IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 9, 2019, accepted May 27, 2019, date of publication May 31, 2019, date of current version June 19, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920304

Fuzzy Localization of Steganographic Flipped Bits via Modification Map

QUANQI LIU[1], TONG QIAO[1,2], MING XU[1], AND NING ZHENG[1]
[1]School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Zhengzhou Science and Technology Institute, Zhengzhou 450001, China

Corresponding author: Ming Xu (mxu@hdu.edu.cn)

This work was supported in part by the National Key Research and Development Plan of China through the Cyberspace Security Major Program under Grant 2016YFB0800201, in part by the Natural Science Foundation of China under Grant 61572165, Grant 61702150, and Grant 61803135, in part by the Public Research Project of Zhejiang Province under Grant LGG19F020015, in part by the Natural Science Foundation of China through the State Key Program of Zhejiang Province under Grant LZ15F020003, and in part by the Key Research and Development Plan Project of Zhejiang Province under Grant 2017C01065.

ABSTRACT Adaptive steganography has become unprecedentedly prevalent compared with non-adaptive ones due to its remarkable performance when resisting modern steganalysis. It prefers hiding bits into pixels from texture regions as such modification is considerably difficult to detect. Current steganalysis capable of locating steganographic payload has only been investigated in the non-adaptive domain, while the works of locating hidden bits modified by adaptive steganographic algorithms have not been studied yet. In this paper, we propose a novel algorithm to locate flipped pixels modified by adaptive steganography in the spatial domain. By re-embedding randomly generated messages upon one single image, we observe that adaptive steganographic methods are prone to modify pixels in the same region, namely texture region. Such property straightforward inspires us to re-embed a random message at the same relative payload into the stego image to obtain the modification map. Then, we extend the modification map with a given margin to locate the modified pixels. The extensive experiments have verified the effectiveness of our designed algorithm in locating flipped pixels modified by the adaptive steganography in the spatial domain.

INDEX TERMS Steganalysis, flipped bits localization, re-embedding random bits, modification map.

I. INTRODUCTION

Steganography is the science and art of concealing secret information. In the field of image steganography, the secret message is carried by an empirical cover under the supervision of the warden, and extracted by the recipient so as to achieve covert communication. A common and practical way in image steganography is to modify cover pixels slightly by ± 1 on the consideration of guaranteeing its undetectability. According to the embedding strategy, steganographic methods nowadays can be divided into two categories: non-adaptive and adaptive steganography.

A. STATE OF THE ART

LSB (Least Significant Bit) replacement is one of the classical non-adaptive steganographic methods. It randomly spreads the modification changes to the whole cover image. To improve the undetectability, LSB matching is then proposed to avoid the asymmetry artifacts by randomly flipping LSBs.

In modern steganography, however, it has been validated that adaptive steganographic methods, mainly relying on texture regions, achieve the optimal undetectability. One of the most successful adaptive models rather treats the message embedding as a source coding problem with a fidelity constraint [1], instead of taking the cover source distribution into account. In this scenario, the design of the distortion function becomes essential especially for the sender who embeds secret message into the cover image, relying on the principle of minimizing the distortion caused by embedding.

In spatial domain, HUGO (Highly Undetectable steGO) [2] allocated high cost to those pixels which caused the feature vector more discriminative after embedding. In such manner, the modification probably happens in texture regions or along edges. HUGO BD was an improved version of HUGO that was implemented using the Gibbs construction with bound

The associate editor coordinating the review of this manuscript and approving it for publication was Zhitao Guan.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

distortion [3]. WOW (Wavelet Obtained Weights) [4] used a bank of high-pass directional filters to avoid embedding in predictable regions such as clean edges, leading to better performance than HUGO when resisting modern steganalysis with SRM (Spatial Rich Models) [5]. UNIWARD (UNIversal WAvelet Relative Distortion) [1] evolved from WOW, and could be applied both in spatial domain (S-UNIWARD) and transformed domain (J-UNIWARD). HILL (HIgh-pass, Low-pass, and Low-pass) [6] averaged pixel values to re-assign the low cost value of pixels (assigned with high cost by prior distortion function) in texture areas, resulting in better performance than that of other aforementioned algorithms. Apart from J-UNIWARD, in JPEG domain, a new steganographic framework with using the constructed reference image and minimizing the feature distortion was proposed for JPEG images [7]. Besides, by considering the robust performance of steganography suffering compression or re-scaling attack, robust image steganography (see [8], [9] for instance) is an emerging research field concerning on data security [10], [11] that draws much attention.

Steganalysis aims to (*i*) detect the existence of hidden information (see [12]–[14] for instance); (*ii*) extract the secret information. Current arts (see [5], [15], [16]) classify between cover and stego images via rich models using ensemble learning. Based on decision rough set α-positive region reduction, [17] selected rich model features to reduce the steganalysis feature dimension and improved its efficiency. Besides, the algorithm proposed in [18] attempted to search stego-key, and extract secret messages given that the key space was relevantly small. To deal with the steganographer detection problem over large-scale social media networks, [19] proposed a method with utilizing high-order joint features and clustering ensembles. Locating hidden bits lies above those objectives. Provided that the payload location is correctly predicted, extracting hidden data would be feasible. In [20] and [21], payload of non-adaptive steganography was successfully located. However, to our knowledge, few studies focus on locating adaptive steganographic payload.

B. CONTRIBUTIONS OF THE PAPER

In this paper, we novelly propose fuzzy localization of flipped bits via the designed modification map. Those bits are within the suspicious regions modified by adaptive steganography. The main contributions of this paper are as follows:

- We propose a new framework to locate the steganographic flipped bits in spatial domain via a modification map, which can be applied in modern adaptive steganography.
- In both two scenarios that the steganographic payload is known or unknown, our proposed locating algorithm performs very well.
- We employ F_1 measure to strike the balance between recall rate and precision rate of fuzzy localization which is capable of evaluating the security of stegonagraphy, involving both undetectability and localization resistance.
- Numerical experiments demonstrate the sharpness of the empirically established results and the good performance of our proposed locating algorithm. Furthermore, cross validation experiments are conducted to verify the effectiveness of our locating algorithm.

C. ORGANIZATIONS OF THE PAPER

The rest of our paper is organized as follows. Relevant existing works are introduced in Section II. In Section III, we expose the locations of suspicious regions modified by adaptive steganography. Our algorithm, that can locate flipped hidden bits, is specified in Section IV. Section V validates our locating algorithm by extensive experiments. This paper is concluded in Section VI.

II. RELATED WORK

To our knowledge, morden steganalysis incorporating with the knowledge of pixel embedding probability achieves high detection performance (see [16], [22], [23]). In [22], the embedding costs of pixels were sorted in ascending order, resulting in that feature extraction only happened in texture regions (instead of the whole image). In such manner, it achieved better detection performance for WOW steganography. Meanwhile, authors of [23] proposed a variant of the SRM that utilized the pixel embedding change probability. This approach, known as Selection Channel Aware (SCA) steganalysis, increased the detection accuracy in comparison with the original SRM. Next, based on the embedding probability, the universal and unified adaptive steganalytic framework was proposed in [16].

The core principle of adaptive steganalytic schemes aforementioned is to take the embedding cost or embedding probability into consideration in the feature extraction step. If a pixel is flipped by adaptive steganography, it owns low embedding cost and high embedding probability, and vice versa. With such a property, our proposed steganographic localization algorithm could be incorporated with modern steganalysis because those to-be-localized pixels probably exist in the texture region that have high embedding probability, and hence further increases detection performance. Moreover, in an ideal scenario, the acquired location information probably helps us extract hidden message. Last but not least, we could also modify only a small portion of the pixels within the located regions slightly by ± 1 while visually retains the image quality, in order to interfere/misguide the secret communication between two adversarial parties.

In previous steganalysis aiming at detecting non-adaptive steganography, authors of [20] located steganographic payload embedded by LSB replacement via WS (Weighted Stego-image) residuals. Relying on WAM (Wavelet Absolute Moments) features [24], authors of [21] located the payload embedded by LSB matching. But, current algorithms that perform very well on locating non-adaptive steganographic payload are invalid in adaptive domain. Because a large number of stego images modified in the same pixels are required, which is an unrealistic assumption in the practical

(a) (b) (c)

FIGURE 1. Illustration of a cover image from BOSSbase, our embedding modification map, embedding probability map of [16]. Note that two maps are acquired using S-UNIWARD steganography at 0.30 bpp. (a) Cover image. (b) Embedding modification map. (c) Embedding probability map.

detection. To our knowledge, few studies focus on locating adaptive steganographic payload. In this paper, we propose a novel algorithm to locate flipped pixels modified by adaptive steganography. Note that our locating algorithm works very well using only one single stego image.

Through multi-embedding operation, the prior work [16] has verified the effectiveness of assigning the embedding probability to each pixel when steganalyzing. Different from that, we locate the flipped pixels of adaptive steganography by once re-embedding. For instance, secret messages are embedded by S-UNIWARD steganography at 0.30 bpp. Fig. 1 compares the experimental results. We can observe that nearly all the pixels modified by re-embedding (pixels within the bright regions in Fig. 1(b)) have relatively large embedding probabilities in Fig. 1(c), meaning that they are more likely to be embedded. Therefore, in our designed algorithm, it is proposed to locate the modified pixels of adaptive steganography simply by once re-embedding rather than multi-embedding. Besides, compared to once re-embedding, multi-embedding is more time-consuming (*e.g.*, time cost of multi-embedding to obtain the embedding probability as Fig. 1(c) is 40 times of obtaining Fig. 1(b)). Previous works successfully predict embedding probability by multi-embedding and hence achieve high steganalysis performance. In Section V, solid experiments verify that our proposed locating algorithm also works very well.

III. LOCATION OF SUSPICIOUS REGION

In this section, we expose the locations of suspicious region modified by adaptive steganography perceptually and empirically. By re-embedding the same, flipped, and random bits, we specifically analyze the modification of an inquiry image caused by adaptive steganography, resulting into the design of our locating algorithm.

A. RE-EMBEDDING SAME BITS

Adaptive steganography inherently prefers slightly modifying pixels within texture regions, meaning that the content distribution of the cover image is preserved. While we investigate if the modification regions of two embeddings upon one single image are partly overlapped since the cost matrix of the cover remains almost unchanged. First, we generate a random bit stream at relative payload 0.1 bpp, which is embedded into a grey-level cover image $\mathbf{Z} = \{z_{i,j}\}, i \in \{1, \ldots, I\}, j \in \{1, \ldots, J\}$. Subsequently, it is proposed to use the same message to generate a new stego image $\mathbf{S}' = \{s'_{i,j}\}$ by modifying the original stego image $\mathbf{S} = \{s_{i,j}\}$[1] acquired from $\mathbf{Z}$. Note that we use one of often-adopted distortion functions, S-UNIWARD, and both embedding operations are with the help of STCs (Syndrome-Trellis Codes). Immediately, let us define the modification map by:

$$\mathbf{M}_{\text{map}}(i,j) = \begin{cases} 255, & \text{if the pixel at } (i,j) \text{ is modified} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where the modification map $\mathbf{M}_{\text{map}}$ has the same dimension as the cover or stego image. For clarity, Fig. 2 illustrates a 512×512 8-bit gray-scale cover image, and two modification maps from its corresponding stego images. $\mathbf{M}_{\text{map}}1$ pointing out the flipped pixels caused by embedding, is obtained from both $\mathbf{Z}$ and $\mathbf{S}$ while $\mathbf{M}_{\text{map}}$ 2 is acquired from both $\mathbf{S}$ and $\mathbf{S}'$. We observe that two embedding operations both prefer selecting pixels almost in the same region, referring to the texture region. Besides, few pixels are both set as 255 in the same position of two modification maps, meaning that those pixels have been flipped twice.

We assume that adaptive steganographic methods might not modify the same pixel at twice while probably its neighboring pixels. In this case, it is proposed to define our proposed steganalysis algorithm as fuzzy localization.

[1] In this context, regardless of hidden bits (the same, flipped or random ones), the first stego image generated from the cover $\mathbf{Z}$ is denoted as $\mathbf{S}$ while the second stego image from $\mathbf{S}$ denoted as $\mathbf{S}'$.

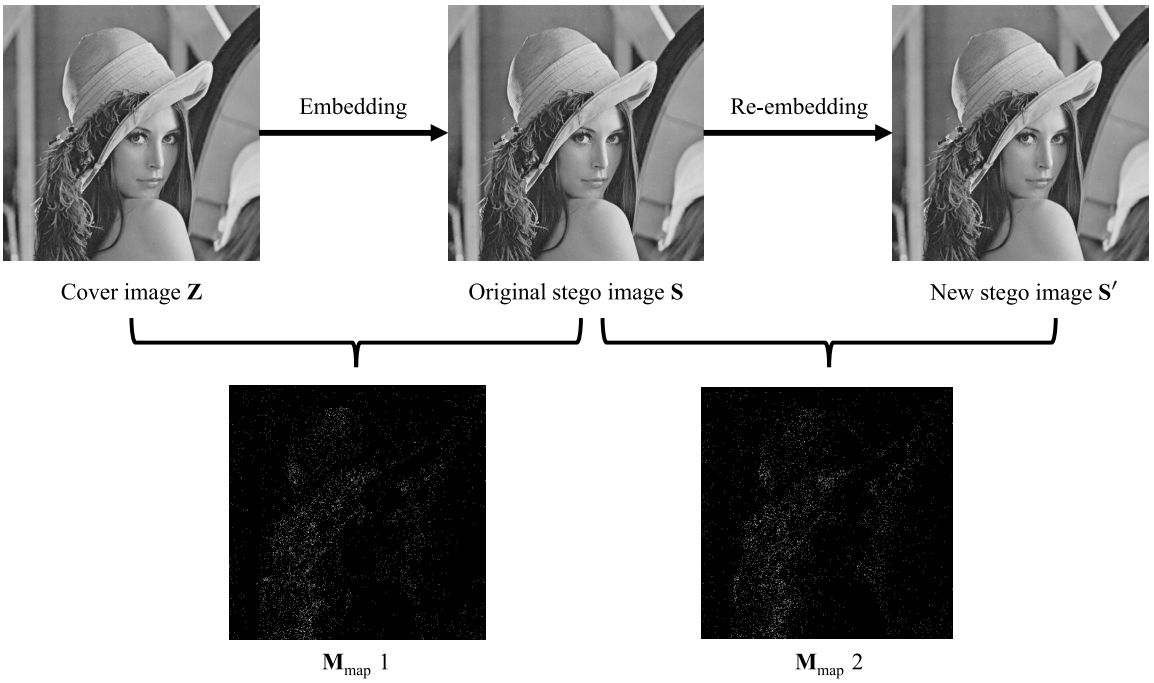

FIGURE 2. Illustration of a cover image and two modification maps from its corresponding stego images using S-UNIWARD at 0.1 bpp.

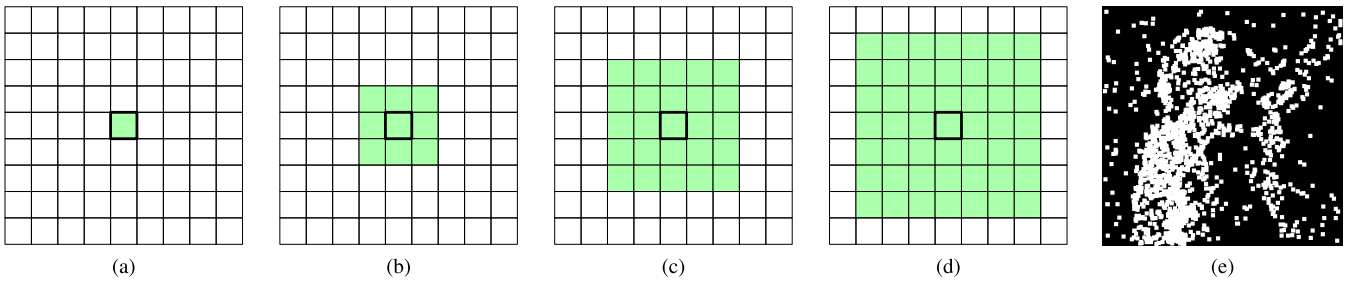

FIGURE 3. Illustration of adjacent regions of a pixel in several different margin values, and the modification map $\mathbf{M}^{\mathrm{e}}_{\mathbf{map}}$ with margin value $K = 3$. (a) $K = 0$. (b) $K = 1$. (c) $K = 2$. (d) $K = 3$. (e) $\mathbf{M}^{\mathrm{e}}_{\mathbf{map}}$.

Inspired by the results of Fig. 2, we extend the $\mathbf{M}_{\mathrm{map}}$ by considering each pixel's neighbors, which can be formulated by:

$$\mathbf{M}^{\mathrm{e}}_{\mathrm{map}}(i+q, j+q) = \begin{cases} 255, & \text{if the pixel at } (i,j) \text{ is modified} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $q \in [-K, K]$ represents an integer set controlled by the margin value K. Next, the adjacent regions of a pixel in several different margin values are illustrated in Fig. 3(a)∼(d). Fig. 3(e) illustrates $\mathbf{M}^{e}_{map}$ with margin value $K = 3$, where the bright regions (the pixels are set as 255) definitely cover a large portion of pixels modified by the first embedding while we cannot guarantee that the bright regions perfectly exclude unmodified pixels. When the margin value equals to K, the dimension of the corresponding adjacent region is formulated as $(2K+1) \times (2K+1)$. Obviously, in the case of $K = 0$, the modification map $\mathbf{M}^{\mathrm{e}}_{\mathrm{map}}$ is reduced back to $\mathbf{M}_{\mathrm{map}}$. Furthermore, one can define p as:

$$p = \frac{m}{n} \tag{3}$$

where n is the number of pixels that are modified when first embedding (those pixels are set as 255 in $\mathbf{M}_{\mathrm{map}}$ 1), and m denotes the number of pixels that are both modified when embedding and re-embedding (those pixels are set as 255 both in $\mathbf{M}_{\mathrm{map}}$ 1 and in $\mathbf{M}_{\mathrm{map}}$ 2). p denotes the proportion of the number of pixels modified by the first embedding and correctly predicted by re-embedding.

We conduct our experiments over 10000 8-bit images from the BOSSbase 1.01 [25]. The experimental results in different margin values are listed in Table 1. One can observe that at a fixed payload, the proportion p can be increased as the margin value K becomes larger. Because the larger adjacent regions can cover more pixels used for information hiding. Besides, in a fixed margin value

TABLE 1. Statistics on p within the margin value K when re-embedding same bits.

Payload α	K										
	0	1	2	3	4	5	6	7	8	9	10
0.05 bpp	0.0645	0.3441	0.5587	0.6884	0.7701	0.8239	0.8609	0.8872	0.9063	0.9205	0.9310
0.10 bpp	0.0863	0.4340	0.6655	0.7906	0.8621	0.9050	0.9318	0.9488	0.9597	0.9669	0.9717
0.20 bpp	0.1170	0.5447	0.7799	0.8875	0.9391	0.9645	0.9771	0.9836	0.9870	0.9890	0.9902
0.30 bpp	0.1427	0.6243	0.8498	0.9364	0.9701	0.9835	0.9890	0.9916	0.9929	0.9936	0.9941
0.40 bpp	0.1659	0.6877	0.8972	0.9634	0.9841	0.9910	0.9936	0.9948	0.9954	0.9958	0.9960
0.50 bpp	0.1890	0.7431	0.9314	0.9791	0.9912	0.9947	0.9961	0.9967	0.9970	0.9972	0.9973

TABLE 2. Statistics on p within the margin value K when re-embedding flipped bits.

Payload α	K										
	0	1	2	3	4	5	6	7	8	9	10
0.05 bpp	0.0667	0.3557	0.5775	0.7116	0.7960	0.8516	0.8897	0.9168	0.9365	0.9510	0.9619
0.10 bpp	0.0879	0.4419	0.6771	0.8040	0.8766	0.9200	0.9471	0.9643	0.9755	0.9827	0.9876
0.20 bpp	0.1180	0.5487	0.7856	0.8937	0.9457	0.9711	0.9839	0.9904	0.9939	0.9959	0.9971
0.30 bpp	0.1433	0.6272	0.8538	0.9409	0.9748	0.9882	0.9938	0.9964	0.9977	0.9984	0.9989
0.40 bpp	0.1666	0.6903	0.9003	0.9668	0.9876	0.9945	0.9971	0.9983	0.9989	0.9993	0.9995
0.50 bpp	0.1895	0.7450	0.9337	0.9816	0.9937	0.9972	0.9986	0.9992	0.9995	0.9997	0.9998

TABLE 3. Statistics on p within the margin value K when re-embedding random bits.

Payload α	K										
	0	1	2	3	4	5	6	7	8	9	10
0.05 bpp	0.0666	0.3555	0.5772	0.7113	0.7958	0.8514	0.8895	0.9166	0.9363	0.9510	0.9619
0.10 bpp	0.0879	0.4421	0.6773	0.8041	0.8766	0.9201	0.9471	0.9644	0.9755	0.9827	0.9876
0.20 bpp	0.1179	0.5487	0.7855	0.8937	0.9457	0.9712	0.9839	0.9904	0.9939	0.9959	0.9971
0.30 bpp	0.1434	0.6273	0.8539	0.9409	0.9748	0.9882	0.9938	0.9964	0.9977	0.9984	0.9989
0.40 bpp	0.1665	0.6901	0.9003	0.9668	0.9876	0.9945	0.9971	0.9983	0.9989	0.9993	0.9995
0.50 bpp	0.1895	0.7450	0.9338	0.9816	0.9937	0.9972	0.9986	0.9992	0.9995	0.9997	0.9998

K, the more the bits embedded into, the larger proportion the modified pixels can be located.

As Table 1 reports, p nearly remains stable with the large K and payload. We assume that the cost value of the pixels modified by the first embedding are slightly changed (see [16] for details). Because those pixels are merely modified by ± 1. When we re-embed the same bits into the stego image $\mathbf{S}$, some pixels carrying the payload might not be flipped again.

B. RE-EMBEDDING FLIPPED BITS

To locate the pixels modified in the cover image $\mathbf{Z}$, we intend to flip the original bits (for the first embedding) when re-embedding. The results are illustrated in Table 2.

As Table 2 illustrates, one can observe that with increasing K, the proportion p is gradually enhanced at the given payload. Also, for a given K, the proportion p increases when increasing payload. For instance, when the payload is 0.05 bpp and K equals to 0, few modified pixels (6.67%) are located by re-embedding. However, when the payload increases to 0.50 bpp and K equals to 10, almost all modified pixels (99.98%) are correctly located.

Compared to the results in Table 1, one can locate more modified pixels. Because small portions of pixels modified in the first embedding cannot flip when re-embedding the same bits. By artificially flipping the bits, we locate those pixels carrying the payload without being flipped for the first embedding.

C. RE-EMBEDDING RANDOM BITS

However, due to the unpredictability of the message embedded by the sender in a covert communication, we can neither predict the hidden bits when steganalyzing nor the flipped ones. Thus, for a practical case, let us generate a random bit stream instead when re-embedding. The results are shown in Table 3.

From Table 3, it is observed that with increasing K, the proportion p is also gradually enhanced. Although we replace the flipped bits with the random ones, the locating performance is marginally degraded, nearly as well as results with the flipped bits. The difference of p is less than 0.03% in each corresponding position from the results between Table 2 and Table 3. Therefore, our assumption is empirically verified, that one can re-embed a random bit stream in place of unknown hidden bits to realize the fuzzy localization.

Despite the fact that adaptive steganography achieves a high level of undetectability when resisting steganalysis, it does have an inherent limitation. That is, adaptive steganography is prone to modify the pixels within texture regions, preserving the content distribution of the cover image. When re-embedding happens, it probably modifies the same regions upon an image. If we take a stego image as cover, and artificially embed the same or even random messages into an image, majority of pixels or their neighbors might be modified twice. Such limitation can be utilized to locate the modification region caused by adaptive steganography.

IV. OUR PROPOSED LOCATING ALGORITHM

In this section, we provide a specific description of our designed algorithm that attempts to locate the region in which the pixels are modified by adaptive steganographic methods in spatial domain. Inspired by the assumption proposed in Section III, a re-embedding operation applied into the stego image $\mathbf{S}$ is introduced.

Our proposed locating algorithm is designed both for the following two scenarios. In the first scenario where the steganographic payload is known, modified pixels are located simply using the acquired payload. More practically, in another scenario where the payload is unknown, we conduct the localization relying on quantitative steganalysis, that is capable of completing payload estimation [26]. In Fig. 7, we have compared the performance from two scenarios, where we can observe that regardless of knowing payload or estimating payload, our proposed locating algorithm always performs very well.

The description of our locating algorithm can be summarized as follows:

- **Step #1: Generating a random bit stream.** A random bit stream $\mathbf{m}$ with the length L is generated. Note that $L = N \times \alpha$ where N denotes the total amount of pixels, and α is the relative payload.
- **Step #2: Calculating a cost matrix.** Relying on an adaptive steganographic algorithm, a bank of designed filters are then utilized to obtain the cost matrix of the stego image. For simplicity and clarity, let us denote the stego image as $\mathbf{S}$, and the cost matrix as ρ.
- **Step #3: Embedding a message using STCs.** Without loss of generality, STCs is used to embed message $\mathbf{m}$ into the stego $\mathbf{S}$ on the principle of minimizing the distortion function based on the cost matrix ρ. In such manner, the stego version of image $\mathbf{S}$ after modification is denoted as $\mathbf{S}'$.
- **Step #4: Obtaining a modification map.** On the purpose of locating modified pixels, the modification map $\mathbf{M}_{\text{map}}$ is obtained as described in Section III.
- **Step #5: Extending the modification map.** To complete fuzzy localization of modified bits, we design the extended modification map $\mathbf{M}^{e}_{map}$ in a given margin value K to locate the modified pixels as more as possible.

Note that the value K would increase if we intend to locate the modified pixels as more as possible. While as the margin value K becomes larger, more and more innocent pixels (without being modified) would be also involved. Thus, it is proposed to design a reasonable metric to acquire the optimal K. Immediately, let us respectively define the precision rate P by:

$$P = \frac{S}{N_1} \tag{4}$$

and recall rate R by:

$$R = \frac{S}{N_2} \tag{5}$$

where S denotes the number of pixels that are modified twice, valuing 255 both in $\mathbf{M}_{\text{map}}$ 1 and $\mathbf{M}^{\text{e}}_{\text{map}}$ 2. N_1 represents the number of pixels that are modified by embedding, valuing 255 in $\mathbf{M}_{\text{map}}$ 1, and N_2 counts the number of pixels that are modified by re-embedding, valuing 255 in $\mathbf{M}^{\text{e}}_{\text{map}}$ 2. Note that if K become large, the recall rate R would increase, leading to that the adjacent regions cover more flipped pixels. Meanwhile, more innocent pixels being contained in the adjacent regions results into the decreased P, and vice versa. To leverage the tradeoff between R and P, let us formulate F_β by:

$$F_\beta = \frac{(1+\beta^2) \times P \times R}{(\beta^2 \times P) + R} \tag{6}$$

where we set β as 1, because we allocate the same weights to P and R as they both are important in our designed method. Thus, F_β reduces to F_1:

$$F_1 = \frac{2 \times P \times R}{P + R} \tag{7}$$

V. EXPERIMENTS

The main contribution of this paper is to realize fuzzy localization of flipped bits by using our proposed modification map. Note that when the hidden bits have been embedded merely via adaptive steganography, our proposed algorithm can perform effectively. In this section, we mainly introduce our experiment setups, and demonstrate the relevant results of the well-performed detector when dealing with the problem of locating flipped bits modified by modern adaptive steganographic methods. Note that we conduct the fuzzy localization experiments under two scenarios, involving known and unknown payload. Besides, it is proposed to determine the

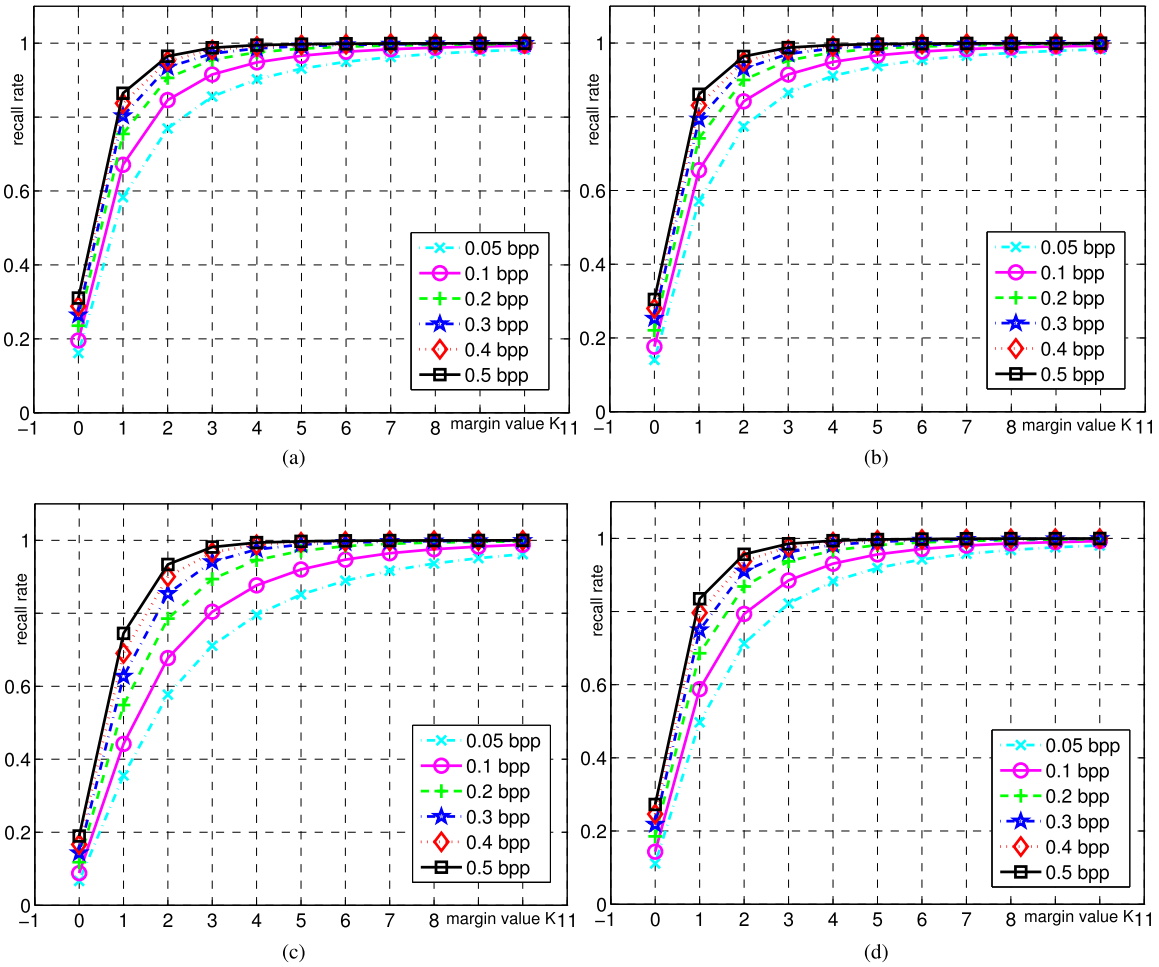

FIGURE 4. Recall rate in different margin values at various payloads. (a) Recall rate of HUGO. (b) Recall rate of WOW. (c) Recall rate of S-UNIWARD. (d) Recall rate of HILL.

parameter K, and implement the cross validation, where four modern steganographic schemes are introduced to investigate the effectiveness of our proposed locating algorithm.

A. EXPERIMENT SETUPS

We conduct our experiments on the benchmark BOSSbase ver.1.01 [25], which contains 10000 8-bit gray-scale images. All images are acquired from eight different digital still cameras in the size of 512 × 512 pixels. Table 4 reports the experimental environment and statistic.

B. FUZZY LOCALIZATION WITH KNOWING PAYLOAD

In our experiments, four modern adaptive steganographic methods: HUGO BD, WOW, S-UNIWARD, and HILL, are investigated. It is proposed to adopt different cost matrices from HUGO BD, WOW, S-UNIWARD, and HILL. In each group, six random bit streams at different payloads from 0.05 bpp to 0.5 bpp are generated, and embedded using STCs. The experimental results are illustrated in Fig. 4.

At the beginning, the recall rate R for four algorithms remarkably increases, and then is slightly improved with increasing K, for all the six payloads from 0.05 bpp to 0.50 bpp. When K equals to 4, all recall rates are larger than 80%. Furthermore, R nearly approaches to 100% when K equals to 9, meaning that almost all modified pixels are successfully located. Dealing with each steganographic algorithm in a fixed K, the larger the payload, the more pixels

TABLE 4. Experimental environment and statistic.

Image source	BOSSbase 1.01 dataset
Image color	Grey-level
Image size	512 × 512
Image format	PGM
Number of original images	10000
Payload	0.05 ∼ 0.5 bpp
Steganographic schemes	HUGO BD, WOW, S-UNIWARD, HILL
Locating method	Our proposed algorithm
CPUs	4 × Intel Xeon E7-4820 2.0GHz CPUs
RAM	16G

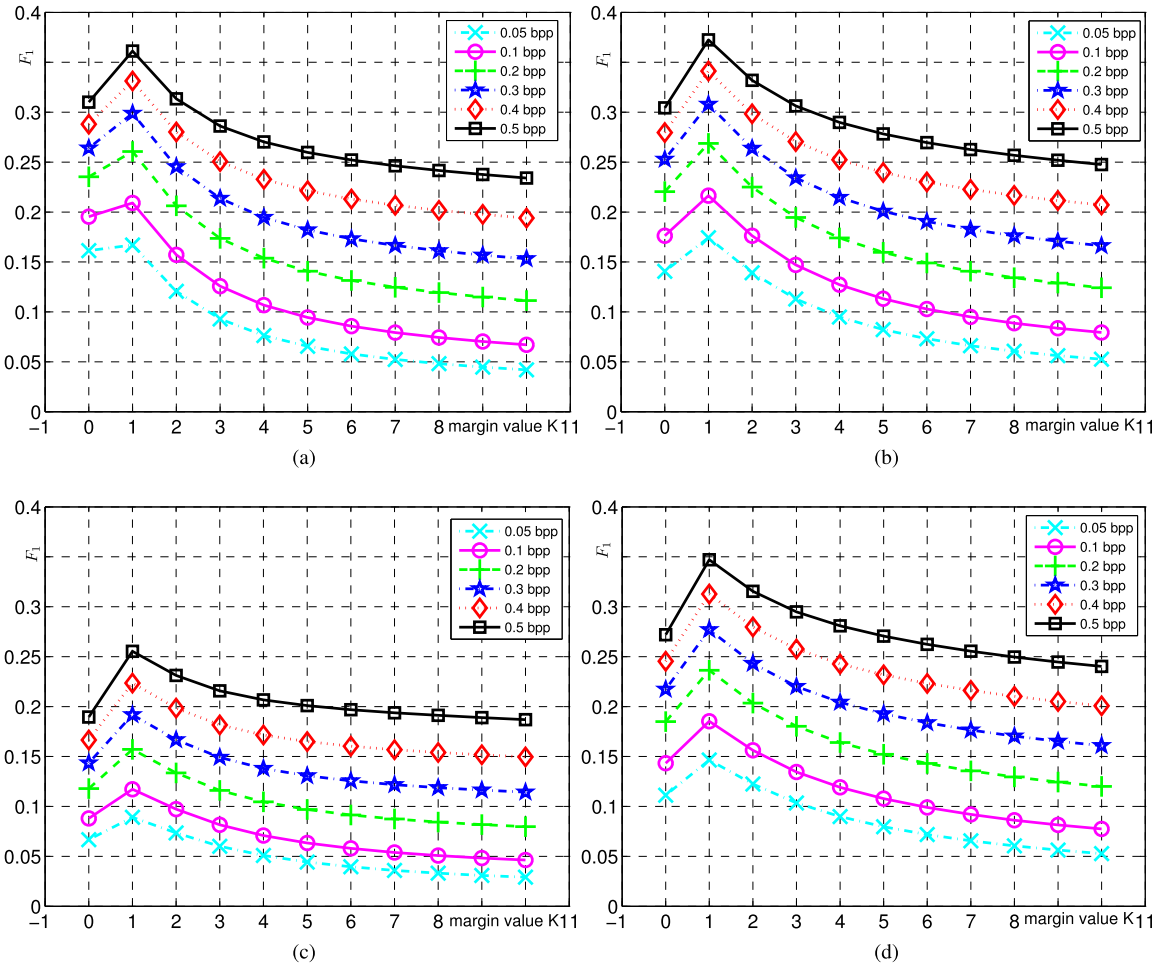

FIGURE 5. F_1 value in different margin values at various payloads. (a) F_1 value of HUGO. (b) F_1 value of WOW. (c) F_1 value of S-UNIWARD. (d) F_1 value of HILL.

modified can be located. Thus, the results empirically verify that our algorithm performs effectively in locating steganographic bits modified by prior arts.

C. DETERMINING PARAMETER K

Since K has a great impact on our algorithm, let us discuss the selection of it. As K becomes larger, more flipped pixels can be located. Meanwhile, more innocent pixels are also incorrectly identified, saying that the precision rate P degrades. To leverage between R and P, it is reasonable that F_1 is introduced. Fig. 5 reports the results of F_1 for four adaptive steganographic methods.

We can observe that as the K becomes larger, the F_1 value increases at first and then gradually decreases. Based on the empirical results, let us set an optimal K as 1 for prior arts at each payload, leading to 3×3 adjacent region. Besides, It can be observed that HILL is more easier to be located than S-UNIWARD. Because HILL adopts two low-pass filters to make the low cost values (corresponding to the texture regions) more clustered, leading to that the modifications caused by HILL are prone to the texture region compared to that of S-UNIWARD. In this case, our texture-region-sensitive algorithm is more easy to localize modification caused by HILL.

To our knowledge, when evaluating the security of steganography, most of prior literature only focus on undetectability, using some metrics such as E_{oob} or P_E. However, few literature focus on the security of localization resistance, namely when a stego image is successfully detected, whether or not the hidden bits of the stego image can be furthermore accurately located. Then let us re-define the security of stegonagraphy, involving both undetectability and localization resistance. Therefore, although HILL is slightly more undetectable than S-UNIWARD, it hardly holds true that HILL performs better than S-UNIWARD since S-UNIWARD is more difficult to locate.

D. FUZZY LOCALIZATION WITHOUT KNOWING PAYLOAD

In subsection V-B, we conduct fuzzy localization of four modern adaptive steganographic schemes under the assumption that the steganographic payload has been acquired. In a more practical case, we intend to locate flipped bits provided that the steganographic payload is unknown. In this subsection, we locate modified pixels of adaptive steganog-

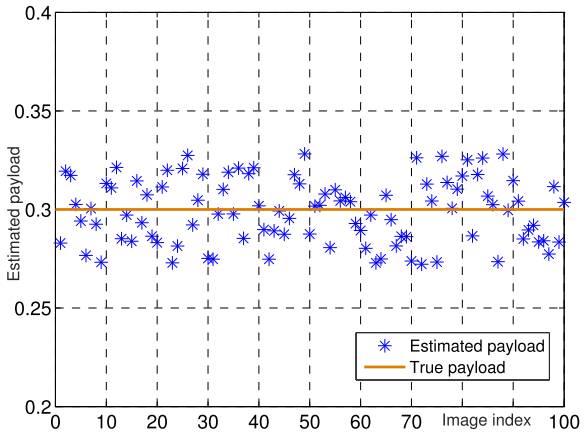

FIGURE 6. Illustration of estimated payloads of 100 images compared to the true values.

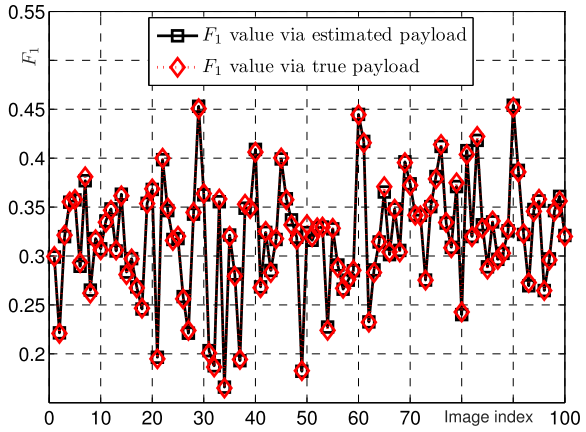

FIGURE 7. Illustration of F_1 values calculated respectively via estimated and true payload.

raphy relying on the estimated payload so as to verify that our locating algorithm still works well with the estimated payload.

In this experiment, we first embed random bits into the cover images at 0.30 bpp according to the rule of HUGO. Then, quantitative steganalysis in [26] is used for providing an estimated payload of the stego images. Fig. 6 reports the estimated payloads of 100 images randomly selected from BOSSbase in comparison with its corresponding true version. We can observe that the subtle differences (less than 0.04 bpp) exist between the estimated payload and the true one. The results of Fig. 6 indirectly verify that locating steganographic bits via the estimated payload is feasible.

Next, let us adopt our proposed algorithm to locate the modified pixels with the estimated payload. For comparison, locating modified pixels by the true payload 0.3 bpp is also conducted. The F_1 values of 100 images are illustrated in Fig. 7. As Fig. 7 reports, each F_1 value via the estimated payload are nearly overlapped with that via the true payload. The difference between two F_1 values attributes to the minor detection error between the estimated payload and the true payload. However, such a minor detection error is acceptable

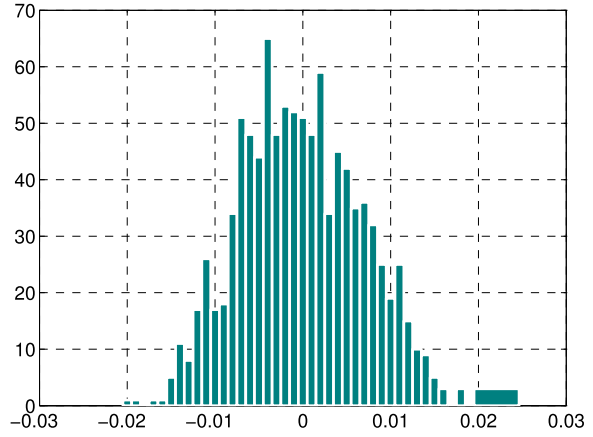

FIGURE 8. Illustration of the difference histogram between the F_1 values via estimated payload and the ones via true payload.

in our proposed locating algorithm. Nevertheless, in the practical scenario that the steganographic payload is unknown, the experimental results directly verify that one can locate steganographic flipped pixels with the payload estimated by quantitative steganalysis.

Furthermore, let us redo the experiment using a large scale set, namely 1000 images randomly selected from BOSSbase. Fig. 8 shows the difference histogram between the F_1 values obtained respectively via estimated payload and true payload. It can be observed that the differences are quite small. All values are within the range from -0.025 to 0.025 and more than 84.2% of the values lay in the range from -0.01 to 0.01. Fig. 8 solidly verifies that regardless of acquiring the steganographic payload, our proposed locating algorithm always works very well.

E. CROSS VALIDATION OF DIFFERENT STEGANOGRAPHIC METHODS

In the aforementioned experiments, our locating algorithm works under two assumptions: (*i*) the steganographic payload is acquired; (*ii*) the steganographic scheme is correctly detected. In subsection V-D, we have verified that the locating algorithm works well with the estimated payload. In this subsection, let us assume the steganographic scheme is unknown. Then we intend to conduct the cross validation experiments over different steganographic algorithms.

For the purpose of realizing cross validation, each of the four steganographic algorithms (*i.e.*, HUGO BD, WOW, S-UNIWARD and HILL) is adopted respectively to hide secret random message into the cover image. Next, those hidden bits are located by using our proposed algorithm, assuming all four steganographic algorithms. We take BOSSbase as our benchmark and embed secret bits into 10000 cover images at 0.30 bpp. The experimental results are reported in Table 5.

We can observe that the underlined F_1 values in boldface along the diagonal direction are larger than others in the table. In each row, assuming the same steganographic method, our proposed algorithm for locating modified pixels

TABLE 5. F_1 value of cross validation by different steganographic schemes.

Steganographic scheme for embedding	Our locating algorithm assuming steganographic scheme			
	HUGO BD	WOW	S-UNIWARD	HILL
HUGO BD	**0.2989**	0.2771	0.2380	0.2462
WOW	0.2716	**0.3076**	0.2532	0.2728
S-UNIWARD	0.2030	**0.2192**	0.1917	0.2002
HILL	0.2362	0.2651	0.2246	**0.2770**

can obtain the largest F_1 value, while by assuming other three steganographic schemes the proposed algorithm has a minor performance decrease. The greatest decrease rate is 20.37%.[2] Because our locating algorithm assuming the same steganographic scheme for embedding is more likely to reproduce the modification of embedding as they share the same embedding distortion function. Furthermore, other three adaptive steganographic schemes can also predict large portion of the modification as they are all prone to embed in texture regions. Experiment results empirically verify that the most accurate way to locate modified pixels is to assume the correct steganographic scheme. Even if the steganographic scheme is unknown, one can also locate flipped pixels using the other three adaptive steganographic methods to obtain the sub-optimal result.

Surprisingly, it's better to assume WOW than S-UNIWARD when locating flipped bitts embedded by S-UNIWARD. This attributes to the overly adaptivity of WOW compared with S-UNIWARD. WOW might concentrate more modification on the regions that are difficult to model, leading to that our proposed locating algorithm with assuming WOW also focuses more on texture regions where the payload of S-UNIWARD is exactly embedded.

VI. CONCLUSION

In this paper, exploiting the modification regions caused by adaptive steganographic method, a steganalytic algorithm is proposed to implement fuzzy localization of modified pixels. Through re-embedding a random message, we design a modification map that is capable of locating hidden bits caused by adaptive steganography in spatial domain. Extensive experiments verifies that when the payload is known or unknown, our locating algorithm performs very well. Furthermore, the experiment of cross validation directly verifies the effectiveness of our algorithm in the practical localization.

Our proposed locating algorithm performs well on locating adaptive steganographic payload while it fails in non-adaptive cases (for instance, LSB replacement and LSB matching). Because non-adaptive steganography spreads the modification to the whole cover image randomly and is not prone to modify the same regions in re-embedding. Besides, it is worth noting our locating algorithm works well on the basic prerequisite that an inquiry image has been classified as a stego one. However, that would not be a limitation since current steganalysis (see [5], [15], [26] for instance) is sufficiently powerful to determine a stego image. In this scenario, we has verified the effectiveness of the proposed algorithm (see details in the experiment of V-D).

In fact, although the fuzzy localization is successfully realized, the stego key is still unavailable, leading to the failure of restoring the secret hidden message. However, it is possible that a latent malicious attacker slightly modifies the stego image by dithering a small portion of localized modified pixels, resulting in that the misguided secret information is transmitted. In our future work, we intend to take the pixel embedding probability into consideration as to narrow down the suspicious regions and improve the detection accuracy. Moreover, it can be promising to straightforward extend our proposed algorithm into JPEG domain to locate the DCT coefficients modified by adaptive steganography.

[2]The rate $(0.2989-0.2380)/0.2989 \approx 20.37\%$, in the case of HUGO BD for embedding and S-UNIWARD for assuming.

ACKNOWLEDGMENT

(Quanqi Liu and Tong Qiao are co-first authors.)

REFERENCES

[1] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.*, vol. 2014, no. 1, pp. 1–13, 2014.
[2] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proc. Int. Workshop Inf. Hiding*. Calgary, AB, Canada: Springer, 2010, pp. 161–177.
[3] T. Filler and J. Fridrich, "Gibbs construction in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 705–720, Dec. 2010.
[4] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2012, pp. 234–239.
[5] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
[6] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 4206–4210.
[7] Z. Wang, Z. Qian, X. Zhang, M. Yang, and D. Ye, "On improving distortion functions for JPEG steganography," *IEEE Access*, vol. 6, pp. 74917–74930, 2018.
[8] Y. Zhang, C. Qin, W. Zhang, F. Liu, and X. Luo, "On the fault-tolerant performance for a class of robust image steganography," *Signal Process.*, vol. 146, pp. 99–111, May 2018.
[9] Y. Zhang, X. Luo, Y. Guo, C. Qin, and F. Liu, "Zernike moment-based spatial image steganography resisting scaling attack and statistic detection," *IEEE Access*, vol. 7, pp. 24282–24289, 2019.

[10] Z. Guan, G. Si, X. Zhang, L. Wu, N. Guizani, X. Du, and Y. Ma, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 82–88, Jul. 2018.
[11] Z. Guan, Y. Zhang, G. Si, Z. Zhou, J. Wu, S. Mumtaz, and J. Rodriguez, "ECOSECURITY: Tackling challenges related to data exchange and security: An edge-computing-enabled secure and efficient data exchange architecture for the energy Internet," *IEEE Consum. Electron. Mag.*, vol. 8, no. 2, pp. 61–65, Mar. 2019.
[12] T. Qiao, C. Zitzmann, F. Retraint, and R. Cogranne, "Statistical detection of jsteg steganography using hypothesis testing theory," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 5517–5521.
[13] T. Qiao, C. Ziitmann, R. Cogranne, and F. Retraint, "Detection of jsteg algorithm using hypothesis testing theory and a statistical model with nuisance parameters," in *Proc. 2nd ACM Workshop Inf. Hiding Multimedia Secur.*, 2014, pp. 3–13.
[14] T. Qiao, F. Retraint, R. Cogranne, and C. Zitzmann, "Steganalysis of JSteg algorithm using hypothesis testing theory," *EURASIP J. Inf. Secur.*, vol. 2015, no. 1, pp. 1–16, 2015.
[15] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 432–444, Apr. 2012.
[16] W. Tang, H. Li, W. Luo, and J. Huang, "Adaptive steganalysis based on embedding probabilities of pixels," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 734–745, Apr. 2016.
[17] Y. Ma, X. Luo, X. Li, Z. Bao, and Y. Zhang, "Selection of rich model steganalysis features based on decision rough set α-positive region reduction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 336–350, Feb. 2019.
[18] J. Fridrich, M. Goljan, and D. Soukal, "Searching for the stego-key," *Proc. SPIE, Electron. Imag., Secur., Steganography, Watermarking Multimedia Contents VI*, San Jose, CA, USA, vol. 5306, pp. 70–83, Jun. 2004.
[19] F. Li, K. Wu, J. Lei, M. Wen, Z. Bi, and C. Gu, "Steganalysis over large-scale social networks with high-order joint features and clustering ensembles," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 2, pp. 344–357, Feb. 2016.
[20] A. D. Ker, "Locating steganographic payload via WS residuals," in *Proc. 10th ACM Workshop Multimedia Secur.*, 2008, pp. 27–32.
[21] A. D. Ker and I. Lubenko, "Feature reduction and payload location with wam steganalysis," *Proc. SPIE, Electron. Imag., Media Forensics Secur. XI*, San Jose, CA, USA, vol. 6072, pp. 0A01–0A13, Jan. 2009.
[22] W. Tang, H. Li, W. Luo, and J. Huang, "Adaptive steganalysis against WOW embedding algorithm," in *Proc. 2nd ACM Workshop Inf. Hiding Multimedia Secur.*, 2014, pp. 91–96.
[23] T. Denemark, V. Sedighi, V. Holub, R. Cogranne, and J. Fridrich, "Selection-channel-aware rich model for steganalysis of digital images," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2014, pp. 48–53.
[24] M. Goljan, J. Fridrich, and T. Holotyak, "New blind steganalysis and its implications," *Proc. SPIE, Electron. Imag., Secur., Steganography, Watermarking Multimedia Contents VIII*, San Jose, CA, USA, vol. 6072, pp. 1–13, Jan. 2006.
[25] P. Bas, T. Filler, and T. Pevný, "'Break Our steganographic system': The ins and outs of organizing boss," in *Proc. Int. Workshop Inf. Hiding*. Prague, Czech Republic: Springer, 2011, pp. 59–70.
[26] T. Pevny, J. Fridrich, and A. D. Ker, "From blind to quantitative steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 445–454, Apr. 2012.

QUANQI LIU received the B.S. degree in computer science and technology from Hangzhou Dianzi University, Hangzhou, China, in 2017, where he has been with the Laboratory of Internet and Network Security, since 2017. His current research interests include image steganography and steganalysis techniques.

TONG QIAO received the B.S. degree in electronic and information engineering from Information Engineering University, Zhengzhou, China, in 2009, the M.S. degree in communication and information system from Shanghai University, Shanghai, China, in 2012, and the Ph.D. degree from the Laboratory of Systems Modelling and Dependability, University of Technology of Troyes, Troyes, France. Since 2016, he has been an Assistant Professor with the School of Cyberspace, Hangzhou Dianzi University. His current research interests include steganalysis and digital image forensics.

MING XU received the M.S. and Ph.D. degrees from Zhejiang University, in 2000 and 2004, respectively. He is currently a Full Professor with Hangzhou Dianzi University. His research interest includes digital forensics.

NING ZHENG received the M.S. degree from Zhejiang University, in 1987. He is currently a Full Professor with Hangzhou Dianzi University. His research interest includes digital forensics.