Reversible Data Hiding with Histogram-Based Difference Expansion for QR Code Applications

Hsiang-Cheh Huang, *Senior Member*, IEEE, Feng-Cheng Chang, *Member*, IEEE,
and Wai-Chi Fang, *Fellow*, IEEE

**Abstract — ***In this paper, we propose a new algorithm in reversible data hiding, with the application associated with the quick response (QR) codes. QR codes are random patterns, which can be commonly observed on the corner of posters or webpages. The goal of QR codes aims at convenience-oriented applications for mobile phone users. People can use the mobile phone cameras to capture QR code at the corner of web page, and then the hyperlink corresponding to the QR code can be accessed instantly. Since QR code looks like random noise and it occupies a corner of the original image, its existence can greatly reduce the value of the original content. Thus, how to retain the value of original image, while keeping the capability for the instant access for webpages, would be the major concern of this paper. With the aid of our reversible data hiding technique, the QR codes can be hidden into the original image, and considerable increase in embedding capacity can be expected. Next, we propose a scheme such that when the image containing the QR code is browsed, the hyperlink corresponding to the QR code is accessed first. Then, the QR code could get vanished and the original image would be recovered to retain the information conveyed therein. Simulation results demonstrate the applicability of the proposed algorithm* [1].

Index Terms — Reversible data hiding, security, quick response (QR) codes, information protection.

I. INTRODUCTION

The proliferation of Internet usage has made people linking to the web pages easily by using PC, PDA, or mobile phone over the wired or wireless networks. Particularly, for users using the mobile phones to browse the web pages, it has brought much more conveniences to their daily lives [1][2]. As people know, in comparison with the time consumption between the computer keyboard and the mobile phone keypad for inputting the URL (Uniform Resource Locator), by using the mobile phone keypad brings much more inconveniences and difficulties for linking to the web pages. To solve this problem, the quick response (QR) code [3] has emerged.

The QR code can be easily seen from web pages or on posters nowadays. It is a two-dimensional code in square shape, mostly represented by binary form (black and white pixels), attached somewhere in the web pages or posters. It is easily recognizable because it looks like a random pattern. Colorized QR codes are also in existence. At the beginning, the purpose for the QR code is to utilize the quick connection to the specific web page with the URL information converted to the QR code pattern. And from the viewpoint of data hiding researches [4][5], QR code can be regarded as the visible watermark. Due to the fact that visible watermark cause the degradation of image quality both objectively and subjectively, how to effectively alleviate the effect caused by the visible watermark, and to retain the information conveyed in the original image, seem an interesting topic for applications. With this kind of requirements, reversible data hiding provides an effective means to fulfill these requirements.

In this paper, we propose a new algorithm in reversible data hiding, and the associated integration that can utilize the capability of the QR code. Reversible data hiding [6]-[10] is a newly developed branch in watermarking researches. Unlike conventional watermarking techniques that only the watermark needs to be extracted and examined at the receiver, reversible data hiding requires that both the hidden data and the original multimedia, an image for instance, should be perfectly recovered. For reversible data hiding, the hidden information needs to be embedded into the original image by algorithm designers, and more importantly, it requires that both the hidden information and the original image should be perfectly recovered at the decoder. Thus, how to reach the perfect recovery of the hidden information and the original image is a major task for the design of algorithm. Moreover, for the application of QR code with our reversible data hiding algorithm, in addition to the removal of the QR code, it can effectively recover the original image. Once the image containing the QR code is browsed, the designated web page is accessed, and the original image is recovered back by use of the reversible data hiding techniques.

This paper is organized as follows. In Sec. II, we review two different kinds of reversible data hiding techniques in literature. Next, in Sec. III, we depict the proposed algorithm

---

[1] This work was supported in part by the Taiwan National Science Council, under Grant Number: NSC99-2221-E-390-023, NSC-99-2220-E-009-028, NSC-99-2220-E-009-030, and NSC-99-2220-E-009-071. This work was also partially supported by the UST-UCSD International Center of Excellence in Advanced Bio-engineering sponsored by the Taiwan National Science Council I-RiCE Program under Grant Number: NSC-99-2911-I-010-101.

H. C. Huang is with Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung 811, Taiwan, ROC (e-mail: huang.hc@gmail.com).

F. C. Chang is with Department of Innovative Information and Technology, Tamkang University, Ilan 262, Taiwan, ROC (e-mail: breeze833@gmail.com).

W. C. Fang is with Department of Electronics Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, ROC (e-mail: Dr.WFang@gmail.com).

Contributed Paper
Manuscript received 03/19/11
Current version published 06/27/11
Electronic version published 06/27/11.

with the characteristics of abundant capacity and little side information. Then, in Sec. IV, we present some fundamental descriptions of the QR codes and the integration of QR codes with our algorithm. Simulation results are demonstrated in Sec. V, which suggest the superiority of the proposed algorithm, and the applicability of the algorithm and the integration proposed. Finally, we conclude this paper in Sec. VI.

II. CONVENTIONAL SCHEMES IN REVERSIBLE DATA HIDING

Reversible data hiding is a new branch in data hiding researches. At the encoder, the data are hidden into original image, and output looks very similar, or even identical, to original image. At the decoder, both the hidden data and the original image should be perfectly recovered. There are two major branches in reversible data hiding; one is the histogram-based scheme, and the other is the difference expansion technique. They are described as follows.

A. Histogram Modification for Reversible Data Hiding

The histogram-based scheme [8] is famous for its ease of implementation and little overhead or side information generated. The histogram of the original image is slightly modified to hide the information at the encoder. With the reverse operations corresponding to the encoder, both the original and hidden information can be perfectly recovered at the decoder. Embedding procedures of data can be described in the following steps.

Step 1. *Generate the histogram of original image.* Luminance values of the original image are integers between 0 and 255, and they can be represented by 8 bits. The luminance with the maximal occurrences in histogram, denoted by a, is labeled as “max point,” while that with no occurrence is labeled as “zero point,” and the value can be denoted by b. Without loss of generality, we suppose that the luminance value with the zero point is larger than that with the max point, or $b > a$.The luminance values of “max” and “zero” points, each is represented by 8 bits, are treated as side information. Hence, a total of 16 bits should be transmitted to the receiver for data extraction.

Step 2. *Select the range between max and zero points.* The range of luminance values between max and zero points, or $[a, b]$, is recorded in the histogram. It can be recognized by use of the side information.

Step 3. *Modify of luminance values in selected range.* In the region between max and zero points recorded in Step 2, luminance values between $[a, b]$ are altered in advance. Those in the selected range are all increased by 1.

Step 4. *Embed the information.* For the embedding of binary information, if the bit value is ‘1,’ keep the luminance value the same as $(a+1)$; if the bit value is ‘0,’ the decrease the luminance value to become a.

In extracting both the hidden data and the original image, the following steps should apply accordingly.

Step 1. *Locate selected range with side information.* After receiving the 16-bit side information, luminance values between the max and zero points are compared.

Step 2. *Extract the hidden data relating to the original.* Every pixel in the output image is scanned and examined sequentially to extract the data bits to compare to Step 3 of the embedding procedure. After scanning the received image sequentially, one of the three cases will be performed.

- ➢ If the luminance value is equal to a, output bit ‘0’ for the hidden information.
- ➢ If the luminance value is equal to $(a+1)$, output bit 1.
- ➢ For all the other luminance values, there’s no output for the hidden information.

Step 3. *Obtain the original image.* By moving the histogram into its original form, the original content is recovered.

The histogram-based reversible data hiding has the advantages of ease of implementation and little side information produced. In addition, because the modification of luminance value is at most one, it leads to the result that the maximum value of mean square error would be at most one. Consequently, the PSNR would be at least 48.13 dB, leading to the guaranteed result in output image quality [8]. On the contrary, the number of bits for embedding, or the capacity, might not be enough for residing all the data to be hidden. Capacity is limited by the number of occurrences of the max point. Hence, the difference expansion (DE) scheme described in Sec. II-B, based on the concept of wavelet transform, was proposed.

B. Difference-expansion for Reversible Data Hiding

The difference expansion (DE) method is one of the earliest schemes for reversible data hiding [6][7]. It follows the concepts directly from wavelet transforms by turning the spatial pixel values into frequency coefficients. In DE, every two neighboring pixels should be grouped together as a pair (x, y). Next, simple calculations can be performed by

$$l = \left\lfloor \frac{x+y}{2} \right\rfloor \text{, and} \tag{1}$$

$$h = x - y, \tag{2}$$

where l and h can be regarded as the lower and higher frequency bands in wavelet transform, respectively, and $\lfloor \cdot \rfloor$ denotes the floor function, or the least nearest integer. We can also find that h is the difference value between the pair of neighboring pixels. For the embedding of one bit b, $b \in \{0,1\}$, in the pair (x, y), the idea is to keep the lower frequency band the same, and only the higher frequency band is modified by

$$\widetilde{h} = 2 \cdot h + b. \tag{3}$$

Thus, $\widetilde{h}$ is called the difference expansion (DE) scheme for hiding information. From Eq. (3), we can find that one bit can be hidden into two pixels in general.

For obtaining the image containing the hidden information, the new pair (x', y'), which serves as the output, can be calculated by

$$x' = l + \left\lfloor \frac{\tilde{h}+1}{2} \right\rfloor \text{, and} \tag{4}$$

$$y' = l - \left\lfloor \frac{\tilde{h}}{2} \right\rfloor . \tag{5}$$

Due to the fact that pixel values should lie between 0 and 255, we set the requirement that x', y' are integers and are in the range of $0 \le x', y' \le 255$. Therefore, some pairs in the original image may be unsuitable for data embedding because they lie beyond the required range. A location map [9][10], which contains the location information of all suitable pairs, is served as the side information for recovering both the original image and the hidden data. However, under the worst case, when every pair is unsuitable for data hiding, the embedding capacity is 0 bit, and the size of the location map can be as high as $\frac{1}{2}MN$-bit for an original image with size $M \times N$, which is a major drawback for this scheme. Fortunately, this kind of situation can seldom be found for natural images.

At the decoder side, both the original image and the hidden data should both be recovered. For obtaining the original image, the original pair should be calculated by using (x', y'). First, we calculate and derive the frequency bands as follows:

$$l' = \left\lfloor \frac{x'+y'}{2} \right\rfloor \text{, and} \tag{6}$$

$$h' = x' - y' . \tag{7}$$

We can verify the perfect recovery of original frequency bands that

$$l' = \left\lfloor \frac{x'+y'}{2} \right\rfloor = \left\lfloor l + \frac{1}{2}\left(\left\lfloor \frac{\tilde{h}+1}{2} \right\rfloor - \left\lfloor \frac{\tilde{h}}{2} \right\rfloor \right) \right\rfloor = l \text{, and} \tag{8}$$

$$h' = x' - y' = \left\lfloor \frac{\tilde{h}+1}{2} \right\rfloor + \left\lfloor \frac{\tilde{h}}{2} \right\rfloor = \tilde{h} . \tag{9}$$

Next, we calculate the recovered pixel pair, (x'', y''), by

$$h'' = \left\lfloor \frac{h'}{2} \right\rfloor = \left\lfloor \frac{\tilde{h}}{2} \right\rfloor = h , \tag{10}$$

$$x'' = l' + \left\lfloor \frac{h''+1}{2} \right\rfloor \text{, and} \tag{11}$$

$$y'' = l' - \left\lfloor \frac{h''}{2} \right\rfloor . \tag{12}$$

We can verify that

$$x'' = l' + \left\lfloor \frac{h''+1}{2} \right\rfloor = l + \left\lfloor \frac{h+1}{2} \right\rfloor = x \text{, and} \tag{13}$$

$$y'' = l' - \left\lfloor \frac{h''}{2} \right\rfloor = l - \left\lfloor \frac{h}{2} \right\rfloor = y . \tag{14}$$

With Eq. (13) and Eq. (14), we can easily find that $(x'', y'') = (x, y)$, meaning that the original image pair can be recovered. Finally, for obtaining the embedded bit, we just retrieve the least significant bit (LSB) in Eq. (3), and then the hidden information can be recovered.

For increasing the capacity, and hence more information can be hidden into original image, the variant for DE, called difference expansion of quads (DEQ) [7], can be devised as follows. If we group every 2×2 block into a unit, called the 'quad,' reversible data hiding can be performed by using the relationships among the four pixels. A quad is a 1×4 vector (u_1, u_2, u_3, u_4) formed from four pixel values in a 2×2 block. By following DE [10][11], we then calculate the following values, which serve as the frequency bands:

$$v_0 = \left\lfloor \frac{u_0 + u_1 + u_2 + u_3}{4} \right\rfloor , \tag{15}$$

$$v_1 = u_1 - u_0 , \tag{16}$$

$$v_2 = u_2 - u_0 \text{, and} \tag{17}$$

$$v_3 = u_3 - u_0 . \tag{18}$$

For embedding three bits, b_1, b_2, b_3 into one quad, we have

$$\tilde{v}_1 = 2 \cdot \left\lfloor \frac{v_1}{2} \right\rfloor + b_1 , \tag{19}$$

$$\tilde{v}_2 = 2 \cdot \left\lfloor \frac{v_2}{2} \right\rfloor + b_2 \text{, and} \tag{20}$$

$$\tilde{v}_3 = 2 \cdot \left\lfloor \frac{v_3}{2} \right\rfloor + b_3 . \tag{21}$$

By doing so, we can find that three bits can be hidden into a quad, or four pixels. Thus, embedding capacity for DEQ is 0.75 bit/pixel, while its counterpart for DE is 0.5 bit/pixel. It means that with DEQ, we can embed 50% more capacity than that with DE. On the other hand, regarding to the location map, under the worst case, the embedding capacity is 0 bit, the size of the location map has $\frac{1}{4}MN$-bit with DEQ. Even though the increase of capacity is obtained, the size of the location map is still a problem for practical applications.

C. Advantages and Drawbacks between DE and Histogram-based Schemes

As we described above, we observe that implementations of both the histogram-based and DE/DEQ methods are fundamentally different. We summarize the advantages and drawbacks of both methods from different aspects as follows.

➢ **Amount of side information**: The histogram-based scheme performs better than the DE/DEQ ones with this respect. For the histogram-based method, the side information is composed of the luminance values of both the max and zero points $[a, b]$ in Sec. II-A, meaning 2 bytes of side information need to be delivered to the decoder [8]. For the DE method and its variants [6][7], the location map plays an important role in extracting the hidden data and recovering the original, which depends on the characteristics of the original image. The location map serves as the side information, with considerable amount in size. Let the size of original image be 512×512. Under the worst case, the side information for DE and DEQ are 131072 bits and 65536 bits, respectively, and these values are much larger than that with the histogram-based one.

- **Data capacity**: The DE/DEQ schemes generally perform better than the histogram-based one with this perspective. For the histogram-based method, the data capacity is constrained by the maximal number of occurrences in the histogram. For a pixel represented by 8-bit, with the luminance values ranging from 0 to 255, the maximal number of occurrences is generally below 10% of the total number of pixels, or 0.1 bit-per-pixel (bpp) [8]. On the other hand, for the DE scheme, the data capacity can be as high as 0.5 bpp, since every pair of consecutive pixels can carry 1 bit information [7]. We can see that the DE method can hide a much larger capacity than the histogram-based method.

By taking the advantages into consideration, we can employ the low overhead property in the histogram-based method and the high data capacity characteristic in the DE method, and we propose an applicable integration of both schemes in this paper.

III. PROPOSED ALGORITHM

Here we describe the reversible data hiding algorithm that considers the advantages in both the histogram-based and the DE schemes. We try to increase the capacity with the histogram-based method. According to simulations in Table I, when we partition the original image with size of $M \times N$ into four quarter-sized sub-images with sizes of $\frac{M}{2} \times \frac{N}{2}$, the maximal number of bit for embedding with the histogram-based method will increase from 6% to 82%, with somewhat increased amount of side information. For the histogram-based scheme, the side information contains the luminance values of peak and zero points (a, b), meaning that 16 bits of overhead would be produced. With the sub-image scheme, a total of 40 bits, including the luminance values of the zero point in the whole image, and four peak points in each sub-image, can be generated. The increased capacity, depicted in Table I, is designed to reside the location map. Next, the information to be hidden can be embedded with the DEQ scheme.

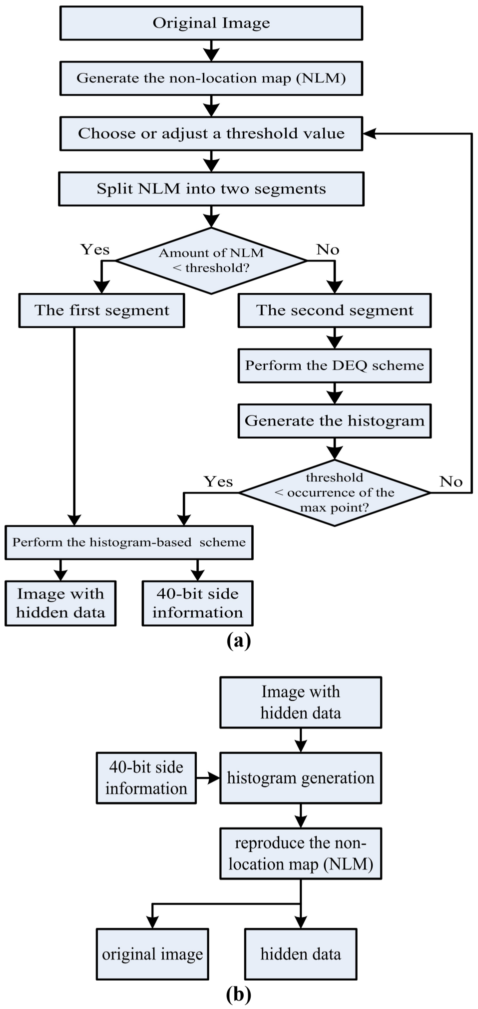
A. *Algorithm for Data Embedding*

Here we present the algorithm for reversible data embedding. Our goal is to hide both the location map and the data to be hidden into the original image, and only 40-bit of side information is produced.

Regarding to the location map in DEQ, we try to minimize its size for the reduction of side information. In general, the number of suitable quads for data hiding is much more than that of the unsuitable quads. Hence, we record the locations of quads that are unsuitable for embedding. By following the concept of delta modulation (DM) [12], the distances between the locations of two consecutive, unsuitable quads are calculated to decrease the amount of side information. This information is called the non-location map (NLM).

TABLE I
AMOUNT OF INCREASE IN EMBEDDING BIT WITH THE HISTOGRAM-BASED SCHEME FOR THE FOUR QUARTER-SIZED SUB-IMAGES

Test image	Lena	airplane	baboon
Capacity with original image	2823	9620	2773
Capacity with 4 sub-images	5139	16105	2959
Amount of increase	82%	67%	6%



(a)

(b)

Fig. 1. The flowchart for (a) reversible data embedding, and (b) reversible data extraction, with our algorithm.

The flowchart for implementing the embedding of our algorithm can also be found in Fig. 1(a). By following the preliminary description at the beginning of Sec. III, embedding can be performed as follows.

Step 1. Perform calculations in DEQ, and record the positions unsuitable for embedding. Generate the non-location map (NLM) for DEQ in the original image.

Step 2. Choose a threshold for embedding the information at the beginning of NLM with histogram-based scheme.

Step 3. Split the original image into four sub-images, generate the four histograms of sub-image, and determine the occurrences for four peak points.

Step 4. If the sum of peak occurrences is larger than the threshold, hide the beginning of NLM into the histogram. If not, lower the threshold value in Step 2.

Step 5. Embed the remaining NLM information with the DEQ method.

Step 6. Output both the image with hidden information, and the 40-bit side information.

By doing so, the hidden data can be embedded into the suitable quads in the original image, and only the side information of 40-bit needs to be transmitted to the decoder.

B. Algorithm for Extraction of Hidden Data and Recovery of Original Image

On the other hand, extraction of data and original is relatively simple, and the flowchart is illustrated in Fig. 1(b). Extraction process for hidden data and original image can be performed as follows.

Step 1. Generate the four histograms corresponding to the four sub-images in the received image.

Step 2. Locate selected range with the 40-bit side information.

Step 3. Produce the beginning of NLM locations from the histogram.

Step 4. Extract the data and recover the original with the DEQ scheme.

With the 40-bit side information obtained from the encoder, the non-location map is reproduced correctly, and then the hidden information and the original image can be recovered with the DEQ.

IV. REVERSIBLE DATA HIDING WITH THE QR CODES

A. Background Descriptions of QR Codes

The QR (quick response) code is a 2-dimensional bar code, created by Japanese corporation Denso-Wave in 1994. It is also standardized by Japanese Industrial Standards (JIS), with the name `JIS-X-0510: 1999 QR Code` [2]. QR Codes can be easily seen from web pages, or advertisements in posters or newspapers. Users can capture the QR code from the newspaper with the mobile phone camera, and the webpage corresponding to the QR code can be accessed instantly. One example is depicted in Fig. 2(a). This QR code contains the URL information of the website of original image in Fig. 2(b), or http://www.lenna.org/. We take the commonly seen test image `Lena` with the image size of 1024×1024 for instance. After encoding, the binary image in square shape with the size of 135×135 is produced. Besides, Fig. 2(b) denotes the practical scenario for the utility of the QR code. The QR code is inserted into the corner of the original image by removing the pixel values at the lower-right portion of the original image.

The major purpose for the QR codes is for mobile phone users to link to the web page corresponding to the QR code quickly. Most mobile phones can read this code by using the camera on the phone, then the hyperlink information contained in the QR Codes can be deciphered, and the web page can be displayed on the screen of the mobile phone. In comparison with conventional schemes for accessing the homepages with the mobile phones, users need not type the alphanumeric characters in the URL; by capturing the QR Code with the mobile phone camera, the webpage can be shown instantly and lots of time for inputting the alphanumeric characters can be saved. However, the QR code still appears in the original image, and hence the degraded quality of image can be expected.

(a)

(b)

Fig. 2. Test materials for grey level test image in this paper. (a) The QR code with size 135×135, containing the hyperlink of http://www.lenna.org/. **(b) The grey level image `Lena` with size 1024×1024, containing the QR code at lower-right corner.**

Different from the conventional bar codes, the QR codes offer much more capacities for hiding information, which can be classified as follows:

- Numeric only: at most 7089 characters;
- Alphanumeric: at most 4296 characters;
- Byte format: at most 2953 bytes;
- Japanese character: at most 1817 characters.

Since the QR code can be captured by mobile phone cameras, some errors might be induced, and hence the captured QR code needs to have some error correcting capabilities. Consequently, the QR code can correct 7% to 30% of the received codeword based on different error correction levels by using the Reed Solomon codes [13].

From the watermarking perspective, the QR code can be regarded as the visible watermark. For instance, at the lower-right portion of Fig. 2(b), the pixels in the original image of this region are directly replaced by the QR code. After capturing the QR code, further procedures, such as shopping online, or obtaining more information about the image itself, can be performed with the browsers. Even though this brings conveniences to the access of web pages, quality degradation of original image can be expected even though only $\frac{135}{1024} \times \frac{135}{1024} \times 100\% = 1.74\%$ of the total image area is occupied. The peak signal-to-noise ratio (PSNR) is only 22.81 dB in Fig. 2(b). In addition, it is sometimes inevitable that important information of the original image might reside in the corner portions. By replacing the corner portions of the original image with the QR code might remove the inherent information conveyed. Thus, we propose the application by using reversible data hiding to hide the corner portion of original image into the rest of the original image in advance, and replace such a portion by the QR code. After browsing the image containing the QR code, the QR code is removed first, and the original data can be recovered back with reversible data hiding from the rest of the image.

B. Message Selection and Generation of QR codes

At the beginning, we select the URL corresponding to the original image. Next, the QR code is produced by the QR code generator, which is available online [14]. Then, the QR code is prepared to be placed at the corner of the original image. On the one hand, if we add the GPS information into the image, users can access the digital map conveniently. On the other hand, if we add the product information into the image, viewers can make evaluations instantly, and such a product shown in the original image may be purchased online subsequently. There is a wide variety in the selection of messages, and the QR code can be produced accordingly to meet the users' needs.

C. Integration with Reversible Data Hiding

Once the size of the QR code is determined, say, 135×135, it is ready to be placed at the bottom-right corner of the original image. Thus, pixels in such a region, consisting of $135 \times 135 \times 8$ bits, should be spread into the rest of the original image for reversible data hiding. The embedding and extraction of QR code follows directly from the schemes in Sec. III-A and Sec. III-B, respectively.

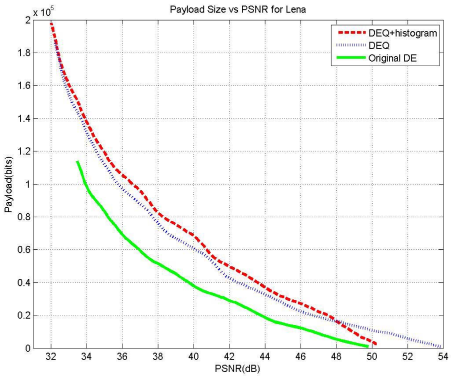
V. EXPERIMENTAL RESULTS

We first evaluate the performance of the proposed algorithm by comparing the output image quality and the embedding capacity with existing schemes in [6] and [7]. Next, we choose the commonly used grey-scale test image, and the ordinary color picture taken by ourselves to demonstrate the applicability of the proposed algorithm.

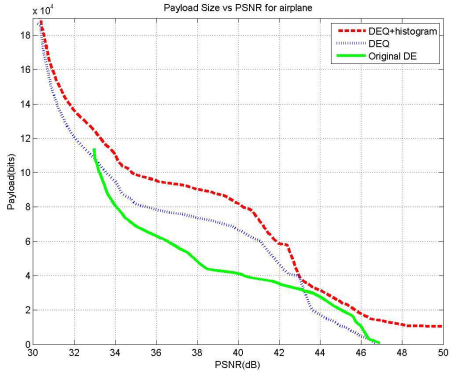
A. Performance Evaluation with the Proposed Algorithm

We have conducted several experiments to examine the effectiveness of the proposed algorithm. We choose three test images, namely, `Lena`, `airplane`, and `baboon`, for making tests. Performances with three algorithms, including the proposed algorithm, the DEQ algorithm [7], and the original DE algorithm [6] in Sec. II-B, are also measured. With the large amount of data to be hidden, the histogram-based scheme fails to work properly, and hence we discard the performance comparisons with the histogram-based scheme.

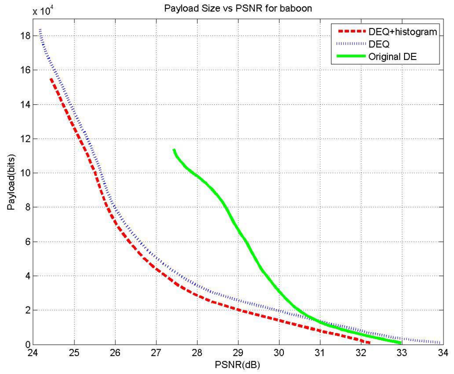
From the results of the `Lena` and `airplane` images in Fig. 3(a) and Fig. 3(b), respectively, we can see that under the same image quality represented by PSNR, the capacity with the proposed algorithm, represented by "DEQ+histogram," is generally larger than that of the DEQ and conventional DE schemes. Under the same capacity, the PSNR values are generally larger with our algorithm. Our algorithm outperforms the DE and DEQ algorithms for the generally smooth images.



(a) Results with `Lena`



(b) Results with `airplane`



(c) Results with `baboon`

Fig. 3. Comparisons between the embedded capacity and image quality.

For highly active images like `baboon`, the performances with our algorithm, the DE, and DEQ schemes have their own advantages. On the one hand, in Fig. 3(c), we have the results with the baboon test image. Since the contents in baboon image is highly active, the number of unsuitably embedded pairs or quads becomes much larger than Lena and airplane images. Thus, the size of non-location map (NLM) grows much larger, which deteriorates the result of our algorithm. We can see that our algorithm and the DEQ algorithm have similar performances, while the result with the conventional DE algorithm performs better than the other two. On the other hand, when we take the data capacity into account, the DE scheme can hide 114000 bits, while the DEQ scheme and ours

TABLE II
COMPARISONS OF IMAGE QUALITIES WITH THE QR CODES.

Test image	`Lena`	`airplane`	`baboon`	`pepper`
Image size	135×135	135×135	135×135	135×135
Image quality with QR code	22.81 dB	22.08 dB	21.71 dB	21.59 dB
QR information	http://www. lenna.org	http://www. nuk.edu.tw	http://www. google.com	http://www. yahoo.com.tw
MSE between original and recovered images	0.00	0.00	0.00	0.00

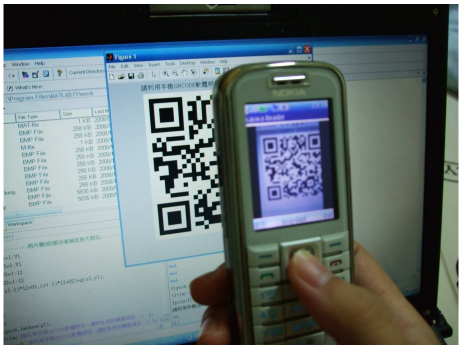


Fig. 4. A demontration for a part of our simulations.

The Lenna Story - www.lenna.org

Imaging Experts Meet Lenna in Person

Yes, it's true! Lenna attended the 50th Anniversary IS&T conference in Boston held in May 1997.

Check out the media coverage from Playboy, Wired, and more from Playboy.

Click here for more info and pictures from the conference.

Introduction

The Lenna (or Lena) picture is one of the most widely used standard test images used for compression algorithms. The comp.compression FAQ says the following:

> For the curious: 'lena' or 'lenna' is a digitized Playboy centerfold, from November 1972. (Lenna is the spelling in Playboy, Lena is the Swedish spelling of the name.) Lena Soderberg (ne Sjööblom) was last reported living in her native Sweden, happily married with three kids and a job with the state liquor monopoly. In 1988, she was interviewed by some Swedish computer related publication, and she was pleasantly amused by what had happened to her picture. That was the first she knew of the use of that picture in the computer business.

Fig. 5. After decoding, both the web page corresponding to QR code and the original test image Lena in grey level can be obtained.

can hide 184000 and 155000 bits, respectively, meaning the improvements of 61% and 36%. Even though our algorithm performs not as good as the DEQ scheme when the image is highly active, it still outperforms the conventional DE scheme from this perspective.

Practically, most images tend to have smooth characteristics. Therefore, the proposed algorithm should have the potential for real application, as depicted in Sec. V-C and Sec. V-D, respectively.

B. *Application with the Quick Response Codes*

We perform the following items for assessing the applicability of our algorithm. At the encoder:

- generate the QR code;
- embed QR code with proposed algorithm;
- post the image containing QR code in some web page.

At the decoder:

- access the image containing QR code with the browser;
- decode the QR code information and remove the QR code;
- recover the original image.

For the generation of QR codes with URL information in Table II, the QR codes have the sizes of 135×135. For all the pictures in our simulations, $135\times135\times8=145800$ bits at the lower-right portion of original image should be hidden with reversible data hiding, thus, our algorithm is capable of hiding such an amount of data.

C. *Application with the Grey-level Images*

Figure 4 is the demonstration for our simulations. In Fig. 5, we show the application with the grey-level image in Fig. 2. At the beginning, the lower-right part of the original image with the size of 135×135, containing 145800 bits, is spread into the other parts of the original image with our algorithm in Sec. III. With the simulation results in Sec. V-A and in Fig. 3, the 145800 bits can be hidden into the original image since the capacity is enough in our reversible data hiding algorithm. After that, the QR code is directly put into the lower-right corner of the image, and the image containing the QR code can open to the public. After inserting the QR code, the image qualities have degraded to 21.59 to 22.81 dB, shown in Table II. Even though the QR code brings conveniences to the users, it deteriorates the quality of original image.

Next, when the image containing the QR code is observed by the viewer, the QR code at the corner is browsed by the mobile phone camera, and the corresponding website can be accessed instantly. Later on, the QR code at the corner is directly discarded, and the original image can be recovered with the 40-bit side information.

Fig. 5 depicts such a scenario for the grey-level image. On the left side of Fig. 5, after decoding the QR code, the URL information represented by the QR code is accessed. A new webpage is popped up for representing the URL in the QR code, and information relating to the original image can be provided, or online shopping can be proceeded consequently. The hidden data, or the 145800 bits on the lower-right portion of original image, are recovered first, and are directly put into the lower-right corner. Consequently, the original image can be recovered. For verifying the correctness of our algorithm, we can see that all the mean square errors (MSE's) at the final row of Table II are all 0.00, meaning that the recovered images are identical to their original counterpart. Therefore, we can conclude that our algorithm can be well applicable to grey-scale images.

D. *Application with the Color Images*

In Fig. 6, we demonstrate the applicability of our algorithm to color images. Fig. 6(a) denotes the QR code with the size of

(a)

(b)

Fig. 6. Test materials for color image in this paper. (a) The QR code with size 155×155, containing the URL of http://www.nuk.edu.tw/**. (b) The color image** `rooftop` **with size 2048×1536, containing the QR code at lower-right corner.**

155×155, which contains the URL information of the first author's affiliation (http://www.nuk.edu.tw/). Fig. 6(b) is the color image named `rooftop`, taken by ourselves, with the size of 2048×1536. The QR code in Fig. 6(a) is intentionally placed into the color image in Fig. 6(b).

Figure 7 demonstrates the practical scenario for the application of reversible data hiding to color images. Similar to the corresponding counterparts for grey-scale images in Fig. 5, when the QR code is captured by the mobile phone camera, a new webpage is popped up for representing the URL in the QR code on the left side. Since the color image is represented in RGB format, the hidden data on the lower-right portion of original image, or $155 \times 155 \times 8 \times 3 = 576600$ bits, should be reversibly hidden into the other part of original image. Due to the fact that the filesize of the color image also grows linearly based on the width, the height, and the three color planes, the 576600-bit data can be hidden into the color image since the capacity is abundant.

For the recovery of original image, by use of the proposed algorithm in Sec. III-B, the 576600-bit data can be extracted, and they can be placed into the lower-right corner of recovered image in the right side of Fig. 7. Besides, we also calculate the MSE between the recovered and original images. Again, the MSE is 0.00, meaning that the recovered image is identical to its original counterpart. Therefore, we can conclude that our algorithm can work well for color images.

Fig. 7. After decoding, both the web page corresponding to QR code on the left, and the original color image on the right, can be obtained.

VI. CONCLUSIONS

In this paper, we have described the popularity of the use of QR codes. QR codes can facilitate the access of web pages with mobile phones by capturing the specific corner in the image. We proposed a new algorithm in reversible data hiding, which has the characteristics of abundant capacity for hidden information, and the little amount for side information. In addition, as we can see from practical scenarios, the existence of such a code degrades the quality of the original image or even conceals some information contained in the original image inherently. Considering the facilities offered by the QR codes, users can access the webpage with the QR code first. Next, with the proposed algorithm, the QR code can be removed from the corner of the image, and the original image can be recovered back. The QR code information can be deciphered to some URL relating to the original image, and more information corresponding to the original image can be discovered by the users, such as online shopping. More applications can also be explored in the future.

REFERENCES

[1] J. S. Pan, H. C. Huang, L. C. Jain, and W. C. Fang (editors), *Intelligent Multimedia Data Hiding*, Springer, Berlin-Heidelberg, Germany, Apr. 2007.

[2] H. C. Huang and W. C. Fang, "Metadata-based image watermarking for copyright protection," *Simulation Modelling Practice and Theory*, vol. 18, no. 4, pp. 436-445, Apr. 2010.

[3] J. S. Tan, "QR code," *Synthesis Journal*, Section 3, pp. 59-78, 2008.

[4] H. C. Huang and Y. H. Chen, "Genetic fingerprinting for copyright protection of multicast media," *Soft Computing*, vol. 13, no. 4, pp. 383-391, Feb. 2009.

[5] F. C. Chang, H. C. Huang, and H. M. Hang, "Layered access control schemes on watermarked scalable media," *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology*, vol. 49, no. 3, pp. 443-455, Dec. 2007.

[6] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, Aug. 2003.

[7] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1147-1156, Aug. 2004.

[8] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354-362, March 2006.

[9] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 721-730, April 2007.

[10] H. J. Kim, V. Sachnev, Y. Q. Shi, J. Nam, and H. G. Choo, "A novel difference expansion transform for reversible data embedding," *IEEE Trans. Information Forensics and Security*, vol. 3, no. 3, pp. 456-465, Sep. 2008.

[11] Y. Hu, H. K. Lee, and J. Li, "DE-based reversible data hiding with improved overflow location map," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 250-260, Feb. 2009.

[12] K. Sayood, *Introduction to Data Compression, 3rd Ed.*, Morgan Kaufmann, 2005.

[13] I. S. Reed and G. Solomon, ``Polynomial codes over certain finite fields," *SIAM Journal of Applied Math.*, vol. 8, no. 2, pp. 300-304, Jun. 1960.

[14] QR-Code Generator, http://qrcode.kaywa.com/, 2010.

BIOGRAPHIES

Hsiang-Cheh Huang (S'97-M'01-SM'06) received his Ph.D. from the Department of Electronics Engineering at National Chiao-Tung University in 2001. He is currently an associate professor in the Department of Electrical Engineering, National University of Kaohsiung in Taiwan.

Dr. Huang's research interests include digital watermarking, video compression, and error resilient coding. He has published over 85 international journal and conference papers, 14 book chapters, and 2 Taiwan patents.

Dr. Huang is an IEEE Senior Member. He has served as a TPC member for a variety of international conferences including the International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP, 2005-2011), and the International Conference on Security Technology (SecTech, 2008-2010). He is also an Associate Editor for the International Journal of Innovative Computing, Information and Control (IJICIC) since 2005. For more details, please refer to the website at http://ee.nuk.edu.tw/~hchuang/.

Feng-Cheng Chang (M'06) received the B.S. and M.S. degrees in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1994 and 1996 respectively. He joined the Java center of the Institute for Information Industry (III), Taiwan, in July 1998. After working at III for a year, he came back to NCTU and received the Ph. D. degree in May 2006. Currently he is an assistant professor in the department of innovative information and technology, Tamkang University (TKU), Taiwan. His research interests include image processing, multimedia database, digital rights management, software development methodology, and Internet applications.

Wai-Chi Fang (S'81-M'86-SM'93-F'03) received his B.S. degree from the Electronics Engineering Department at National Chiao Tung University in 1978. He completed his M.S. degree in 1982 from the State University of New York at Stony Brook and his Ph.D. degree in 1992 at the University of Southern California.

Dr. Fang is currently the TSMC Chair Professor and the Director of System-on-Chip Research Center of National Chiao Tung University. From 1985 to 2009, he had been with NASA's Jet Propulsion Laboratory (JPL), California Institute of Technology. He holds thirteen NASA new technologies and seven US patents. He is the recipient of NASA Certificates of Recognition for these creative technical innovations. He won the NASA Space Act award in 2002 and 2003. His inventions on advanced computing engines and data compression systems were used in space missions. His subjects of interest include intelligent green electronic and system-on-chip design for health care applications, VLSI neural networks and intelligent systems, multimedia signal processing and compression, wireless sensor networks, and space integrated avionic systems. He has published over a hundred papers and ten books. He is the recipient of 1995 IEEE VLSI Transactions Best Paper Award.

Dr. Fang is an IEEE Fellow (2003). He serves as the Vice President of IEEE Systems Council and the Chairman of Transnational and Liaison Committee. He was a Board of Governor member of the IEEE Circuits and Systems Society (CASS) (2004-2006 and 2007-2009) and an Administrative Committee member of the IEEE Nanotechnology Council (2005-2006). He was the Chairman of IEEE CASS Technical Committee on Nanoelectronics and Gigascale Systems (2005-2007), Chairman of IEEE CASS Technical Committee on Multimedia Systems and Applications (2003-2005), and Chairman of IEEE CASS Technical Committee on Neural Systems and Applications (2001-2003). He serves on the Advisory Board of IEEE Systems Journal and the Advisory Board of International Journal of Innovative Computing, Information & Control. He served as Associated Editor for IEEE Transactions on Circuits and Systems I (2002-2003), IEEE Transactions on Multimedia (2000-2001), IEEE CASS Circuit and Device Magazine (1999-2000), and the IEEE Transactions on Very Large Scale Systems (1997-1998). He serves on Organization Committee and Technical Program Committee of many international conferences and workshops. He was the general chairman of the 2006 IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing.