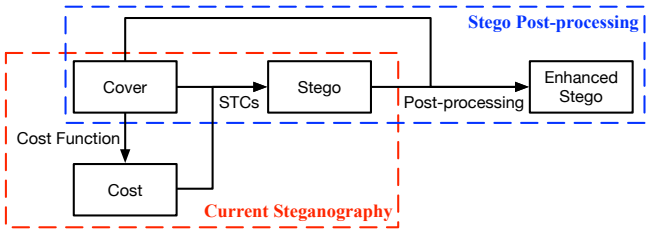

Figure 2: The current steganography vs. the proposed stego post-processing

3 PROPOSED METHOD

As illustrated in Fig. 2, the current steganography firstly defines costs for all pixels in a given cover, and then uses STCs to embed secret message into cover to obtain the resulting stego. While the proposed method tries to perform some post-processing on the stego obtained by the current steganography to improve the security by reducing the image residual difference between cover and stego. Since most current steganography methods, such as WOW and HILL, use ternary STCs for data embedding, only the ternary case is considered in this paper. Please note that it is easy to extend our method for other cases.

First of all, we will describe how to update a target pixel in a stego using the proposed method. As illustrated in Fig. 1, Let X denote cover; Z denote the candidate stego which is initialized as the corresponding stego Y of X using some steganography, and it may be updated in our method; T denote the temporary variable for the modified stego after changing a target pixel Z_{ij} of Z according to the rule as described in Section 2.2 (i.e. $T = Z, T_{ij} = T_{ij} + 4n, n \in \mathcal{Z}$). Please note that messages extracted from Z and T are exactly the same. To determine whether the modified stego T is better than the candidate one Z, we firstly we apply some high-pass filter to cover image X and two stego images Z, T to get the corresponding image residuals R_X, R_Z and R_T separately. And then we calculate the absolute difference between R_X, R_Z and R_T separately, i.e. $D_{ZX} = |R_Z - R_X|$, $D_{TX} = |R_T - R_X|$. Finally, we compare D_{ZX} and D_{TX} under some criterion to determine whether the candidate stego Z should be updated as T or not.

Table 1: Detection errors (%) of the proposed method using different combinations of filter and criterion. The best result for a specific payload is labeled with an asterisk(*). We underline those results which become poorer after using our method.

Criterion	Filter	SRM					maxSRMd2				
		0.1	0.2	0.3	0.4	0.5	0.1	0.2	0.3	0.4	0.5
-	-	43.37	35.83	29.82	24.59	20.30	37.49	30.75	26.04	21.70	18.10
Element-wise	A	43.43*	36.43*	30.80*	26.32*	22.12*	38.37*	31.92*	27.59*	23.71*	20.22*
	B	43.32	36.24	30.52	25.70	21.73	37.80	31.66	27.19	23.32	19.78
	C	43.23	35.84	29.68	24.64	20.24	37.43	30.76	26.16	21.83	18.30
	D	43.06	35.76	29.80	24.69	20.27	37.47	30.81	26.16	21.76	18.13
Sum	A	42.88	34.56	27.92	22.15	17.66	36.68	29.47	23.95	18.96	15.35
	B	43.38	36.21	30.77	25.59	21.02	38.06	31.59	26.89	22.53	18.44
	C	43.21	35.64	29.69	24.54	20.10	37.56	30.78	26.10	21.67	18.06
	D	43.31	36.41	30.76	25.66	21.84	38.13	31.89	27.36	23.55	19.82

Algorithm 1 Pseudo-code for the proposed method. Images X, Y, Z are of size $n_1 \times n_2$; Variable R denotes the image residual, D denotes the image residual difference. The for loop in Line 5 traverses all pixels row by row.

1: **Input:** cover image X; stego image Y; filter F
2: **Output:** enhanced stego images Z
3: Initialize $Z = Y$
4: $R_X = conv(X, F)$
5: **for** $i \in \{1, ..., n_1\},\ j \in \{1, ..., n_2\}$ **do**
6: **for** $\delta \in \{+4, -4\}$ **do**
7: **while** $(Z_{ij} + \delta \geq 0)\ \ \&\ \ (Z_{ij} + \delta \leq 255)$ **do**
8: $T = Z$
9: $T_{ij} = T_{ij} + \delta$
10: $R_Z = conv(Z, F)$
11: $R_T = conv(T, F)$
12: $D_{ZX} = |R_Z - R_X|$
13: $D_{TX} = |R_T - R_X|$
14: **if** D_{TX} is better than D_{ZX} **then**
15: Update $Z_{ij} = T_{ij}$
16: **else**
17: **break**
18: **end if**
19: **end while**
20: **end for**
21: **end for**
22: **return** Z

The proposed method will traverse the stego image pixel by pixel, and deal with each pixel as described above. The Pseudo-code for our method is shown in Algorithm 1. The proposed method includes three loops. In the first loop (i.e. line 5-21), it traverses all the pixels row by row. In the second loop (i.e. line 6-20), it considers different directions of modification to a pixel (positive or negative direction). In the third loop (i.e. line 7-19), it considers different modification amplitudes to a pixel (e.g. +4, +8, $\cdots$). After the three loops, the proposed method have dealt with all pixels in stego Y, and finally output an enhanced stego image Z. [1]

[1]Codes available at https://github.com/bolin-chen/Stego-Post-processing

Please note that there are two important components in the proposed method, that is, the filter for obtaining image residual and the criterion for comparing image residual differences. In Section 4, we will provide some experimental analysis for the two components.

4 EXPERIMENTS

In this section, we conduct experiments on 10,000 gray-scale images of size 512×512 from BOSSBase-v1.01 [1]. We randomly divide them into two non-overlapping and equal parts, one for training and the other for testing. Like most related literatures, we use the optimal simulator for data embedding. Two typical steganalytic feature sets i.e. SRM [5] and its selection-channel-aware version maxSRMd2 [3] with the ensemble classifier [8] are used separately for security evaluation. To achieve convincing results, we randomly split the training set and test set 3 times and report the average results in the following experiments.

4.1 Filter and Criterion Selection

In this experiment, we compare four filters and two criterions on stego images obtained by HILL. As shown in Fig. 3, four high-pass filters commonly used in steganalysis are considered. The two criterions are formulated as follows:

$$\text{Element-wise: } (D_{TX} \neq D_{ZX})\ \ \&\ \ (D_{TX} \leq D_{ZX}) \tag{4}$$

$$\text{Sum: } \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} D_{TX}(i,j) < \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} D_{ZX}(i,j) \tag{5}$$

where D_T and D_Z are non-negative matrices of size $n_1 \times n_2$, the symbols $\leq$ in (4) is an element-wise comparison, $D_T(i,j)$ in (5) denotes the ij^{th} element of D_T. If formula (4) or (5) is satisfied, we believe that the modified stego T becomes much closer to the original X compared to Z. Thus, the candidate stego Z should be updated as T in this case.

The results of the proposed method with different filters and criterions are shown in Table 1. From Table 1, we observe that in most cases, our method can improve the security of HILL especially for large payloads. It is also observed that the Element-wise criterion is usually better than the Sum criterion. Among all the combinations of filter and criterion, the combination {Element-wise

Table 2: Detection errors (%) for different steganography methods. We name the enhanced version of some steganography such as "A" with the proposed Stego Post-Processing as "A-SPP" for short. We underline those results which become poorer after using our method.

Steganography	SRM					maxSRMd2				
	0.1	0.2	0.3	0.4	0.5	0.1	0.2	0.3	0.4	0.5
WOW	40.19	31.71	25.75	20.72	16.94	29.94	23.39	19.05	15.70	13.06
WOW-SPP	**40.39**	**32.00**	**26.41**	**21.73**	**17.88**	**30.09**	**23.80**	**19.45**	**16.30**	**13.95**
S-UNIWARD	40.35	31.91	25.43	20.54	16.35	36.23	29.10	23.65	19.02	15.67
S-UNIWARD-SPP	**40.25**	**32.03**	**25.88**	**21.07**	**16.93**	**36.41**	**29.40**	**23.99**	**19.72**	**16.37**
MiPOD	41.38	34.28	28.73	23.94	19.77	39.58	32.46	27.14	22.46	18.94
MiPOD-SPP	**41.50**	**34.78**	**29.49**	**25.44**	**21.48**	**39.92**	**33.37**	**28.24**	**24.13**	**20.96**
HILL	43.37	35.83	29.82	24.59	20.30	37.49	30.75	26.04	21.70	18.10
HILL-SPP	**43.43**	**36.43**	**30.80**	**26.32**	**22.12**	**38.37**	**31.92**	**27.59**	**23.71**	**20.22**
CMD-HILL	45.18	39.78	34.27	29.88	25.85	40.28	34.54	30.24	26.74	23.45
CMD-HILL-SPP	**45.16**	**39.81**	**34.36**	**29.96**	**26.21**	**40.13**	**34.66**	**30.39**	**27.02**	**23.84**

$$A=\begin{bmatrix}-1&+2&-2&+2&-1\\+2&-6&+8&-6&+2\\-2&+8&-12&+8&-2\\+2&-6&+8&-6&+2\\-1&+2&-2&+2&-1\end{bmatrix}\quad B=\begin{bmatrix}-1&+2&-1\\+2&-4&+2\\-1&+2&-1\end{bmatrix}$$

$$C=\begin{bmatrix}+1&+1&+1\\+1&-8&+1\\+1&+1&+1\end{bmatrix}\quad D=\begin{bmatrix}0&+1&0\\+1&-4&+1\\0&+1&0\end{bmatrix}$$

Figure 3: Four filters tested in the experiment

Table 3: The post-modification rates (‰) with the proposed method. The asterisk (*) here means the value is less than 0.01 ‰.

Criterion	Filter	0.1	0.2	0.3	0.4	0.5
Element-wise	A	0.71	1.92	3.45	5.26	7.33
	B	0.95	2.82	5.45	8.82	12.93
	C	*	*	*	*	*
	D	0.05	0.17	0.39	0.73	1.23
Sum	A	11.57	29.93	52.27	78.00	106.61
	B	6.33	17.97	33.50	52.65	75.33
	C	0.27	0.98	2.21	4.14	6.94
	D	1.96	6.16	12.40	20.86	31.66

+ Filter-A} always performs the best in this experiment. For instance, the proposed method can increase the detection error by 1.82% under SRM and 2.12% under maxSRMd2 for payload 0.5 bpp, which is a significant improvement in steganalysis for HILL. However, it is interesting that the combination {Sum + Filter-A} performs the worst. It decreases the detection errors for all payloads under two feature sets. Therefore, the combination of filter and criterion is the key issue in the proposed method, which will greatly influence the security performance.

Table 3 gives the post-modification rates with our method. From Table 3, we observe that the post-modification rate is rather low in most cases, and it usually increases with increasing payload for a given combination of filter and criterion. Taking the combination {Element-wise + Filter-A} for example, all modification rates are less than 7.5 ‰ even for payload as large as 0.5 bpp, which means that on average less than 197 pixels have been changed for an image of size 512×512. For a given filter, using Element-wise criterion always has relative lower post-modification rates than using Sum criterion, since formula (4) imposes stronger constraints for updating the target pixel compared to formula (5).

Based on above experiments, the combination {Element-wise + Filter-A} usually have the best security performance for HILL while just changing a comparatively small amount of pixels. For simplicity, we also use this combination in the following experiments although it may not be the best one for other steganography methods.

4.2 Security Evaluation for Different Steganography Methods

In this experiment, five existing steganography methods including WOW, S-UNIWARD, MiPOD, HILL and CMD-HILL are considered. The results are shown in Table 2. From Table 2, we observe that the proposed method can also improve security in most cases, and the improvement usually increases with increasing payload. We also observe that MiPOD and HILL benefit more from the proposed method. For 0.5 payload, we can achieve around 2% improvements both for SRM and maxSRMd2. The improvements for the methods WOW, S-UNIWARD and CMD-HILL are smaller. In this case, we may further improve the security via designing new filters and/or criterions and searching the optimal combination as in Section 4.1.

Table 4 gives the corresponding post-modification rates. From Table 4, we can observe that the post-modification rates increase

Table 4: Post-modification rates (‰) for different steganography methods and payloads

Steganography	0.1	0.2	0.3	0.4	0.5
WOW	0.90	2.30	3.95	5.83	7.91
S-UNIWARD	0.39	1.19	2.30	3.72	5.42
MiPOD	0.32	1.37	2.97	4.97	7.27
HILL	0.71	1.92	3.45	5.26	7.33
CMD-HILL	0.17	0.51	0.98	1.61	2.41

with increasing payload. Taking WOW for an example, its modification rate is 0.90‰ for payload 0.1 bpp while becomes 7.91‰ for payload 0.5 bpp. Overall, the maximum modification rate of all steganography methods for different payloads does not exceed 8‰. It is interesting that we can achieve better security via modifying such a small quantity of pixels in a stego.

4.3 Analysis on Post-Modifications

The experimental results in Section 4.1 and 4.2 show that post-modifications on stego would affect the steganography security. In this section, we provide some supplementary results to analyze the post-modifications for HILL.

4.3.1 About Number of Post-modifications. Fig. 4 shows the violin plot of the numbers of post-modifications using the proposed method for payloads ranging form 0.1 bpp to 0.5 bpp. From Fig. 4, we observe that the median number of post-modifications will increase with increasing payload, which means that the more steganography modifications by previous data embedding, the more post-modifications after using the proposed method, and the greater improvements we made as shown in Table 2. In addition, we observe that the interquartile range will enlarge with increasing payload.

Fig.5 gives three typical image examples and the corresponding steganograpghy modifications using HILL for payload 0.1 bpp. From Fig.5, we observe that the modification distributions of the three images are quite different. For the first image, the modifications seem uniformly dispersed throughout the whole image; while for the third one, they are highly concentrated on a small part of image. After performing our method, the numbers of post-modifications for the three images are 0, 17 and 70 separately, which seems the more concentrated of the modified pixels in previous steganography embedding, the more post-modified pixels would be modified with our method. Similar results can also be obtained for other steganography methods.

4.3.2 About Amplitudes of Post-modifications. As described in Section 2.2, adding a multiple of 4 to any pixel of the resulting stego obtained by ternary STCs embedding will not confuse the message extraction at all. Based on our experiments, we found that most amplitudes of post-modifications are 4, only a small part is 8. For example, only 83 images out of the 10,000 images in the BOSSBase contain post-modifications with a amplitude of 8 for payload 0.5 bpp, while only 2 images for payload 0.1 bpp. No images contain post-modifications with amplitudes larger than 8 for all payloads. It means that proposed method will not introduce visual artifacts due to large amplitude. In addition, we also found that the directions (i.e. + or −) of post-modifications are roughly the same for all payloads.

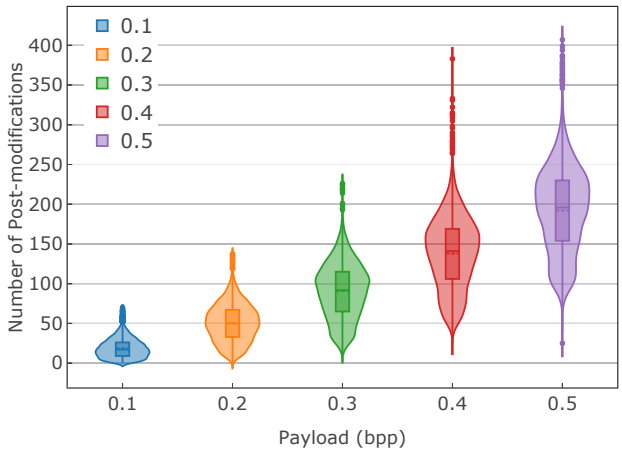

Figure 4: The violin plot of the numbers of post-modifications with our method

5 CONCLUSIONS

In this paper, we first propose a stego post-processing to enhance steganography security by reducing image residual difference between cover and stego. In our experiments, we test the proposed method with four filters and two criterions, and evaluate it on five existing steganography methods in the spatial domain. Experimental results show the proposed method is very promising to improve the security of existing stegangoraphy. This is our initial attempt to enhance steganography via stego post-processing. In future, several important issues in our method are worthing further studying, such as designing a content adaptive filter for calculating image residual, proposing a more reasonable criterion and/or combining with adversarial examples for pixel modification. In addition, it is expected that the proposed method can be extended to JPEG steganography.

ACKNOWLEDGMENTS

This work is supported in part by the NSFC (61672551), the Special Research Plan of Guangdong Province under Grant 2015TQ01X365, and the Guangzhou Science and Technology Plan Project under Grant 201707010167.

REFERENCES

[1] Patrick Bas, Tomáš Filler, and Tomáš Pevnỳ. 2011. "Break Our Steganographic System": The ins and outs of organizing BOSS. In *Springer International Workshop on Information Hiding*. 59–70.
[2] Tomáš Denemark and Jessica Fridrich. 2015. Improving steganographic security by synchronizing the selection channel. In *ACM Workshop on Information Hiding and Multimedia Security*. 5–14.
[3] Tomáš Denemark, Vahid Sedighi, Vojtěch Holub, Rémi Cogranne, and Jessica Fridrich. 2014. Selection-channel-aware rich model for steganalysis of digital images. In *IEEE International Workshop on Information Forensics and Security*. 48–53.
[4] Tomáš Filler, Jan Judas, and Jessica Fridrich. 2011. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security* 6, 3 (2011), 920–935.

Figure 5: Three cover examples (i.e. (a)No.1063, (b)No.7083, (c)No.7680 in BOSSBase) and the corresponding steganograpghy modifications (i.e. (d), (e), (f)) using HILL for payload 0.1 bpp

[5] Jessica Fridrich and Jan Kodovsky. 2012. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security* 7, 3 (2012), 868–882.

[6] Vojtěch Holub and Jessica Fridrich. 2012. Designing steganographic distortion using directional filters. In *IEEE International Workshop on Information Forensics and Security*. 234–239.

[7] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. 2014. Universal distortion function for steganography in an arbitrary domain. *Springer EURASIP Journal on Information Security* 2014, 1 (2014), 1.

[8] Jan Kodovsky, Jessica Fridrich, and Vojtěch Holub. 2012. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security* 7, 2 (2012), 432–444.

[9] Bin Li, Ming Wang, Jiwu Huang, and Xiaolong Li. 2014. A new cost function for spatial image steganography. In *IEEE International Conference on Image Processing*. 4206–4210.

[10] Bin Li, Ming Wang, Xiaolong Li, Shunquan Tan, and Jiwu Huang. 2015. A strategy of clustering modification directions in spatial image steganography. *IEEE Transactions on Information Forensics and Security* 10, 9 (2015), 1905–1917.

[11] Tomáš Pevny, Patrick Bas, and Jessica Fridrich. 2010. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on information Forensics and Security* 5, 2 (2010), 215–224.

[12] Tomáš Pevnỳ, Tomáš Filler, and Patrick Bas. 2010. Using high-dimensional image models to perform highly undetectable steganography. In *Springer International Workshop on Information Hiding*. 161–177.

[13] Vahid Sedighi, Rémi Cogranne, and Jessica Fridrich. 2016. Content-adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security* 11, 2 (2016), 221–234.

[14] Weiming Zhang, Zhuo Zhang, Lili Zhang, Hanyi Li, and Nenghai Yu. 2017. Decomposing joint distortion for adaptive steganography. *IEEE Transactions on Circuits and Systems for Video Technology* 27, 10 (2017), 2274–2280.