Signal Processing: *Image Communication* 41 (2016) 85–100

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Halftone image watermarking via optimization

Yuanfang Guo [a,*], Oscar C. Au [a], Jiantao Zhou [b], Ketan Tang [a], Xiaopeng Fan [a,c]

[a] *Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology, Hong Kong*
[b] *Department of Computer and Information Science, University of Macau, Macau*
[c] *School of Computer Science, Harbin Institute of Technology, China*

ARTICLE INFO

Article history:
Received 23 March 2015
Received in revised form
9 October 2015
Accepted 7 December 2015
Available online 18 December 2015

Keywords:
Halftone image
Error diffusion
Watermarking
Optimization
Halftone image watermarking

ABSTRACT

Although halftone image watermarking technologies have been rapidly developing in the 21st century, the existing techniques lack a theoretical basis. In this paper, we tackle halftone image watermarking problems from a theoretical perspective. First, we propose a general optimization framework for Halftone Visual Watermarking (HVW), which is a certain category of halftone image watermarking techniques. Then two specific HVW problems, Single-sided Embedding Error Diffusion (SEED) and Double-sided Embedding Error Diffusion (DEED) are presented and solved by applying the proposed framework. With SEED and DEED obtained, both the theoretical solutions and experimental results indicate that our previous heuristic methods, Data Hiding by Conjugate Error Diffusion (DHCED) and Data Hiding by Dual Conjugate Error Diffusion (DHDCED), are special cases of SEED and DEED, respectively. We also demonstrate that DEED can achieve outstanding performance compared to DHDCED and other previous methods by selecting different parameters. With this paper, we essentially build a bridge between the theory and practical implementations of HVW problems.

1. Introduction

In the past two decades, multimedia technologies have been progressing rapidly and multimedia content is everywhere in people's daily lives. Currently, the distribution of multimedia content through printed matter, such as newspapers, magazines, books, etc., is still as important as digital distribution. To protect this printed multimedia content from malicious attacks, such as illegal copying, halftone image watermarking technologies have been invented.

Usually, a greyscale printed image, which is formally referred to as a greyscale halftone image, contains only two colors, black and white (1-bit per pixel). When a halftone image is viewed from a certain distance, it still resembles the original greyscale image. A color halftone image works similarly to a greyscale halftone image. The process to produce a halftone image is called the halftoning process. Over the past several decades, many halftoning methods have been developed. These halftoning methods can be categorized into several types of technologies: ordered dithering [1], error diffusion [2–4], dot diffusion [5–8] and direct binary search [9–12].

Due to the 1-bit nature of halftone images, most general watermarking methods, such as Least Significant Bit embedding [13], cannot be directly applied. Specific research has been carried out in the past 20 years for halftone image copyright protections, authentications and steganography. In general, halftone image watermarking methods can be classified into two classes.

The Class 1 methods usually embed a secret binary bitstream into a single halftone image, despite whether

* Corresponding author.
E-mail address: eeandyguo@connect.ust.hk (Y. Guo).

http://dx.doi.org/10.1016/j.image.2015.12.002
0923-5965/

the embedding is performed in spatial domain [14–20] via manipulating the pixel values, or in wavelet-transformed frequency domain via directly change the coefficient values [21] or exploiting embedding by merging the singular values of the selected frequency band of the cover image and the watermark [22]. Later, when the embedded bitstream needs to be extracted, the computer will apply certain algorithm to the stego halftone image (the scanned version or the electrical copy) to extract the secret bitstream.

On the other hand, the Class 2 methods, which employ halftone visual watermarking (HVW) as the representation throughout this paper, has different secret message (to be embedded) and a much simpler decoding mechanism compared to Class 1 methods. During the embedding procedures, HVW usually embed a secret pattern into multiple halftone images (with N stego halftone images obtained from N cover images, $N > 1$). In decoding process, HVW methods possess two common and simple decoding methods. When extraction is needed later and no computer is available, the secret pattern can be directly perceived by the users when the stego halftone images are overlaid (equivalent to binary AND operation). If the users need a better visual quality of the decoded secret pattern, they can apply a binary not-exclusive-or (XNOR) operation to the scanned stego halftone images with a computer.

Since different halftoning techniques have different mechanisms, different HVW methods will be specifically designed accordingly. Among them, HVW methods for error diffused halftone images have been studied by various researchers because error diffusion gives good visual quality, while it remains very simple to implement among all kinds of halftoning techniques. With such advantages, error diffusion and its improved versions have been widely adopted in the printing industry since the 1970s.

The first HVW method for error diffused halftone images [23] was proposed in 2001 to embed a secret pattern into two halftone images generated by using the stochastic property. However, the contrast of the decoded secret pattern was low and the secret pattern was hard to perceive. Later, [24] imposed the conjugate properties of two stego halftone images according to a watermark pattern. The performance was shown to have improved significantly compared to [23] from both theoretical analysis and experimental results. Still, [24] possessed weaknesses, which were the boundary artifacts and low contrast of the decoded secret pattern when the two original images were different. Meanwhile, by shifting the quantization threshold when generating one of two halftone images, [25] gave similar performance and problems compared to [24]. In the following years, [26] adjusted the dynamic range of the original greyscale image and then exploited a pattern look up table and pixel swapping to achieve embedding. Although the detail of the embedded halftone image was preserved, the contrast of the revealed secret pattern was low and the visual quality of the stego halftone images was quite poor compared to other methods. When [27] applied a gradient attack to the stego images generated by [24], the boundaries of the secret pattern appeared in the edge map. Based on this observation, [27] extended the traditional visual cryptography. But the visual quality of the second generated stego image dropped significantly compared to the first generated stego image and the contrast of the revealed secret pattern in [27] was much lower than in [24], which would make it inconvenient for users to distinguish the secret pattern from the background. Ref. [28] relied heavily on the look-up table training results, though it proposed to adaptively shift the quantization threshold, and improved the performance compared to [25]. Concurrently, after analyzing [24,29] proposed to make amendments on both stego halftone images rather than only on the second halftone image. Despite the fact that the performance was improved significantly, there are still some refinements that can be applied to [29]. After then, [30] proposed a simple extended HVW method of [29] from greyscale domain to color domain.

With many error diffusion based HVW methods developed, recently, researchers started to explore more HVW methods for other halftoning mechanisms. Ref. [18] explored to apply the concept of [25] in direct binary search generated halftone images in 2007. In 2011, inspired by the conjugate property in [24], two HVW methods for dot diffused halftone images are proposed in [31]. Later, [31] is extended to embed one binary secret pattern into two color dot diffused halftone images [32]. In the same year, [33] adopts conjugate property and mathematical morphology to create a new HVW method for ordered dithering. However, [33] shows similar problem to [27] with low quality stego halftone images. According to their paper, the revealed secret pattern of [27] is even less distinguishable compared to [27]. This year, [34] designed the first HVW method for a special kind of error diffusion techniques called multiscale error diffusion (MED) by adopting the concept of visual cryptography. Although MED is originated from error diffusion, researchers only maintained the concept of diffusing current error to future pixels and developed a new framework. Thus, previous error diffusion based HVW methods cannot be directly applied to.

Although plenty of HVW methods have been proposed over the years in studies, most of them have been developed from the implementation perspective. Despite the individual deficiencies above, these heuristic HVW methods cannot prove any optimality for their current solutions because of the lack of a theoretical basis. To resolve this, the proposed work tackles the HVW problems from a theoretical perspective. By considering optimization, we propose a general formulation to describe the HVW problems. Then we can essentially employ this formulation as a general optimization framework to describe any HVW problem and proceed to solve it accordingly.

To elaborate how to apply the proposed framework to specific HVW problems, two practical HVW problems are exploited. Then a viable method is designed to solve the two optimization problems, and thus we propose two general HVW methods called Single-sided Embedding Error Diffusion (SEED) and Double-sided Embedding Error Diffusion (DEED). With different setups, SEED and DEED can accommodate different applications. In this paper, we not only show that DHCED and DHDCED are special cases of SEED and DEED respectively in theoretical comparisons, but also provide experimental results to support our

conclusions. In the experiments, we also demonstrate that by tuning the parameters of DEED, it shows superior performances compared to DHDCED and other latest/classical methods.

The rest of the paper is organized as follows. Section 2 reviews the previous methods, DHCED and DHDCED. Section 3 gives the proposed general formulation for HVW problems and proposes SEED and DEED. Section 4 presents the experimental results and discussion, and Section 5 concludes the paper.

2. Reviews of DHCED and DHDCED

In this section, Data Hiding by Conjugate Error Diffusion (DHCED) [24] and Data Hiding by Dual Conjugate Error Diffusion (DHDCED) [29] are reviewed.

First, we define some common notations which will be used throughout this paper. Let X_1 and X_2 be two original grey-scale images, which can be identical or not. Let Y_1 and Y_2 be the generated stego halftone images. Let W be the secret binary pattern to be embedded. Let W_w be the collection of the locations of the white pixels in W and W_b be the collection of the locations of the black pixels in W. Let $\otimes$ represent the binary AND decoding operation. Let $\odot$ be the binary not-exclusive-or (XNOR) decoding operation.

2.1. Data hiding by conjugate error diffusion

DHCED generates two stego halftone images, Y_1 and Y_2, referenced to X_1, X_2 and W such that when Y_1 and Y_2 are overlaid or an XNOR operation is carried out between them, the secret pattern W is revealed. The core idea of DHCED is to let $y_2(i,j)$ be identical or different(conjugate) compared with $y_1(i,j)$ according to the current secret pattern pixel value to be embedded.

In DHCED, the first stego halftone image Y_1 is generated using standard error diffusion, as shown in Fig. 1. The core idea of error diffusion is to use the feedback loop to diffuse the current quantization error to future pixels, such that when the future pixels are processed, they will compensate for the previous quantization errors. The mathematical description of error diffusion is described in (1)–(4).

$$a_1(i,j) = \sum h(k,l) \times e_1(i-k, j-l) \tag{1}$$

$$u_1(i,j) = x_1(i,j) + a_1(i,j) \tag{2}$$

$$y_1(i,j) = \begin{cases} 0, & u_1(i,j) < 128 \\ 255, & u_1(i,j) \geq 128 \end{cases} \tag{3}$$

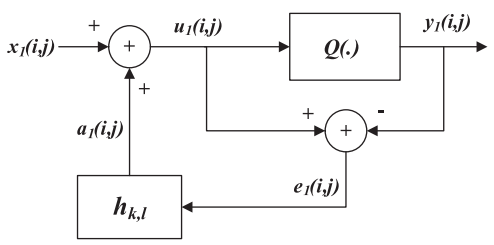

Fig. 1. Error diffusion system.

$$e_1(i,j) = u_1(i,j) - y_1(i,j) \tag{4}$$

where $x_1(i,j)$ is the ij^{th} pixel of X_1, $a_1(i,j)$ is the past error (diffused from past pixels) to be carried by the ij^{th} pixel, $h(k,l)$ is the error diffusion kernel, $e_1(i,j)$ is the quantization error at the ij^{th} pixel, and $y_1(i,j)$ is the ij^{th} pixel of Y_1. Note that error e_1 is defined as the difference between u_1 and y_1, instead of between x_1 and y_1. Two common error diffusion kernels are the Steinberg kernel [2] and Jarvis kernel [3].

After the generation of Y_1, the second halftone image Y_2 is generated by employing the DHCED system with X_2, Y_1 and W as the inputs. The DHCED system is shown in Fig. 2.

In DHCED, for $(i,j) \in W_b$, $y_2(i,j)$ is 'favored' to be conjugate to $y_1(i,j)$. Meanwhile, for $(i,j) \in W_w$, if $X_1 \neq X_2$, then $y_2(i,j)$ is 'favored' to be identical to $y_1(i,j)$. Note that for $(i,j) \in W_w$, if $X_1 = X_2$, then $y_2(i,j)$ is forced to be identical to $y_1(i,j)$. The 'favor' mechanism works as follows. First, the DHCED system performs a trial quantization on the current $u_2(i,j)$. Then if $y_2(i,j) \neq y_1(i,j) \oplus \overline{w(i,j)}$, DHCED will calculate the minimum distortion $\Delta u(i,j)$ to toggle the current pixel. If the minimum distortion is not excessive, i.e. $\Delta u(i,j) \leq T$, then the toggling will be performed. In the 'favor' mechanism, T controls the tradeoff between the visual quality of the stego halftone image and the contrast of the revealed secret pattern. When T increases, the visual quality of Y_2 decreases, while the contrast of the revealed secret pattern increases.

2.2. Data hiding by dual conjugate error diffusion

Although DHCED performs well in many cases, it still has its limitations. In DHCED, we observe that boundary artifacts mainly appear in the flat regions of Y_2 at the bottom and right boundaries of the locations where (i,j) are co-located in W_b when $X_1 = X_2$. If we obtain the edge map of Y_2 by applying the Sobel filter to Y_2, the boundary artifacts will be even more obvious. Also, DHCED does not perform well when X_1 and X_2 are not identical, especially when T is low.

To solve the problems above, DHDCED has been proposed. Still, DHDCED generates two stego halftone images, Y_1 and Y_2, such that when the stego images are overlaid or an XNOR operation is carried out between them, the secret pattern will be revealed. The DHDCED system diagram is shown in Fig. 3.

Unlike DHCED, DHDCED generates Y_1 and Y_2 simultaneously. For $(i,j) \in W_b$, DHDCED will firstly trial quantize

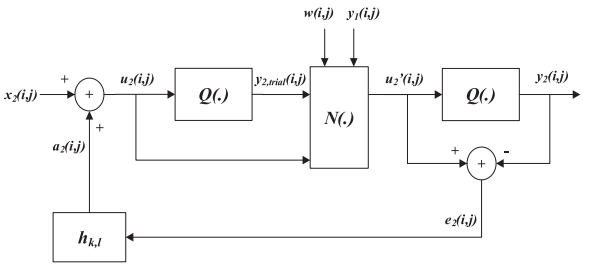

Fig. 2. DHCED system.

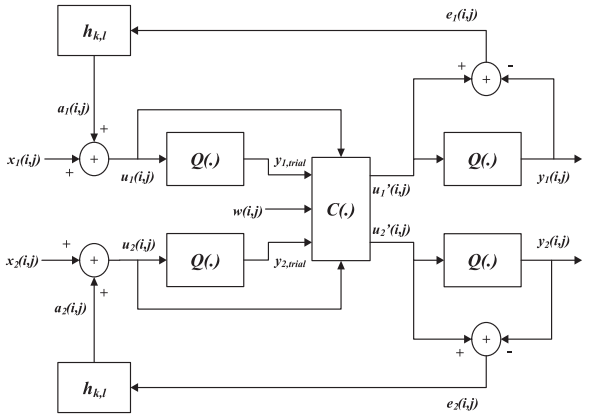

Fig. 3. DHDCED system.

$u_1(i,j)$ and $u_2(i,j)$. Then two strategies are computed accordingly.

Strategy 1: Obtain the minimum distortion $\Delta u_2(i,j)$ when $y_2(i,j)$ is favored to be conjugate to $y_1(i,j)$.
Strategy 2: Obtain the minimum distortion $\Delta u_1(i,j)$ when $y_1(i,j)$ is favored to be conjugate to $y_2(i,j)$.

After the two minimum distortions have been calculated, the strategy which causes smaller distortion will be selected if either of the two distortions is less than or equal to T. If both distortions calculated are larger than T, the toggling will not be performed.

As for $(i,j) \in W_w$, DHDCED performs similar steps to those above except that DHDCED favors $y_1(i,j)$ and $y_2(i,j)$ to be identical. Note that DHDCED forces $y_1(i,j)$ and $y_2(i,j)$ to be identical when $X_1 = X_2$.

3. Proposed work

Halftone visual watermarking tends to hide a secret pattern in halftone images such that when AND or XNOR operation is performed on the stego halftone images, the secret pattern is revealed. During the embedding process, there are distortions caused by modifying the content of the cover images, where the total distortion is represented as D_h. When decoding the secret pattern later, a total distortion D_w can also be measured between the decoded secret pattern and the original secret pattern. D_h and D_w cannot both be small concurrently. Actually, there exists a tradeoff between them. When lower D_h is allowed, D_w tends to be higher; i.e., the decoded pattern tends to be less similar to the original secret pattern. In our formulation, λ will be employed to control the tradeoff. As λ increases, the main focus of our formulation will be shifted from minimizing the embedding distortions while achieving certain decoded visual quality to achieving a high quality decoded secret pattern while considers less about the embedding distortions. Therefore, to find the optimal total distortions under different situations (different λ), a general HVW problem can be formulated as in (5).

$$\min D_h + \lambda \cdot D_w \tag{5}$$

where $\lambda \geq 0$ is a constant which controls the tradeoff between D_h and D_w and is user determined.

Eq. (5) can also be employed as a general optimization framework for HVW problems. With this framework, the existing HVW problems can be reformulated and thus can be solved accordingly. To illustrate how this framework can be applied to specific HVW problems, Eq. (5) will be employed to formulate two HVW problems and then solve them accordingly. The two HVW problems will both generate two error diffused halftone images, Y_1 and Y_2, from two greyscale images, X_1 and X_2. During the generation, a secret pattern W will be embedded into Y_1 and Y_2 such that when the stego images are overlaid or an XNOR operation is carried out between them, the secret pattern is revealed. In these two HVW problems, one only embeds the secret pattern into one out of two cover images, while the other embeds the secret pattern into both cover images. By solving them, we propose two general HVW methods, Single-sided Embedding Error Diffusion (SEED) and Double-sided Embedding Error Diffusion (DEED).

For SEED, Y_1 is obtained by carrying out standard error diffusion on X_1, which means SEED allows no modifications, i.e., zero distortion $D_{Y_1} = 0$ when generating Y_1. Distortion D_{Y_2} is generated while obtaining Y_2 from X_2. To obtain $y_2(i,j)$ while embedding the secret pattern $w(i,j)$, the temporary output value will be compared to $y_1(i,j)$. If the temporary output value is not satisfactory, toggling the final halftone output value is necessary, which can be achieved by adding a distortion $\Delta u(i,j)$ to the intermediate value $u_2(i,j)$ before quantization. Let ΔU be the distortions to be added to each pixel during the embedding process and have the same size as Y_2. Then, adding ΔU during the embedding process is equivalent to adding ΔU onto the original image X_2 first and then performing standard error diffusion on X_2. Thus the total embedding distortion D_h is defined as in (6).

$$D_h = D_{Y_1} + D_{Y_2} = D_{Y_2} = \|\mathbf{X_2} - (\mathbf{X_2} + \mathbf{\Delta U})\|_p^p = \|\mathbf{\Delta U}\|_p^p \tag{6}$$

where $\mathbf{X_2}$ and $\mathbf{\Delta U}$ stand for the vector form of X_2 and ΔU and each of them contains N elements.

Then let $ED(\cdot)$ stand for standard error diffusion, which is described by (1)–(4). Y_1 and Y_2 can be calculated by (7) and (8).

$$\mathbf{Y_1} = ED(\mathbf{X_1}) \tag{7}$$

$$\mathbf{Y_2} = ED(\mathbf{X_2} + \mathbf{\Delta U}) \tag{8}$$

where $\mathbf{X_1}$, $\mathbf{Y_1}$ and $\mathbf{Y_2}$ stand for the vector form of X_1, Y_1 and Y_2 and each of them contains N elements.

After Y_1 and Y_2 are generated, the total decoding distortion D_w can be described as in (9).

$$D_w = \|\mathbf{W} - (\mathbf{Y_1} \circ \mathbf{Y_2})\|_p^p \tag{9}$$

where $\mathbf{W}$ stands for the vector form of W and contains N elements. The symbol '$\circ$' stands for the decoding operation, i.e. $\circ = \otimes, \odot$.

Then substitute (6), (7), (8) and (9) into (5), SEED can be described as in (10).

$$\min_{\mathbf{\Delta U}} \|\mathbf{\Delta U}\|_p^p + \lambda \cdot \|\mathbf{W} - (ED(\mathbf{X_1}) \circ ED(\mathbf{X_2} + \mathbf{\Delta U}))\|_p^p \tag{10}$$

Eq. (10) provides a global optimal solution to SEED. To our best knowledge, there is no closed form solution to (10) due to the feedback loop of the error diffusion and the 1-bit property of halftone images. Also, brute force methods cannot be utilized because the number of potential solutions increases drastically as the image size increases. However, we can obtain an optimal solution to a relaxed problem of (10) as follows.

Since later pixels depend on all the former pixels in standard error diffusion, the distortions to be added to the later pixels will also depend on previous distortions. By replacing the vectors with their elements and also considering the dependencies between each element within every vector, (10) can be reformed into (11).

$$\min_{\Delta u_1, \Delta u_2, \ldots, \Delta u_N} \left\{ \sum_{i=1}^{N} (|\Delta u_i|^p + \lambda \cdot |ed(x_{2,i} + \Delta u_i) \circ ed(x_{1,i}) - w_i|^p) \right\} \tag{11}$$

where $\forall i \in \{2, 3, \ldots, N\}, \Delta u_i$ depends on $\{\Delta u_1, \ldots, \Delta u_{i-1}\}$, and $ed(\cdot)$ stands for carrying out the standard error diffusion described in (1)–(4) on the current pixel and diffusing its quantization error to later pixels according to the processing order of the standard error diffusion.

In general, joint optimization can be reformed to separate optimization if there are no dependencies between them. Because of the processing order of standard error diffusion, previous output values do not depend on later output values. Then if we assume the standard error diffusion processing order is from the first pixel $x_{2,1}$ to the last pixel $x_{2,N}$, by optimizing from Δu_N to Δu_1, we can essentially reform the joint optimization problem (11) into separate optimization problems (12).

$$\min_{\Delta u_1} \left\{ \min_{\Delta u_2} \left\{ \ldots \min_{\Delta u_N} \left\{ \sum_{i=1}^{N} (|\Delta u_i|^p + \lambda \cdot |ed(x_{2,i} + \Delta u_i) \circ ed(x_{1,i}) - w_i|^p) \right\} \ldots \right\} \right\} \tag{12}$$

Let C_i, which is shown in (13), be the cost calculated when processing the pixel i.

$$C_i = |\Delta u_i|^p + \lambda \cdot |ed(x_{2,i} + \Delta u_i) \circ ed(x_{1,i}) - w_i|^p, \quad \forall i \in \{1, 2, 3, \ldots, N\} \tag{13}$$

where $\forall i \in \{2, 3, \ldots, N\}, C_i$ depends on $\{C_1, \ldots, C_{i-1}\}$, since $\forall i \in \{2, 3, \ldots, N\}, \Delta u_i$ depends on $\{\Delta u_1, \ldots, \Delta u_{i-1}\}$.

By substituting (13), (12) can be reformed into (14).

$$\min_{\Delta u_1} \left\{ C_1 + \min_{\Delta u_2} \left\{ C_2 + \ldots + \min_{\Delta u_{N-2}} \left\{ C_{N-2} + \min_{\Delta u_{N-1}} \left\{ C_{N-1} + \min_{\Delta u_N} \{C_N\} \right\} \right\} \ldots \right\} \right\} \tag{14}$$

Since the optimization order is from Δu_N to Δu_1 and the dependencies among the variables cannot be neglected, if the optimization order can be reversed by relaxations, the optimization problem can be solved accordingly.

Since the variable distortion Δu_N depends on every other variable, while all other variables do not depend on Δu_N, then we can relax the joint optimization of Δu_{N-1} and Δu_N to an individual optimization as in (15).

$$\min_{\Delta u_1} \left\{ C_1 + \min_{\Delta u_2} \left\{ C_2 + \cdots + \min_{\Delta u_{N-2}} \left\{ C_{N-2} + \min_{\Delta u_{N-1}} \{C_{N-1}\} + \min_{\Delta u_N} \{C_N\} \right\} \ldots \right\} \right\} \tag{15}$$

Then by carrying out similar relaxation from (14) to (15), (15) can be relaxed to (16).

$$\min_{\Delta u_1} \left\{ C_1 + \min_{\Delta u_2} \left\{ C_2 + \ldots + \min_{\Delta u_{N-3}} \left\{ C_{N-3} + \min_{\Delta u_{N-2}} \{C_{N-2}\} + \min_{\Delta u_{N-1}} \{C_{N-1}\} + \min_{\Delta u_N} \{C_N\} \right\} \ldots \right\} \right\} \tag{16}$$

If we keep carrying out the relaxation, (17) can be obtained in the end.

$$\min_{\Delta u_1} \{C_1\} + \min_{\Delta u_2} \{C_2\} + \cdots + \min_{\Delta u_{N-1}} \{C_{N-1}\} + \min_{\Delta u_N} \{C_N\} \tag{17}$$

Since the processing order is from pixel $x_{2,1}$ to pixel $x_{2,N}$, then (17) can be reformed into (18).

$$\min_{\Delta u_N} \{\ldots \min_{\Delta u_2} \{\min_{\Delta u_1} \{C_1\} + C_2\} + \cdots + C_N\} \tag{18}$$

y substituting (13) into (18), (19) and (20) can be finally obtained. Then the variables can be solved from Δu_1 to Δu_N without being affected by the dependencies among them.

$$J_1 = \min_{\Delta u_1} \{|\Delta u_1|^p + \lambda \cdot |ed(x_{2,1} + \Delta u_1) \circ ed(x_{1,1}) - w_1|^p\} \tag{19}$$

$$J_n = \min_{\Delta u_n} \{|\Delta u_n|^p + \lambda \cdot |ed(x_{2,n} + \Delta u_n) \circ ed(x_{1,n}) - w_n|^p + J_{n-1}\} \tag{20}$$

where $n \in \{2, 3, \ldots, N\}$ and J_1 and J_n are the minimum cost obtained by the optimization process.

After (19) and (20) are obtained, the optimum solution, i.e. SEED, can be calculated easily by obtaining the best Δu_i for each current pixel $x_{2,i}$ with the optimization order from Δu_1 to Δu_N. With ΔU being solved, Y_2 can be easily generated by (8). Before the generation of Y_2, Y_1 is already generated by (7). Thus SEED finishes the generation process of the stego halftone images Y_1 and Y_2.

Algorithm 1. SEED.

Input: λ, error diffusion kernel, $\mathbf{X_1}$, $\mathbf{X_2}$, $\mathbf{W}$.
Output: $\mathbf{Y_1}$, $\mathbf{Y_2}$
1: $\mathbf{Y_1} = ED(\mathbf{X_1})$
2: **for** $i = 1$ to N **do**
3: Obtain $\Delta u_i = \underset{\Delta u_i}{\operatorname{argmin}} C_i$ refer to (13)
4: $y_{2,i} = ed(x_{2,i} + \Delta u_i)$
5: **end for**
6: **return** $\mathbf{Y_1}$, $\mathbf{Y_2}$

As shown in Algorithm 1, in a real implementation of SEED, Y_1 can be directly generated by standard error diffusion. When generating Y_2, SEED will firstly solve the best Δu_i for the current pixel $x_{2,i}$, and then SEED will apply standard error diffusion to $x_{2,i} + \Delta u_i$. In step 3, to solve the best Δu_i for the current pixel $x_{(2,i)}$, searching in a large range is unnecessary. It is easy to eliminate other possible solutions and only maintain two candidate solutions,

which are $\Delta u_i = 0$ and Δu_i (the minimum distortion required to toggle the potential output value $y_{2,i}$), while costs calculated by other solutions will certainly be larger than either of them.

Fig. 4. Test Images: (a) Boat, (b) Lena, (c) Baboon, (d) Elaine, (e) Pepper, (f) Barbara, (g) Bridge, (h) Tank, (i) Cameraman, (j) Couple, (k) Airfield, (l) Clown, (m) Crowd, (n) Dollar, (o) Girlface, (p) Houses, (q) Kiel, (r) Lighthouse, (s) Truck, (t) Trucks, (u) Zelda, (v) Secret pattern to be embedded.

SEED serves as a general method because SEED can accommodate different situations with different p and $\circ$ setups. For example, by comparing the implementation of SEED with that of DHCED described in Section 2.1, with λ performing as T to control the tradeoff between the visual quality of stego images and the contrast of the revealed secret pattern, we can observe that SEED is equivalent to DHCED when $p=1$ and $\circ=\odot$. Thus DHCED is a special case of SEED, and DHCED provides a relaxed optimal solution when $p=1$ and $\circ$. When $p\neq 1$ and $\circ$ varies, SEED becomes new methods.

Similar to SEED, DEED also generates two error diffused halftone images, Y_1 and Y_2, from X_1 and X_2 respectively. However, during the generation, W is embedded into both Y_1 and Y_2 by toggling the output halftone value. The distortion caused during the generation of Y_1 is denoted as D_{Y_1}, and the distortion caused during the generation of Y_2 is denoted as D_{Y_2}. To perform the necessary toggling, a distortion $\Delta u_1(i,j)$ is added to $x_1(i,j)$ and a distortion $\Delta u_2(i,j)$ is added to $x_2(i,j)$ to change the pixel values before quantization. Let ΔU_1 be the distortions to be added to each pixel during the embedding process and have the same size as Y_1. Let ΔU_2 be the distortions to be added to each pixel during the embedding process and have the same size as Y_2. The total distortion D_h for DEED is defined as in (21).

$$D_h = D_{Y_1}+D_{Y_2} = \|\mathbf{X_1}-(\mathbf{X_1}+\mathbf{\Delta U_1})\|_p^p+\|\mathbf{X_2}-(\mathbf{X_2}+\mathbf{\Delta U_2})\|_p^p = \|\mathbf{\Delta U_1}\|_p^p+\|\mathbf{\Delta U_2}\|_p^p \tag{21}$$

where $\mathbf{X_1}$, $\mathbf{X_2}$, $\mathbf{\Delta U_1}$ and $\mathbf{\Delta U_2}$ stand for the vector form of X_1, X_2, ΔU_1 and ΔU_2 and each of them contains N elements.

Still, let $ED(\cdot)$ stand for the standard error diffusion, which is described by (1)–(4). DEED will generate Y_1 and Y_2 by (22) and (23).

$$\mathbf{Y_1} = ED(\mathbf{X_1}+\mathbf{\Delta U_1}) \tag{22}$$

$$\mathbf{Y_2} = ED(\mathbf{X_2}+\mathbf{\Delta U_2}) \tag{23}$$

where $\mathbf{Y_1}$ and $\mathbf{Y_2}$ stand for the vector form of Y_1 and Y_2 and each of them contains N elements.

Since the total distortion D_w between the original secret pattern and the decoded secret pattern possesses an identical definition to SEED, (9) will be directly employed.

Then DEED can be formulated by substituting (21)–(23) and (9) into (5).

$$\min_{\mathbf{\Delta U_1},\mathbf{\Delta U_2}} \|\mathbf{\Delta U_1}\|_p^p+\|\mathbf{\Delta U_2}\|_p^p+\lambda\cdot\|\mathbf{W}-(ED(\mathbf{X_1}+\mathbf{\Delta U_1})\circ ED(\mathbf{X_2}+\mathbf{\Delta U_2}))\|_p^p \tag{24}$$

where $ED(\cdot)$ stands for the standard error diffusion. The symbol '$\circ$' stands for the decoding operation, i.e., $\circ=\otimes,\odot$.

For DEED, if we assume Y_1 and Y_2 are generated simultaneously and the standard error diffusion processing order is from the first pixel pair $(x_{1,1}, x_{2,1})$ to the last pixel pair $(x_{1,N}, x_{2,N})$, (25) and (26) can be obtained by reforming and relaxing (24), with a similar approach to obtaining (19) and (20).

$$J_1 = \min_{\Delta u_{1,1},\Delta u_{2,1}} \{|\Delta u_{1,1}|^p+|\Delta u_{2,1}|^p+\lambda\cdot|ed(x_{2,1}+\Delta u_{2,1})\circ ed(x_{1,1}+\Delta u_{1,1})-w_1|^p\} \tag{25}$$

$$J_n = \min_{\Delta u_{1,n},\Delta u_{2,n}} \{|\Delta u_{1,n}|^p+|\Delta u_{2,n}|^p+\lambda\cdot|ed(x_{2,n}+\Delta u_{2,n})\circ ed(x_{1,n}+\Delta u_{1,n})-w_n|^p+J_{n-1}\} \tag{26}$$

where $n\in\{2,3,\ldots,N\}$ and J_1 and J_n are the minimum cost obtained by the optimization process.

Then DEED can be obtained by calculating the optimum solution of (25) and (26). DEED also selects the best $\Delta u_{1,i}$ and $\Delta u_{2,i}$ for the current pixels $x_{1,i}$ and $x_{2,i}$, with the processing order from $(\Delta u_{1,1}, \Delta u_{2,1})$ to $(\Delta u_{1,N}, \Delta u_{2,N})$. With ΔU_1 and ΔU_2 being solved, Y_1 and Y_2 can be generated by (22) and (23), respectively.

Algorithm 2. DEED.

Input: λ, error diffusion kernel, $\mathbf{X_1}$, $\mathbf{X_2}$, $\mathbf{W}$.
Output: $\mathbf{Y_1}$, $\mathbf{Y_2}$

1: **for** $i=1$ to N **do**
2: Obtain $[\Delta u_{1,i}\ \Delta u_{2,i}] = \underset{\Delta u_{1,i},\Delta u_{2,i}}{\operatorname{argmin}} |\Delta u_{1,i}|+|\Delta u_{2,i}| + \lambda * |ed(x_{2,i}+\Delta u_{2,i})\odot ed(x_{1,i}+\Delta u_{1,i})-w_i|$
3: $y_{1,i} = ed(x_{1,i}+\Delta u_{1,i})$
4: $y_{2,i} = ed(x_{2,i}+\Delta u_{2,i})$
5: **end for**
6: **return** $\mathbf{Y_1}$, $\mathbf{Y_2}$

As shown in Algorithm 2, in practical implementation, DEED firstly solves the best $\Delta u_{1,i}$ and $\Delta u_{2,i}$ for the current pixels, $x_{1,i}$ and $x_{2,i}$. Then standard error diffusion will be performed on $x_{1,i}+\Delta u_{1,i}$ and $x_{2,i}+\Delta u_{2,i}$ separately. For step 2, we can eliminate other possible solutions and maintain only four possible solutions: (a) $\Delta u_{1,i}=\Delta u_{2,i}=0$; (b) $\Delta u_{1,i}=0$, $\Delta u_{2,i}$ being the minimum distortion required to toggle the potential output $y_{2,i}$; (c) $\Delta u_{2,i}=0$, $\Delta u_{1,i}$ being the minimum distortion required to toggle the potential output value $y_{1,i}$; (d) $\Delta u_{1,i}$ being the minimum distortion required to toggle the potential output value $y_{1,i}$ and $\Delta u_{2,i}$ being the minimum distortion required to toggle the potential output value $y_{2,i}$. However, option (d) can be easily proved to have a larger cost than the optimal cost and thus (d) is eliminated. Therefore, there are three possible solutions available.

Similar to SEED, DEED also accommodates different situations with different p and $\circ$ setups. By comparing the implementation of DEED and that of DHDCED described in Section 2.2, with λ performing as T to control the tradeoff between the visual quality of the stego images and the contrast of the revealed secret pattern, we can observe that DEED is equivalent to DHDCED when $p=1$ and $\circ=\odot$. Thus DHDCED is a special case of DEED and DHDCED provides a relaxed optimal solution when $p=1$ and $\circ$. When $p\neq 1$ and $\circ$ varies, DEED becomes new methods.

4. Results and discussion

Firstly, the experimental setups are introduced. The 512×512 test images are shown in Fig. 4. The 512×512 watermark to be embedded is shown in Fig. 4(v). For convenience, the Steinberg kernel [2] and $\circ=\odot$ are

employed in the experiments. (Similar results can be obtained if $\circ = \otimes$.) Note that in a real scenario, SEED and DEED will force $y_1(i,j) = y_2(i,j)$ when $X_1 = X_2$.

The correct decoding rate (CDR) is employed to measure the decoded secret pattern. The CDR is defined as in (27), where we let D be the decoded image and $q = r = 512$ be the test images' size, throughout this paper.

$$\mathrm{CDR} = \frac{\sum |w(i,j) \odot d(i,j)|}{q \times r} \tag{27}$$

4.1. Validation tests

The validations of SEED and DEED are presented first. For SEED, Figs. 5 and 6 show the validations of SEED (L-1 Norm). In Fig. 5, the original grey-scale image is 'pepper', as shown in Fig. 4(e), and $X_1 = X_2$. Y_1 is shown in Fig. 5(a) and Y_2 is shown in Fig. 5(b). The AND decoded image generated from Y_1 and Y_2 is in Fig. 6(a). The XNOR decoded image is in Fig. 6(b).

Figs. 7 and 8 show the validations of SEED (L-2 Norm). In Fig. 7, the original grey-scale image is 'pepper', as shown in Fig. 4(e), and $X_1 = X_2$. Y_1 is shown in Fig. 7(a) and Y_2 is shown in Fig. 7(b). The AND decoded image generated from Y_1 and Y_2 is shown in Fig. 8(a). The XNOR decoded image is shown in Fig. 8(b).

Similarly, for DEED, Figs. 9 and 10 show the validations of DEED (L-1 Norm). In Fig. 9, the original grey-scale image is 'pepper', as shown in Fig. 4(e), and $X_1 = X_2$. Y_1 is shown in Fig. 9(a) and Y_2 is shown in Fig. 9(b). The AND decoded image generated from Y_1 and Y_2 is shown in Fig. 10(a). The XNOR decoded image is shown in Fig. 10(b).

Figs. 11 and 12 show the validations of DEED (L-2 Norm). In Fig. 11, the original grey-scale image is 'pepper',

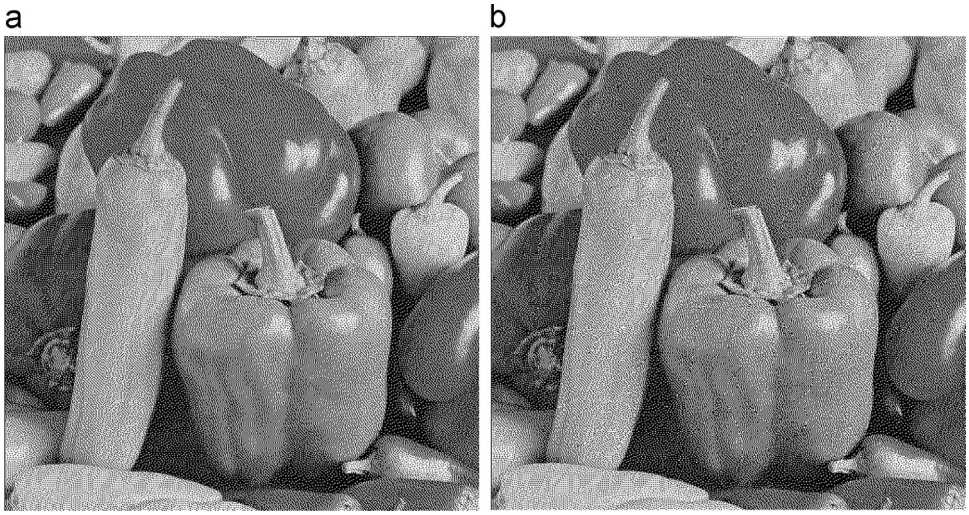

Fig. 5. (a) SEED (L-1 Norm) Y_1, $X_1 = X_2$, Steinberg kernel; (b) SEED (L-1 Norm) Y_2, $X_1 = X_2$, Steinberg kernel.

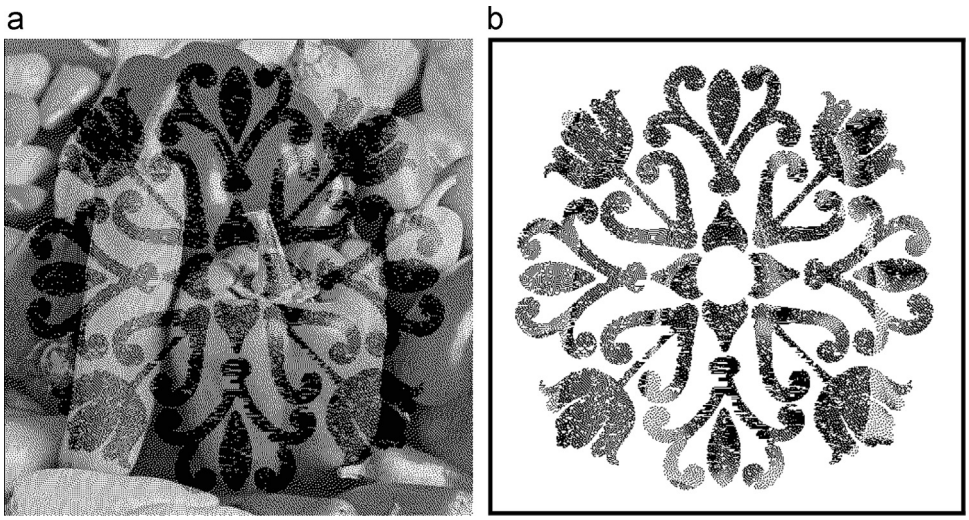

Fig. 6. (a) SEED (L-1 Norm) AND operation decoded image, $X_1 = X_2$, Steinberg kernel; (b) SEED (L-1 Norm) XNOR operation decoded image, $X_1 = X_2$, Steinberg kernel.

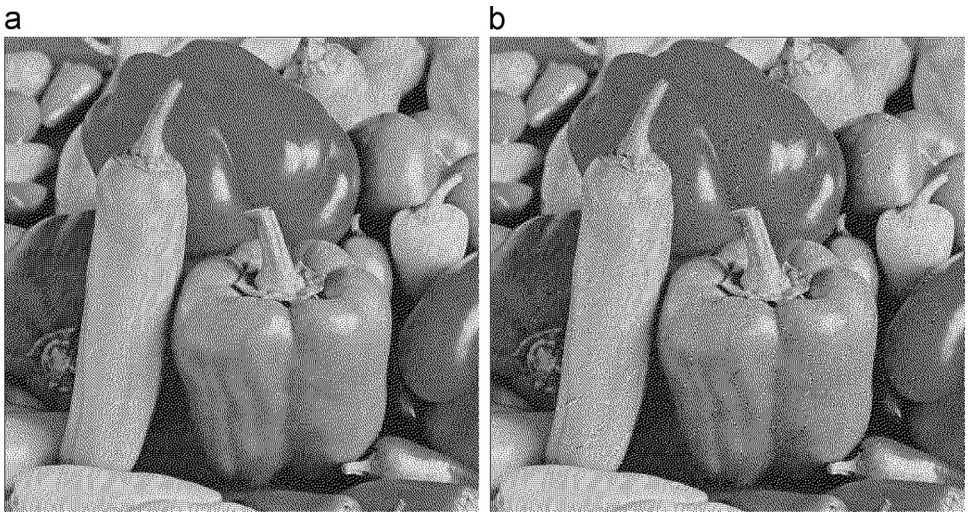

Fig. 7. (a) SEED (L-2 Norm) Y_1, $X_1 = X_2$, Steinberg kernel; (b) SEED (L-2 Norm) Y_2, $X_1 = X_2$, Steinberg kernel.

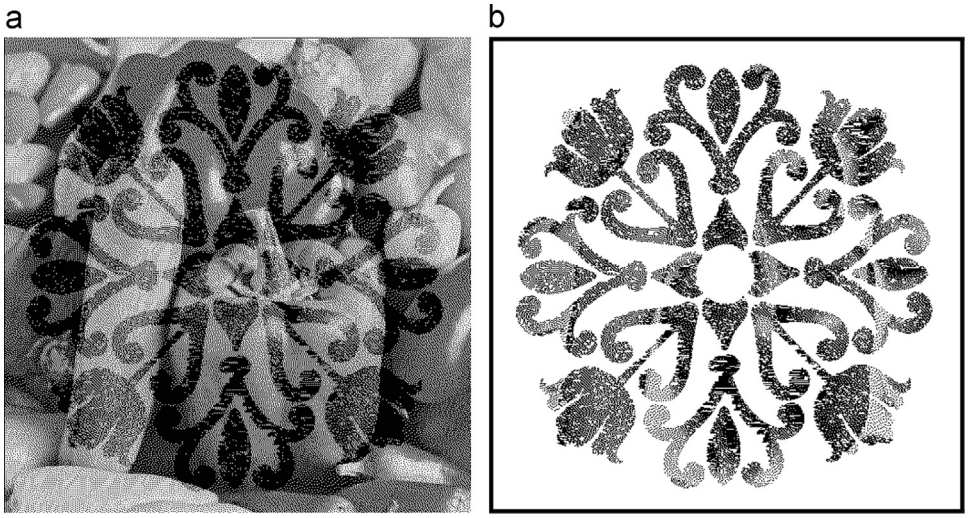

Fig. 8. (a) SEED (L-2 Norm) AND operation decoded image, $X_1 = X_2$, Steinberg kernel; (b) SEED (L-2 Norm) XNOR operation decoded image, $X_1 = X_2$, Steinberg kernel.

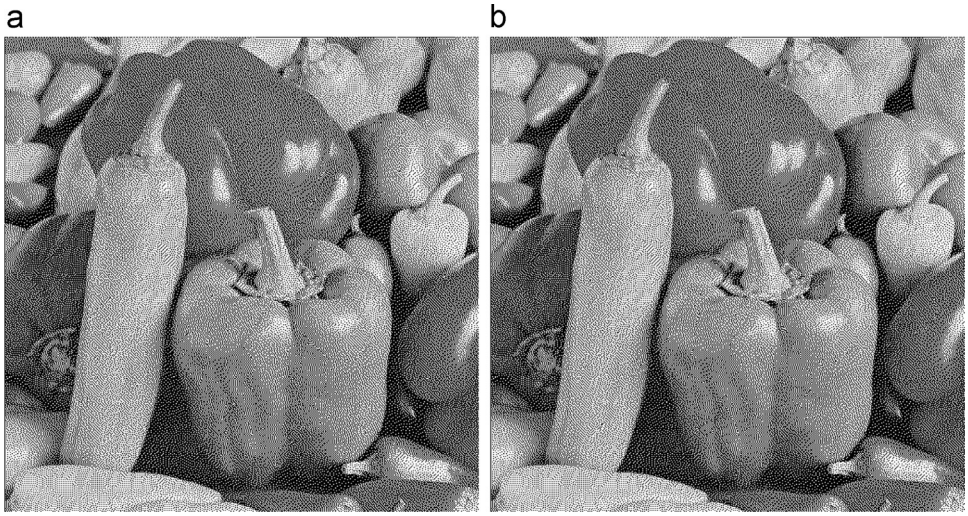

Fig. 9. (a) DEED (L-1 Norm) Y_1, $X_1 = X_2$, Steinberg kernel; (b) DEED (L-1 Norm) Y_2, $X_1 = X_2$, Steinberg kernel.

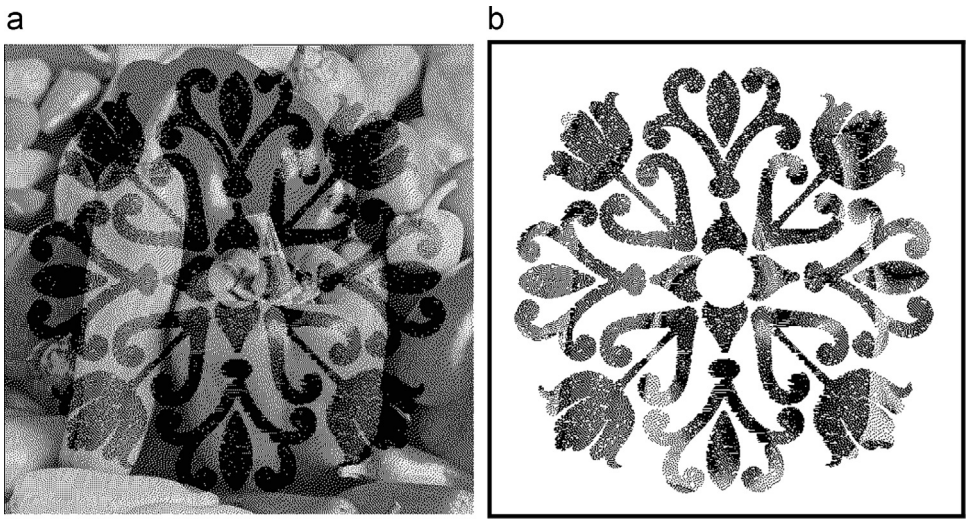

Fig. 10. (a) DEED (L-1 Norm) AND operation decoded image, $X_1 = X_2$, Steinberg kernel; (b) DEED (L-1 Norm) XNOR operation decoded image, $X_1 = X_2$, Steinberg kernel.

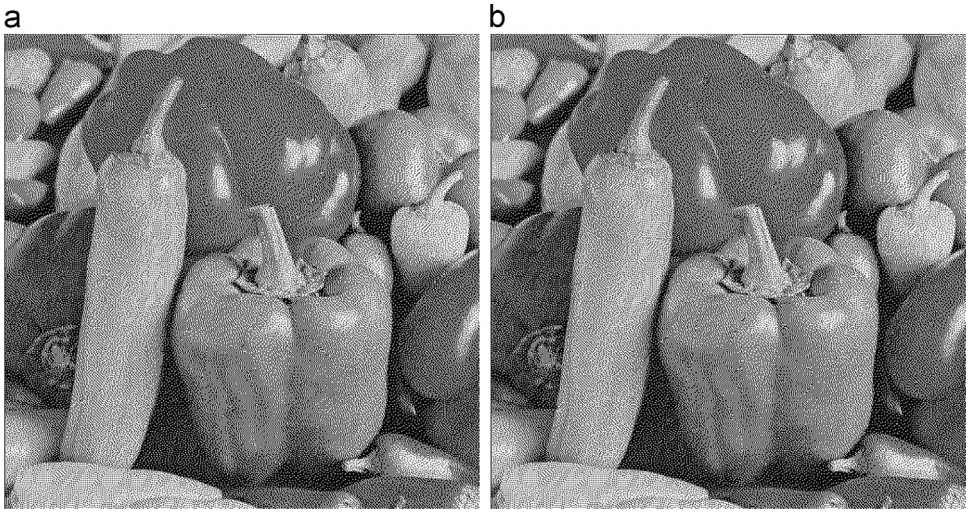

Fig. 11. (a) DEED (L-2 Norm) Y_1, $X_1 = X_2$, Steinberg kernel; (b) DEED (L-2 Norm) Y_2, $X_1 = X_2$, Steinberg kernel.

as shown in Fig. 4(e), and $X_1 = X_2$. Y_1 is shown in Fig. 11 (a) and Y_2 is shown in Fig. 11(b). The AND decoded image generated from Y_1 and Y_2 is shown in Fig. 12(a). The XNOR decoded image is shown in Fig. 12(b).

4.2. Performance tests

In Figs. 5–12, since the validation results are all generated at the same distortion level, we can also observe that DEED outperforms SEED and the L-2 Norm methods outperform the L-1 Norm methods when comparing the contrast of the revealed secret pattern. After this subjective comparison, objective tests are performed.

In objective tests, image quality measurement is very important. Although many researchers prefer to employ modified PSNR as a quality measuring tool, which carries out a low-pass filter on the halftone images first and then calculates the PSNR between the low-passed images and the original images, different low-pass filters will have different effects on different halftone images. Also, due to the imperfection of the halftoning process, distortions during the embedding may not be measured accurately with the modified PSNR because some distortions may actually improve the measured quality of the stego halftone images. In SEED and DEED, since the generation process of Y_1 and Y_2 is equivalent to adding specific bounded noise to the original images and then carries out the standard error diffusion process on the noisy images, we can calculate the PSNR between the noisy images and the original cover images directly. By performing this, the distortions caused during the embedding process can be directly measured without being affected by the imperfection of the error diffusion process and the selections of different low-pass filters.

Figs. 13–16 show partial results of SEED, DEED, DHCED and DHDCED, with $X_1 = X_2 =$ '*lena*' for Fig. 13, $X_1 = X_2 =$

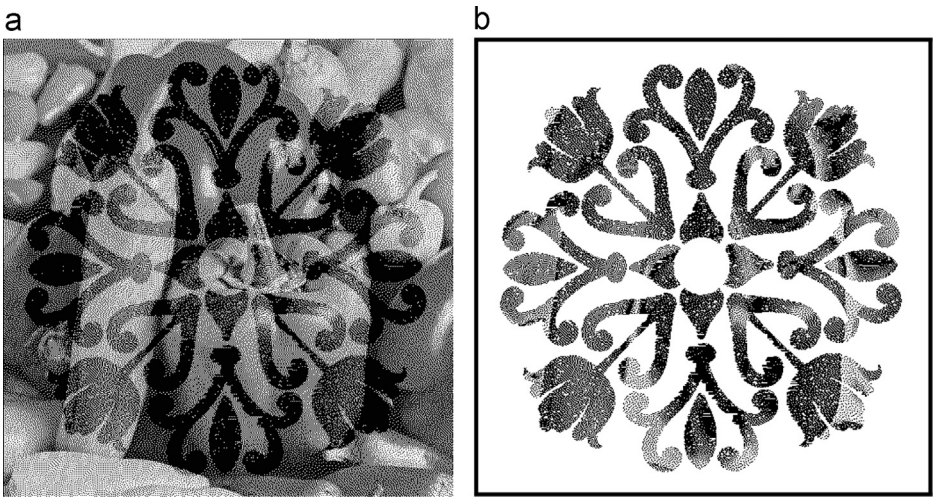

Fig. 12. (a) DEED (L-2 Norm) AND operation decoded image, $X_1 = X_2$, Steinberg kernel; (b) DEED (L-2 Norm) XNOR operation decoded image, $X_1 = X_2$, Steinberg kernel.

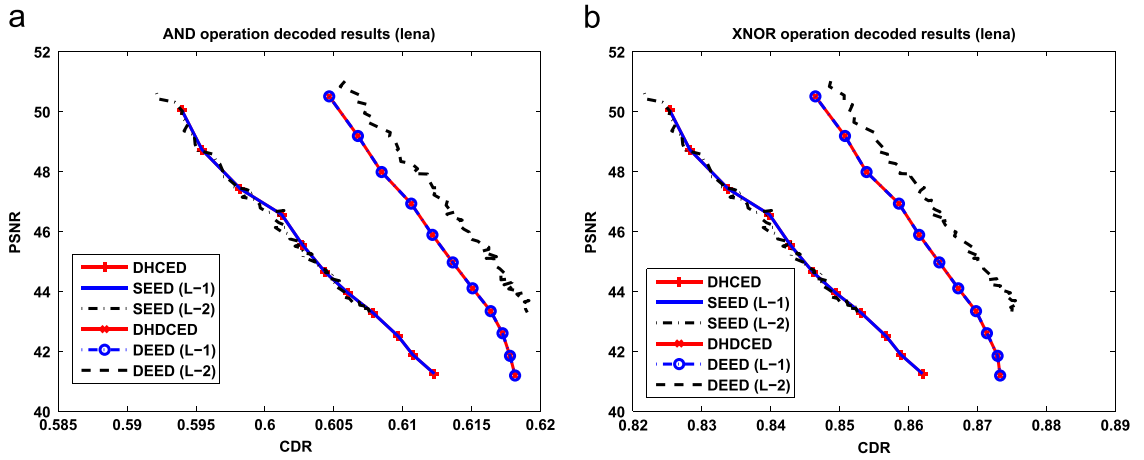

Fig. 13. (a) AND operation decoded PSNR results, $X_1 = X_2 =$ '*lena*', Steinberg kernel; (b) XNOR operation decoded PSNR results, $X_1 = X_2 =$ '*lena*', Steinberg kernel.

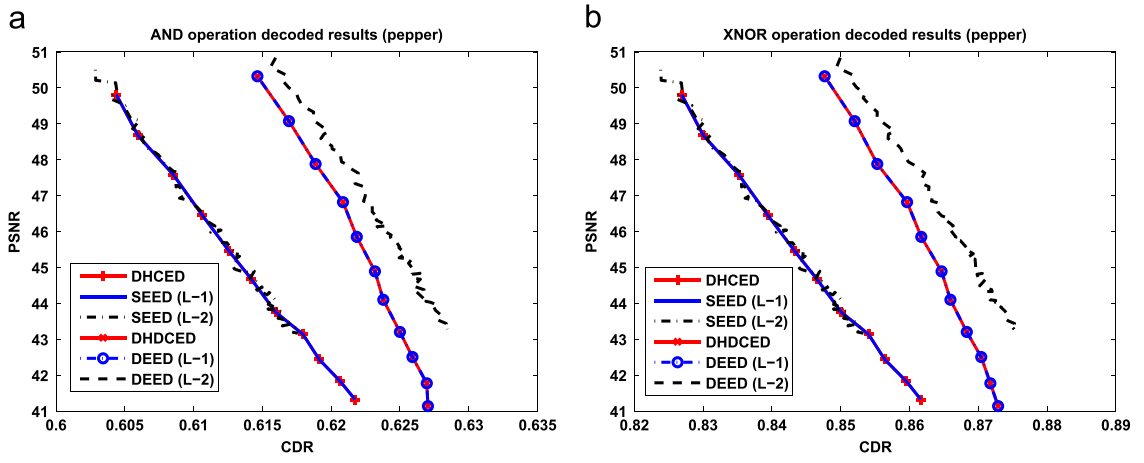

Fig. 14. (a) AND operation decoded PSNR results, $X_1 = X_2 =$ '*pepper*', Steinberg kernel; (b) XNOR operation decoded PSNR results, $X_1 = X_2 =$ '*pepper*', Steinberg kernel.

'*pepper*' for Fig. 14, $X_1 = X_2 =$'*cameraman*' for Fig. 15 and $X_1 = X_2 =$'*houses*' for Fig. 16, by varing λ and T. Note that to make the objective test fair, the PSNRs of DHDCED and DEED shown are average PSNRs between noisy X_1 and noisy X_2. As we can observe from this, SEED (L-1 Norm) presents identical results to DHCED, while DEED (L-1 Norm) presents identical results to DHDCED. SEED (L-2 Norm) has similar performance compared to SEED (L-1 Norm), and DEED (L-2 Norm) demonstrates better performance compared to DEED (L-1 Norm).

Previous experiments indicate that DEED outperforms SEED significantly. Therefore, later results will focus on comparing the L-1 Norm methods with L-2 Norm methods under different circumstances. Here, Table 1 shows the PSNR results of SEED and DEED with $X_1 = X_2$. For convenience, in each pair of comparison, we select the CDR with the corresponding PSNR is about 45 dB for each SEED/DEED with L-1 Norm result as the basis, and then select the SEED/DEED with L-2 Norm result accordingly for fair comparison. Based on the observations, we can conclude that DEED (L-2 Norm) performs significantly better (1.83 dB on average) than DEED (L-1 Norm), while SEED (L-2 Norm) have similar performance compared to SEED (L-1 norm). Note that for convenience, only DEED with L-2 Norm will be employed in the latter experiments of this paper.

4.3. Comparisons to existing work

In the comparison tests, several latest and classical HVW methods, NBED in [25], DHCDD and DHDCDD in [31], DHDCCED in [30], DCCDD in [32] and CoPMED in [34] will be compared to DEED with L-2 Norm. Note that since the methods in [30] and [32] are designed only for color images, they are tested on color versions of Fig. 4, (unfortunately, some color versions of test images are not found,) and the results are averaged over the three channels for comparison.

Since some existing work cannot utilize our previous assessment method, Human-visual Peak Signal-to-Noise

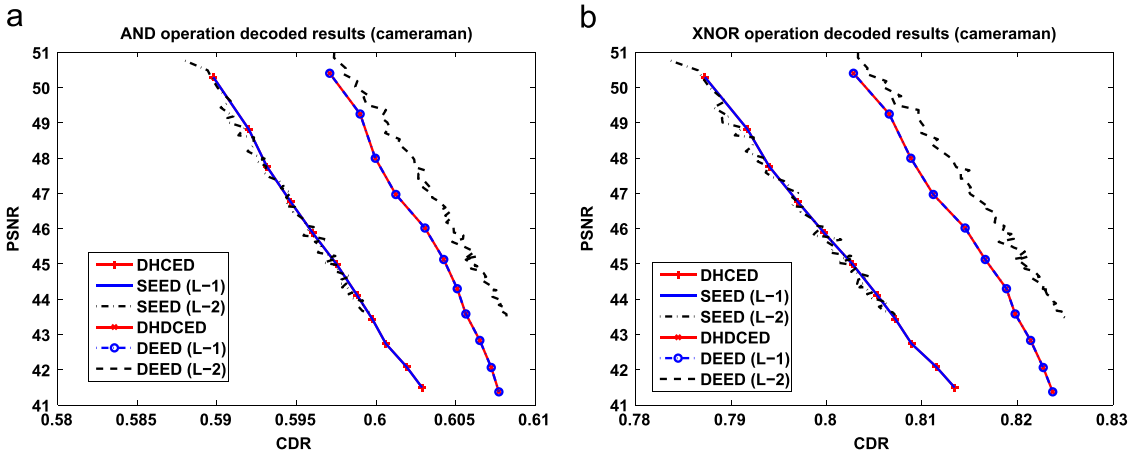

Fig. 15. (a) AND operation decoded PSNR results, $X_1 = X_2 =$'*cameraman*', Steinberg kernel; (b) XNOR operation decoded PSNR results, $X_1 = X_2 =$'*cameraman*', Steinberg kernel.

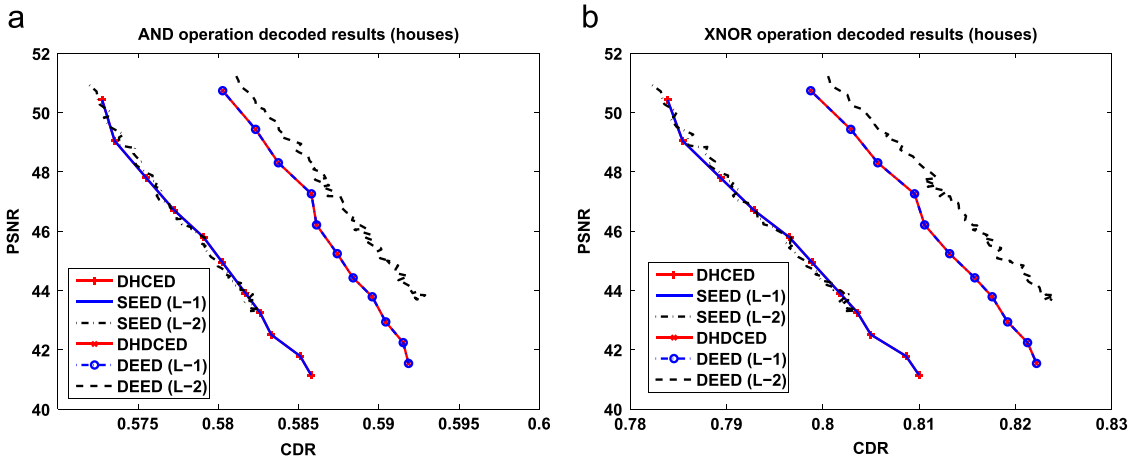

Fig. 16. (a) AND operation decoded PSNR results, $X_1 = X_2 =$'*houses*', Steinberg kernel; (b) XNOR operation decoded PSNR results, $X_1 = X_2 =$'*houses*', Steinberg kernel.

Ratio (HPSNR) [7] will be employed to measure the qualities of stego halftone images. Assume the dynamic range of the original grayscale and the to-be-measured halftone images X and Y are [0, 1], then HPSNR is defined as follows:

$$\mathrm{HPSNR} = 10\log_{10}\frac{q\times r}{\sum_j\sum_j[\sum_{m,n}g(m,n)\times(x(i+m,j+n)-y(i+m,j+n))]^2} \tag{28}$$

where G is a normalized 7∗7 circular symmetrical Gaussian filter with a standard deviation of 1 for simulating the human visual system.

Figs. 17–19 show partial results of the comparison results. In the experiments, different halftoning mechanism based HVW method gives different performance, and DEED(L-2) clearly gives the best performance compared to all these latest and classical methods.

4.4. Discussion of robustness

Previous experiments are conducted in ideal situations. In practice, during the distribution and transmission process, the stego halftone images may suffer various attacks such as noise, print-and-scan distortions, human-marking and cropping.

According to previous documents [24], when suffering cropping or human-marking attacks, the decoded secret pattern will lose the part of the secret pattern which has been cropped or human-marked.

Table 1
PSNR comparisons of the decoded images.

Test image	SEED		DEED	
	L-1	L-2	L-1	L-2
Lena	44.63	44.69	44.96	46.65
Pepper	44.68	44.70	44.89	46.55
Baboon	44.56	44.62	45.25	46.61
Boat	45.38	45.64	45.01	46.68
Elaine	44.95	44.94	45.12	47.15
Tank	44.84	44.80	45.34	47.33
Bridge	44.62	44.71	45.10	46.83
Couple	44.78	44.80	45.19	46.60
Cameraman	44.99	45.23	45.13	46.78
Barbara	44.61	45.02	45.06	46.80
Airfield	44.79	45.04	45.22	47.08
Clown	45.15	45.38	44.70	47.85
Crowd	45.41	45.49	44.97	47.39
Dollar	44.66	44.78	45.13	46.60
Girlface	45.44	45.48	44.79	47.11
Houses	44.95	44.97	45.24	47.35
Kiel	45.10	45.01	44.65	45.82
Lighthouse	44.59	45.26	45.09	47.16
Truck	45.30	45.48	45.42	46.93
Trucks	44.85	44.60	45.20	46.67
Zelda	44.78	44.99	45.08	46.87
Average	44.91	45.03	45.07	46.90

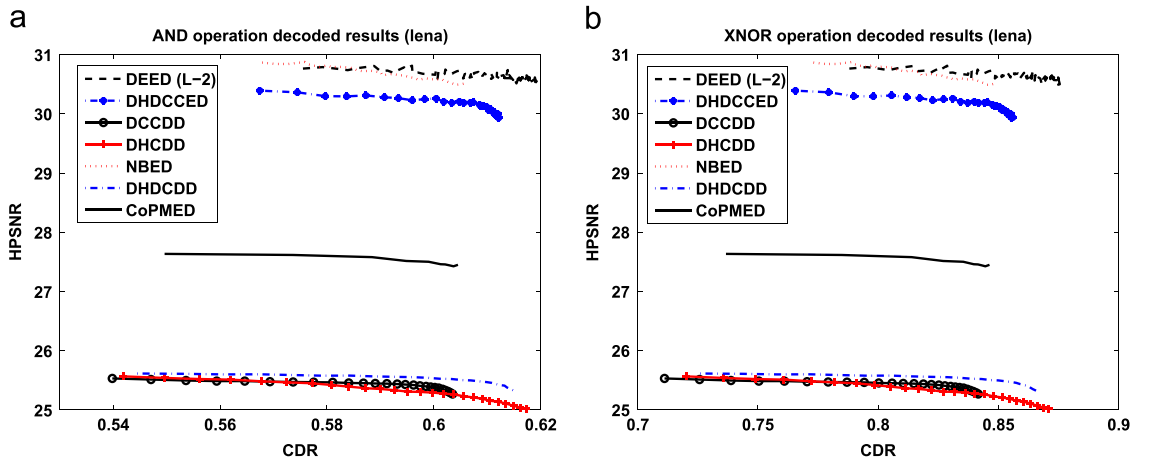

Fig. 17. (a) AND operation decoded HPSNR results, $X_1 = X_2 =$ '*lena*'; (b) XNOR operation decoded HPSNR results, $X_1 = X_2 =$ '*lena*'.

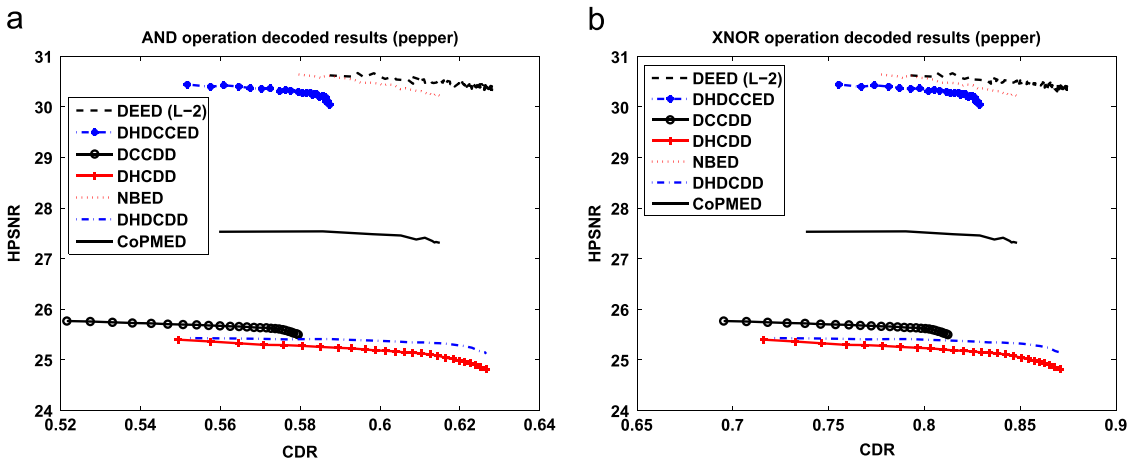

Fig. 18. (a) AND operation decoded HPSNR results, $X_1 = X_2 =$ '*pepper*'; (b) XNOR operation decoded HPSNR results, $X_1 = X_2 =$ '*pepper*'.

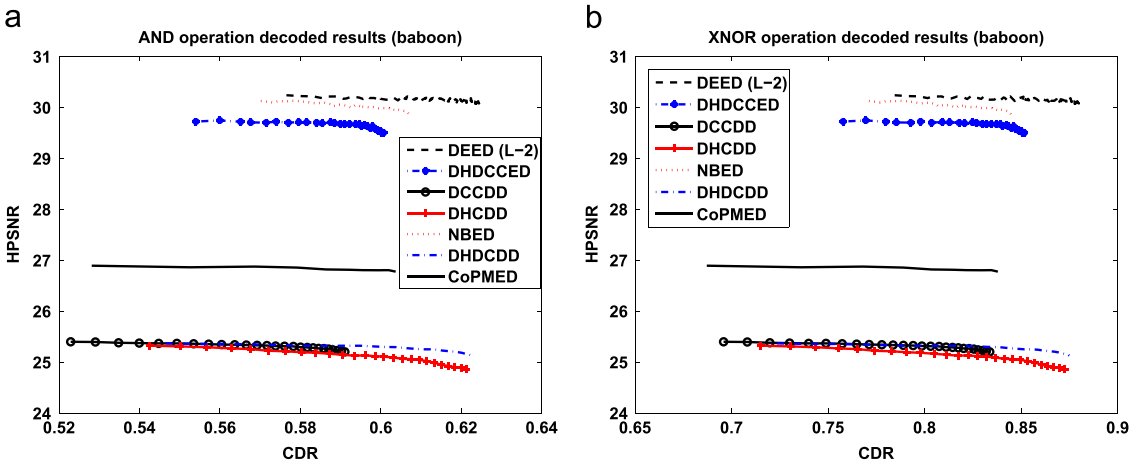

Fig. 19. (a) AND operation decoded HPSNR results, $X_1 = X_2 =$ '*baboon*'; (b) XNOR operation decoded HPSNR results, $X_1 = X_2 =$ '*baboon*'.

According to previous documents [18] and [28], usually, HVW methods will suffer print distortions when performing the overlaying operation for decoding. If the users employ the XNOR decoding procedure, HVW methods will suffer the print-and-scan distortions. Figs. 20 and 21 show a real example, where DEED(L-2) is employed to embed the secret pattern in Fig. 4(v) into two halftone images in Fig. 20.

In the experiment, Fig. 20(a) is printed on a regular paper while Fig. 20(b) is printed on a transparent plastic. As Fig. 21(a) shows, when DEED(L-2) suffers the print distortions, the overlaying revealed secret pattern maintains a good visual quality. When suffering print-and-scan distortions or noise, the final decoded secret pattern can still maintain a distinguishable visual quality if the scanning resolution is high enough and certain preprocessing steps, such as resizing, contrast adjustment and 1-bit quantization, are carried out before the decoding process, as Fig. 21(b) shows. Based on our observation in the experiments, the scanning resolution should be over 600 dpi to obtain a distinguishable decoded result. Also, people should try to avoid embedding information into very dark and very bright areas since the information will be largely damaged after print-and-scan attack.

4.5. Discussion of complexity

After evaluating the performances of proposed work, here we will briefly discuss the complexities of SEED and DEED.

Although optimization is exploited in SEED/DEED, solving SEED/DEED is actually simple as stated in 3. In real implementations, solving SEED only requires $(p-1)$ extra multiplications per pixel compared to DHCED while solving DEED only requires $2*(p-1)$ extra multiplications per pixel compared to DHDCED, where p is the L−p Norm employed.

Table 2 summarizes operations expected for processing one pixel in SEED, DHCED, NBED, DHCDD and CoPMED, and operations expected for processing one pair of pixels in DEED, DHDCED, DHDCDD, DHDCCED, DCCDD. Here we assume that the probability of changing the current output

Fig. 20. (a) The to be printed DEED (L-2 Norm) Y_1, $X_1 = X_2$, Steinberg kernel; (b) The to be printed DEED (L-2 Norm) Y_2, $X_1 = X_2$, Steinberg kernel.

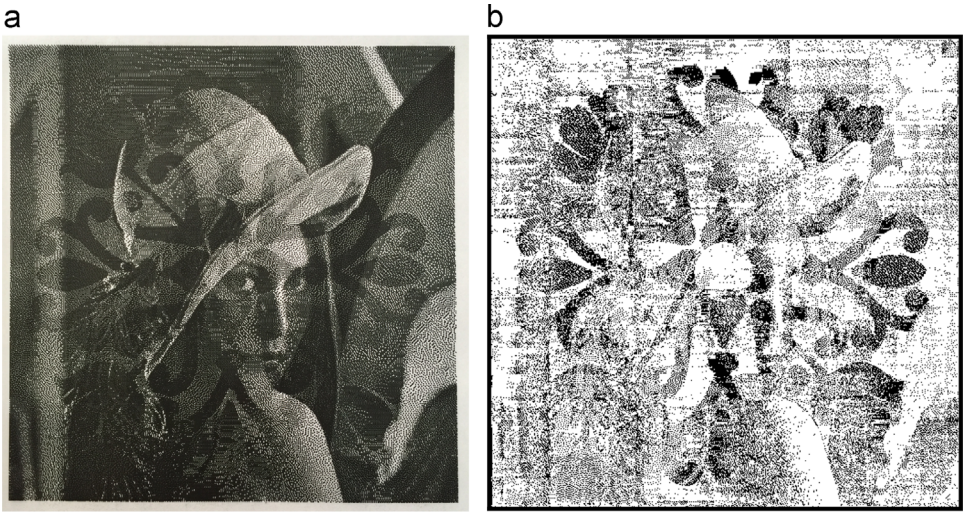

Fig. 21. (a) After print, DEED (L-2 Norm) overlaying decoded image, $X_1 = X_2$, Steinberg kernel; (b) After print and scan, DEED (L-2 Norm) XNOR operation decoded image, $X_1 = X_2$, Steinberg kernel.

Table 2
Operations expected to process one pixel (one pair of pixels).

Method	Addition	Multiplication	Comparison
SEED	2.5	$h+(p-1)$	3
DHCED	2.5	h	3
NBED	3	h	2
DHCDD	2.5	dm	3
CoPMED	$\sum_{n=1}^{6}\{2^{2n+6}\}+40$	0	37.375
DEED	6	$2h+2*(p-1)$	5
DHDCED	6	$2h$	5
DHDCDD	6	2dm	5
DHDCCED	18	$6h$	15
DCCDD	18	6dm	15

halftone value is 0.5 when assessing. The variable h and dm stand for the error kernel and diffusion matrix, respectively.

As we can observe, CoPMED is the most complicated method. SEED possesses similar complexity to DHCED, NBED and DHCDD while DEED possesses similar complexity to DHDCED and DHDCDD.

5. Conclusion

In this paper, we propose a general formulation for HVW problems with Optimization where the formulation can also be viewed as a general optimization framework. We also demonstrate how to apply the general framework to practical problems by presenting two general methods called Single-sided Embedding Error Diffusion (SEED) and Double-sided Embedding Error Diffusion (DEED), which can accommodate multiple situations with different set-ups. Although the global optimal solutions cannot be achieved, we manage to relax the problems and find two

viable relaxed optimal solutions. With SEED and DEED obtained, from both theoretical and experimental comparison, DHCED and DHDCED are found to be special cases of SEED and DEED, respectively. The experimental results also indicate that DEED performs significantly better than DHDCED when certain parameters are selected, while different settings of SEED give similar performance compared to DHCED. When comparing to several latest and classical methods, DEED(L-2) also gives the best performance.

References

[1] B. Bayers, An optimum method for two level rendition of continuous tone pictures, In: Proceedings of IEEE International Conference on Communications, 1973, pp. 2611–2615.

[2] R. Floyd, L. Steinberg, An adaptive algorithm for spatial grayscale, In: Proceedings of Society of Information Display, 1976, pp. 75–77.

[3] J. Jarvis, C. Judice, W. Ninke, A survey of techniques for the display of continuous tone pictures on bilevel displays, Comput. Graph. Image Process. 5 (1976) 13–40.

[4] X. Li, Edge-directed error diffusion halftoning, IEEE Signal Process. Lett. 13 (11) (2006) 688–690.

[5] D. Knuth, Digital halftones by dot diffusion, ACM Trans. Graph. 6 (1987) 245–273.

[6] M. Mese, P. Vaidyanathan, Optimized halftoning using dot diffusion and methods for inverse halftoning, IEEE Trans. Image Process. 9 (4) (2000) 691–709.

[7] J. Guo, Y. Liu, Improved dot diffusion by diffused matrix and class matrix co-optimization, IEEE Trans. Image Process. 18 (8) (2009) 1804–1816.

[8] Y. Liu, J. Guo, New class tiling design for dot-diffused halftoning, IEEE Trans. Image Process. 22 (3) (2013) 1199–1208.

[9] S. Kim, J. Allebach, Impact of hvs models on model-based halftoning, IEEE Trans. Image Process. 11 (3) (2002) 258–269.

[10] J. Guo, Y. Liu, J. Chang, High efficient direct binary search using multiple lookup tables, In: Proceedings of IEEE International Conference on Image Processing, 2012, pp. 813–816.

[11] P. Goyal, M. Gupta, C. Staelin, M. Fischer, O. Shacham, J. Allebach, Clustered-dot halftoning with direct binary search, IEEE Trans. Image Process. 22 (2) (2013) 473–487.

[12] J. Guo, Y. Liu, J. Chang, J. Lee, Efficient halftoning based on multiple look-up tables, IEEE Trans. Image Process. 22 (11) (2013) 4522–4531.

[13] P. Moulin, R. Koetter, Data-hiding codes, IEEE Proc. 93 (12) (2005) 2083–2126.

[14] M. Fu, O. Au, Data hiding by smart pair toggling for halftone images, In: Proceedings of. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 6, 2000, pp. 2318–2321.

[15] M. Fu, O. Au, Halftone image data hiding with intensity selection and connection selection, Signal Process. Image Commun. 16 (10) (2001) 909–930.

[16] M. Fu, O. Au, Data hiding watermarking for halftone images, IEEE Trans. Image Process. 11 (2002) 477–484.

[17] S. Pei, J. Guo, High-capacity data hiding in halftone images using minimal-error bit searching and least-mean square filter, IEEE Trans. Image Process. 15 (6) (2006) 1665–1679.

[18] J. Guo, A new model-based digital halftoning and data hiding designed with lms optimization, IEEE Trans. Multimed. 9 (4) (2007) 687–700.

[19] O. Bulan, G. Sharma, V. Monga, Orientation modulation for data hiding in clustered-dot halftone prints, IEEE Trans. Image Process. 19 (8) (2010) 2070–2084.

[20] J. Guo, Y. Liu, Halftone-image security improving using overall minimal-error searching, IEEE Trans. Image Process. 20 (10) (2011) 2800–2812.

[21] H. Luo, Z. Zhao, J. Huang, Z. Lu, Halftone image watermarking based on the binary pseudo-wavelet transform, In: Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing, vol. 1, 2007, pp. 299–302.

[22] M. Amini, K. Yaghmaie, H. Sadreazami, Error diffusion halftone image watermarking based on svd-dwt technique, In: Proceedings of Iranian Machine Vision and Image Processing, 2010, pp. 1–4.

[23] M. Fu, O. Au, Data hiding in halftone images by stochastic error diffusion, In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, 2001, pp. 1965–1968.

[24] M. Fu, O. Au, Steganography in halftone images: conjugate error diffusion, Signal Process. 83 (2003) 2171–2178.

[25] S. Pei, J. Guo, Data hiding in halftone images with noise-balanced error diffusion, IEEE Signal Process. Lett. 10 (2003) 349–351.

[26] C. Chang, C. Chan, W. Tai, Hiding a halftone secret image in two camouflaged halftone images, Pattern Recognit. Image Anal. 16 (2006) 486–496.

[27] C. Yang, Y. Yang, T. Chen, G. Ye, New steganography scheme in halftone images, In: Proceedings of IEEE International Conference on Intelligent Information on Hiding and Multimedia Signal Processing, 2008, pp. 1520–1523.

[28] J. Guo, J. Tsai, Data-hiding in halftone images using adaptive noise-balanced error diffusion, IEEE MultiMed. 18 (2) (2011) 48–59.

[29] Y. Guo, O. Au, L. Fang, K. Tang, Data hiding in halftone images by dual conjugate error diffusion, In: Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, 2011.

[30] Y. Guo, O. Au, K. Tang, J. Pang, W. Sun, L. Xu, J. Li, X. Zhang, Data hiding in error diffused color halftone images, In: Proceedings of IEEE International Symposium on Circuits and Systems, 2013, pp. 2996–2999.

[31] Y. Guo, O. Au, K. Tang, L. Fang, Z. Yu, Data hiding in dot diffused halftone images, In: Proceedings of IEEE International Conference on Multimedia and Expo, 2011, pp. 1–6.

[32] Y. Guo, O. Au, K. Tang, J. Pang, Hiding a Secret Pattern into Color Halftone Images, In Digital-Forensics and Watermarking, Springer, Berlin, Heidelberg, 2014, 465–474.

[33] S. Sanchev, A. Nayak, T. Pradhan, Data hiding in halftone images using mathematical morphology and conjugate ordered dithering, In: Proceedings of International Conference on High Performance Computing and Applications, 2014, pp. 1–7.

[34] Y. Guo, O. Au, K. Tang, Watermark embedding for multiscale error diffused halftone images by adopting visual cryptography, Int. J. Digital Crime Forensics 7 (1) (2015) 51–68.