DIGITAL IMAGE SELF-RECOVERY USING UNEQUAL ERROR PROTECTION

Saeed Sarreshtedari, Mohammad Ali Akhaee, Aliazam Abbasfar

School of Electrical and Computer Eng., College of Eng., University of Tehran, Tehran, Iran
s.sarreshtedari, akhaee, abbasfar@ut.ac.ir

ABSTRACT

In this paper, an unequal error protection (UEP)-based scheme is presented to generate tamper-proof images, in which the lost content of the original image is recoverable despite the malicious tampering. For this purpose, a representation of the original image is embedded into itself, after being protected by the proper channel coding. Since better protection is considered for the more important bits of the image representation through a dynamic programming (DP) optimization scheme, they survive higher tampering rates than the less important image information. As a result, the quality of the restored image degrades with respect to the true tampering rate.

Index Terms— Digital image self-recovery, Tamper-proof images, Unequal error protection (UEP)

1. INTRODUCTION

Since its emerge, digital image watermarking has been widely applied for image authentication and copy-right protection. However, recently its application is extended to digital image self-recovery, in which the lost content of the original image due to the tampering is recoverable with the help of information embedded into the image itself.

The common approach in such techniques is to embed a representation of the original image into itself [1, 2]. This information can be protected against tampering using proper channel codes [3]. However, it has been recently shown that the digital image self-recovery can be modeled as a source-channel coding problem [4]. In this approach, the source-coded image information is channel-coded using proper codes, and embedded into itself. Therefore, the quality of the restored image and the tolerable tampering rate (TTR) depends on the bit-rate dedicated to the source and channel code bits, respectively.

In source-channel coding based approaches, the recovery process fails when the tampering exceeds the rate tolerable by the applied channel coding. In this paper, a joint source-channel coding (JSCC) method using UEP is proposed, in which better protection is regarded for more important information. As a result, less information bits and thus lower quality of the restored image is accessible with the increase in the tampering rate. However, a rough representation of the original image is available even in the high tampering rates, where the previous source-channel coding schemes fail to work.

2. PROPOSED WATERMARKING METHOD

The proposed embedding method is presented in Figure 1. Original image is encoded at first, using the set partitioning in hierarchical trees (SPIHT) algorithm [5]. The SPIHT output bit-stream is composed of L bit-planes (BP), where higher (first,lower indexes) bit-planes contribute more to the image compression peak signal to noise ratio (PSNR), though they are shorter meanwhile. The length and PSNR share of each bit-plane i is denoted by b_i and δ_i, and they form the SPIHT rate-distortion (R-D) profile.

Another feature of the SPIHT bit-stream is that the lower bit-planes are usable only when the higher bit-planes are completely available. Therefore, the higher bit-planes must be carefully protected, while protecting the lengthy and less important lower bit-planes is not necessary. Dedication of the protection bit-budget to the SPIHT BP's is decided by a JSCC optimization module, shown in Figure 1. This module works based on the UEP rule, that is, more protection bit-budget is assigned to more important bit-planes. When the protection bit-budget is limited, UEP optimizer does not bother to protect the lowest bit-planes, and totally discard them. As a result, this module outputs the number of the BP's included in the protection process J, and the optimal protection determined for them (JSCC profile). The optimization process is discussed in more details in Section 3.

The protection information derived by the UEP scheme is passed to the Reed-Solomon (RS) channel encoder. Based on this information, the channel coding module fetches the J first BP's and outputs the channel coded compressed image bit-stream.

On the above path, the original image is decomposed into $B \times B$ blocks, and each block is further divided into $b_w = n_w \times B^2$ and $b_m = n_m \times B^2$ least and most significant bits (LSB and MSB), where $n_w + n_m = 8$. The n_w LSB per pixel are used for data embedding, while the rest n_m MSB are left as is. Applying MD5 hash algorithm, b_h hash bits are generated from the b_m MSB of each block. On the other hand, UEP optimizer decides the total length of the included SPIHT

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

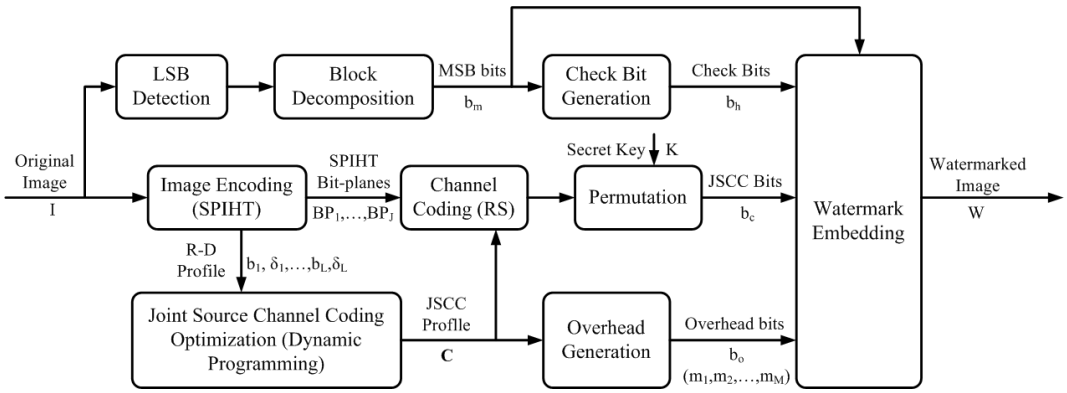


Fig. 1. Block diagram of watermark generation and embedding

BP's and protection information to be b_c bits per block. b_o bit per block is also assigned to record the UEP information that is properly channel coded to be later extracted to the decoder even in high tampering rates. Channel coding output bits undergo a permutation controlled by a secret key shared between the transmitter and receiver to increase the security. After that, each block hosts b_c JSCC bits, b_h hash bits of the same block, and b_o overhead bits. This $b_w = b_c + b_h + b_o$ bits form the watermark information of each block and replaces its b_w LSB of that block to generate the tamper-proof watermarked image.

At the receiver side, hash bits of blocks are reproduced and compared to those of embedded ones to determine the tampered blocks. List of the erased channel code symbols are generated from the list of the tampered blocks. With the help of this list and the RS channel erasure decoder, the overhead is decoded and the JSCC profile is extracted. JSCC profile helps the channel erasure decoder of the SPIHT bit-stream to determine which bit-planes are included and how protected they are, in addition to the list of erased symbols. As a consequence, channel erasure decoder manages to retrieve some higher SPIHT BP's, while it fails at recovering the less important bit-planes. The restored information replaces the lost content at the tampering area. A sketch of the watermark extraction and lost content recovery algorithm is given in Figure 2.

3. UNEQUAL ERROR PROTECTION AND DYNAMIC PROGRAMMING OPTIMIZATION

Our UEP problem here is to optimally protect SPIHT-encoded bit-planes. Kim *et. al.* adopted a progressive image transmission approach to develop a SPIHT-UEP scheme designed for Internet-based progressive image and video transmission [6]. In their proposed scheme, different parity byte lengths are assigned to different SPIHT bit-planes, depending on their importance. A DP approach is applied to find the optimal bit allocation according to the packet loss mass probability function of the channel.

Kim's work guarantees that the lowest bit-planes are lost before highest ones at every packet loss rate. This has inspired our JSCC scheme. The proposed UEP framework is shown in Figure 3. Assume that the bit-planes are going to be protected using M blocks of RS(N, k_i) codes over $GF(2^t)$ where $N = 2^t - 1$. In this way, SPIHT bit stream must be represented in t-bit symbols. Remaining bits of bit-plane i are appended to the beginning of the $(i+1)^{th}$ bit-plane when b_i is not divisible to t. Therefore, the length of bit-plane i will be S_i symbols calculated as follows:

$$S_i = \left\lfloor \frac{b_i + e_{i-1}}{t} \right\rfloor$$
$$e_i = b_i + e_{i-1} - S_i t, \quad e_0 = 0. \tag{1}$$

In our UEP scheme $k_i = N - c_i$ where c_i equals the optimum number of parity symbols obtained by JSCC optimizer to protect bit-plane i. The symbol length S_i of some bit-planes (especially the last lengthy ones) may exceed the k_i decided by the JSCC optimizer. In such cases, that bit-plane must be splitted into more than one part, each of which protected by one block of RS(N, k_i). Again, if the bit-plane symbol length S_i is not divisible to k_i, it will be further modified to symbol length B_i in a similar way:

$$B_i = \left\lfloor \frac{S_i + Ei - 1}{k_i} \right\rfloor$$
$$E_i = S_i + E_{i-1} - B_i k_i, \quad E_0 = 0. \tag{2}$$

Thus, the number of channel code blocks (protection layers) assigned to the bit-plane i is determined as $l_i = \frac{B_i}{k_i}$. Therefore, the UEP scheme is set up having the optimum channel code parity allocation vector $\mathbf{C}$ found by JSCC optimization where:

$$\mathbf{C} = (c_1, c_2, \ldots, c_J). \tag{3}$$

Having c_i found by optimization, k_i and l_i are available. Thus, we can define:

$$L(i, \mathbf{C}) = \sum_{j=i}^{J} l_j \tag{4}$$

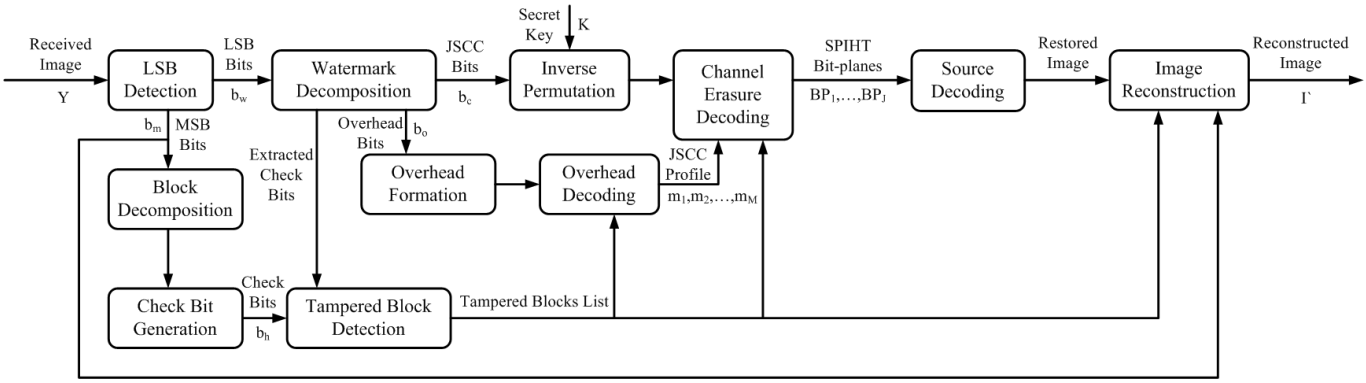


Fig. 2. Block diagram of tampering detection and image recovery

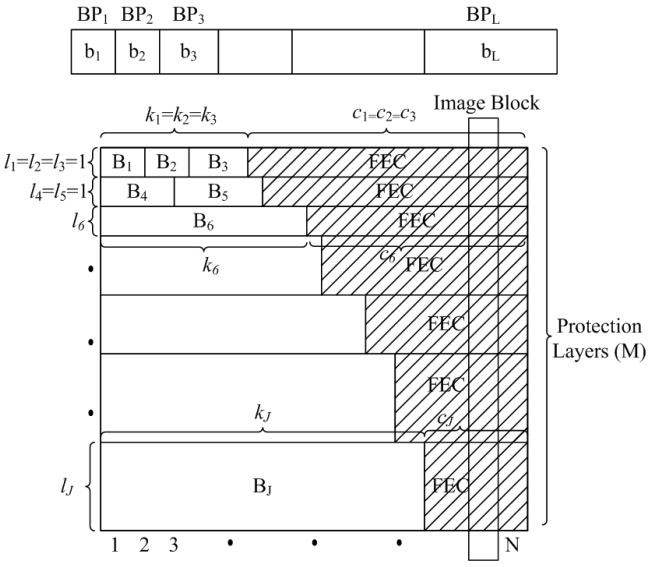


Fig. 3. Unequal error protection design

as the total channel coding blocks occupied by i'th to the last included bit-planes. Since the proposed method is a flexible scheme, TTR can be defined for the individual bit-planes rather than the whole image. According to such design the TTR of the bit-plane i equals to:

$$\text{TTR}(i) = \frac{N - k_i}{N} = \frac{c_i}{N}. \tag{5}$$

A DP optimization is applied to find the optimum parity allocation vector. According to Figure 1, SPIHT rate-distortion profile for the original image is available to the optimizer in terms of BP lengths $(b_1, \ldots, b_L)$ and their corresponding PSNR shares $(\delta_1, \ldots, \delta_L)$. According to the UEP design in Figure 3, every image block must include one symbol of every channel code block so that when an image block is lost, all channel code blocks are affected equally. It can be simply shown that in this way, $N = N_B - 1$. Thus, we have $t = \log_2(N + 1)$. Consequently, the number of protection layers will be $M = b_c/t$.

Original b_c must be chosen in a way to be divisible to t. Hereafter, from number of image blocks we mean N rather than N_B. In this way, higher bit-planes that are better protected will survive for the tampering rates higher than those of the lower ones. Assume that the total number of J bit-planes are included for protection by optimizer decision. Having c_i parity symbols dedicated to the bit-plane i, this bit-plane will survive if no more than c_i out of N image blocks are tampered. Therefore, defining $p(j)$ as the probability of having j blocks tampered, the average received PSNR will be:

$$\text{PSNR}(J, C) = \sum_{i=1}^{J} \left(\delta_i \sum_{j=0}^{c_i} p(j) \right). \tag{6}$$

Next we calculate $p(j)$. Assuming $\bar{\alpha}$ as the expected tampering rate $p(j)$ follows the binomial distribution:

$$p(j) = \binom{N}{j} \bar{\alpha}^j (1 - \bar{\alpha})^{N-j}. \tag{7}$$

For large N values, this distribution can be approximated as a Gaussian distribution with mean $\mu = N\bar{\alpha}$ and variance $\sigma^2 = N\alpha(1 - \alpha)$, i.e.

$$p(j) \approx \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(j - \mu)^2}{2\sigma^2} \right). \tag{8}$$

We define $P(j)$ as the cumulative distribution function (CDF) corresponding to $p(j)$, that is, $P(j) = \sum_{k=0}^{j} p(j)$. Using function $\Phi(x)$ as the CDF of the normal distribution, $P(j)$ can be approximated as:

$$P(j) = \Phi\left(\frac{j - \mu}{\sigma} \right). \tag{9}$$

Hence (6) can be rewritten as:

$$\text{PSNR}(J, \mathbf{C}) = \sum_{i=1}^{J} \delta_i P(c_i). \tag{10}$$

(a) (b) (c)

Fig. 4. (a) 512 × 512 gray-scale image protected against expected tampering rate of $\bar{\alpha}$=0.5 using 3-LSB is 56% tampered. (b) Tampering area is detected (c) Tampered content is recovered with PSNR=35.61 dB.

Table 1. Reconstruction Profile for Lena 512 × 512 3-LSB Protection

		TTR of bit-planes for different $\bar{\alpha}$ designs						
Bit-Plane	PSNR	0.1	0.2	0.3	0.5	0.6	0.75	0.85
1	13.25	0.92	0.92	0.92	0.92	0.99	0.99	0.99
2	14.27	0.92	0.92	0.92	0.92	0.92	0.92	0.97
3	14.5	0.92	0.92	0.92	0.92	0.92	0.92	0.97
4	15.02	0.92	0.92	0.92	0.92	0.92	0.92	0.97
5	16.37	0.92	0.92	0.92	0.92	0.92	0.92	0.97
6	18.27	0.92	0.92	0.92	0.92	0.92	0.92	0.97
7	20.96	0.92	0.92	0.92	0.92	0.92	0.92	0.97
8	23.16	0.92	0.92	0.92	0.92	0.92	0.92	0.95
9	25.95	0.87	0.87	0.87	0.87	0.87	0.87	0.93
10	29.1	0.85	0.85	0.85	0.85	0.85	0.85	0.9
11	32.37	0.71	0.71	0.71	0.71	0.81	0.81	0.88
12	35.61	0.45	0.45	0.64	0.64	0.73	0.78	0
13	38.8	0.28	0.28	0.46	0.46	0	0	0
14	43.02	0	0	0	0	0	0	0
15	48.86	0	0	0	0	0	0	0
16	55.23	0	0	0	0	0	0	0

Thus, the problem of finding optimum parity allocation can be stated as

$$\max_{\mathbf{C}} \text{PSNR}(J, \mathbf{C}) \quad s.t. \quad L(1, \mathbf{C}) \leq M, \tag{11}$$

which can be solved through a DP approach similar to [7]. Note that the case of discarding the current bit-plane must be also considered in the optimization stage of that bit-plane. Furthermore, our proposed DP scheme also considers the possibility of concatenating two or more bit-planes as a single bit-plane as long as the current bit-plane is fit into one channel coding block. These two cases of discarding the current bit-plane or appending it to the last one are the main differences of our proposed UEP scheme compared to [6] in which the number of protected bit-planes is fixed and no more than one bit-plane is protected in a single encoding block.

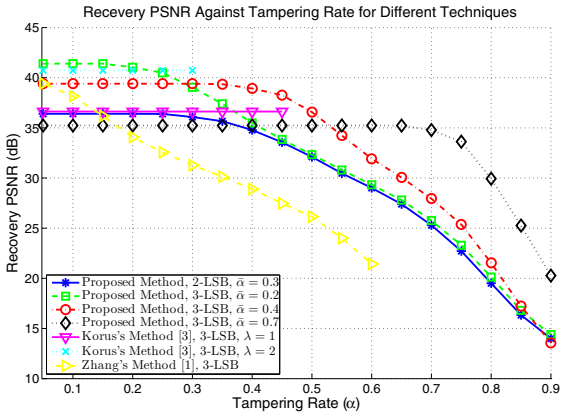
It is worth to mention that although the expected tampering rate $\bar{\alpha}$ is included in the above design, it must not be necessarily known to the embedder prior to the watermarking. Instead, this very quick optimization process can be run for a variety of the possible tampering rates to find the optimum protection profiles. One of these optimum profiles will be then chosen and implemented based on the desired tampering-recovery performance.

4. EXPERIMENTAL RESULTS

In this Section, we present the results of some experiments conducted to verify the efficiency of the proposed framework. Following at first, the performance of dynamic programming is investigated. Then the restoration performance of the proposed framework is shown through some sample recovered image. Finally in this section, the performance of the proposed framework is compared to the most significant state of the art image-recovery methods to confirm its superiority.

Table 1 summarizes the results of DP optimization for 512×512 Lena image for tampering rates varying from 0.1 to 0.85. It can be seen that for higher expected tampering rates, the UEP optimizer sacrifices less significant bit-planes to provide better protection for more important bit-planes. On the other hand, it allows more bit-planes to be included in the protection process for lower risks.

Image recovery performance of the proposed framework is shown in Figure 4. Figure 4(a) shows the original gray-scale 512× 512 Lena image after being protected against $\bar{\alpha} = 0.5$ using a 3-LSB scheme and then being tampered at the rate of $\alpha = 56\%$. As shown in Table 1, 13 SPIHT bit-planes are included in UEP and channel coded. Since a 3-LSB scheme is applied, PSNR of the watermarked image equals 37.9 dB. Figure 4(b) shows how the tampering area is detected by check bit examination. Finally, Figure 4(c) shows how the proposed image recovery scheme have managed to restore the content in the lost area. Image quality of the re-



(a)

Fig. 5. Simulation results for different methods expressed in terms of the recovered PSNR in tampered area against tampering rate.

stored area compared to the original image equals 35.61 dB. According to Table 1, this value means that 12 bit-planes have survived tampering as it was expected.

Now we compare the recovery performance of the proposed framework to the state-of-the-art works of Zhang [1] and Korus [3]. One thousand randomly chosen 512× 512 gray-scale images from BOWS2 image dataset [8] were applied to fairly compare the performance of these methods. For each tampering rate α, α percent of image blocks were selected and randomly tampered. Results for tampering restoration performance in each tampering rate were averaged over entire sample database and shown in Figure 5. Our algorithm was implemented in 4 cases: A 2-LSB version designed for $\bar{\alpha} = 0.3$, and three 3-LSB versions for $\bar{\alpha} = 0.2, 0.4$ and 0.7. All algorithms imposed the same embedding distortion to the original image since they use three LSB for embedding, except the proposed 2-LSB scheme that offers significantly higher quality of the watermarked image measured around 6 dB in terms of PSNR.

From Figure 5 it can be seen that while the restoration capability of Korus's is limited to $\alpha = 0.5$, different versions of our 3-LSB scheme offers restoration with fair quality even for 80% tampered images. Meanwhile, the restoration quality of our scheme is almost the same or even better for tampering rates below Korus's TTR. Although the main advantage of our 2-LSB algorithm is its 6 dB less distortion imposed to the original image compared to the other 3-LSB techniques, its recovery performance is yet comparable to Korus's method and quite superior to the Zhang's method.

5. CONCLUSION

A novel technique to generate tamper-proof images based on the unequal error protection was introduced in this paper. Original image is compressed using SPIHT encoder, and the output bit-stream is then channel coded to exhibit robustness against tampering. In this scheme and according to a dynamic programming optimization technique, channel code parity bits are optimally assigned to the SPIHT bit-planes such that the more important bits enjoy more protection. As a result, the quality of the restored image is high when the tampering rate is low, and the tampered image can be recovered even in the very high tampering rates. Simulation results confirm the superiority of the proposed method compared to the recent techniques, in terms of both the tampering rate and the quality of the restored image.

REFERENCES

[1] X. Zhang, Z. Qian, Y. Ren, and G. Feng, "Watermarking with flexible self-recovery quality based on compressive sensing and compositive reconstruction," *Information Forensics and Security, IEEE Transactions on*, vol. 6, no. 4, pp. 1223–1232, 2011.

[2] Z. Qian, G. Feng, X. Zhang, and S. Wang, "Image self-embedding with high-quality restoration capability," *Digital Signal Processing*, vol. 21, no. 2, pp. 278 – 286, 2011.

[3] P. Korus and A. Dziech, "Efficient method for content reconstruction with self-embedding," *Image Processing, IEEE Transactions on*, vol. 22, no. 3, pp. 1134–1147, 2013.

[4] S. Sarreshtedari and M.A Akhaee, "Source-channel coding approach to generate tamper-proof images," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 7435–7439.

[5] A. Said and W.A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 6, no. 3, pp. 243–250, 1996.

[6] J. Kim, R.M. Mersereau, and Y. Altunbasak, "Error-resilient image and video transmission over the internet using unequal error protection," *Image Processing, IEEE Transactions on*, vol. 12, no. 2, pp. 121–131, Feb 2003.

[7] V. Chande and N. Farvardin, "Progressive transmission of images over memoryless noisy channels," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 6, pp. 850–860, June 2000.

[8] "The dataset from the 2nd bows contest.," (2012, Mar. 26) [Online]. Available: http://bows2.ec-lille.fr/.