Multimed Tools Appl (2018) 77:12111–12138
DOI 10.1007/s11042-017-4861-0

Visual secret sharing in halftone images by multi-scale error diffusion

Po-Chyi Su[1] · **Tzung-Fu Tsai**[1] · **Yu-Chien Chien**[1]

Received: 21 October 2016 / Revised: 19 April 2017 / Accepted: 22 May 2017 /
Published online: 30 May 2017

Abstract A secret sharing scheme in halftone images is proposed in this research. Given a secret halftone image, several gray-level images with the same resolution will be chosen from the database to collaboratively carry the secret information. The selected images will be transferred to halftone ones using Multi-scale Error Diffusion (MED), with the constraint imposed by the pixels of halftone secret image. The modified MED ensures that the resultant pixels of host halftone images should satisfy the required conditions such that collecting all the processed halftone images can successfully reveal the secret one. In addition to facilitating the perfect extraction of hidden information, maintaining the quality of all the halftone images in this secret sharing scenario is the other major objective, which is achieved by the usage of MED and selection of suitable host images. The experimental results demonstrate the satisfactory performance of the proposed method.

Keywords Secret sharing · Halftone · Multi-scale error diffusion · Information hiding

1 Introduction

Secret sharing is a practice of distributing a message among a group of participants, each of whom is allocated a share. The message can only be reconstructed by combing a sufficient number of shares while individual shares are of no use. This idea was invented by Adi Shamir [1] and George Blakley [5] independently in 1979 and is considered a feasible mechanism when storing highly sensitive information in multiple locations for greater

This research is supported by the Ministry of Science and Technology in Taiwan, under Grants MOST104-2221-E-008-072 and MOST105-2221-E-008-086

✉ Po-Chyi Su
pochyisu@csie.ncu.edu.tw

[1] Department of Computer Science and Information Engineering, National Central University, Jhongli District, Taoyuan, 32001, Taiwan

reliability, rather than keeping it in one position with maximum secrecy. Secret-sharing schemes are important tools in cryptography and used as a building box in many secure protocols for multiparty computation, threshold cryptography, access control, attribute-based encryption, and generalized oblivious transfer [4]. One type of secret sharing is visual cryptography [20], in which the message is encoded as an image and can be revealed by human visual system. In a k-out-of-n visual cryptographic scheme, the shares composed of random pixels are given to n participants. Any k or more participants can visually discover the secret message by stacking these shares together. A simplest version of visual secret sharing assumes that the message consists of a collection of black/white pixels. Figure 1 illustrates a 2-out-of-2 visual cryptographic scheme [9]. A secret binary pixel is encrypted into two (n) blocks, A and B, with $m_1 \times m_2 = 2 \times 2$ binary pixels as an example. The share blocks are randomly generated through the column permutation of the six $m_1 \times m_2$ basis matrices as shown in Fig. 1. By repeating the process for all the pixels of a $K_1 \times K_2$ secret (input) image, the visual cryptographic scheme produces two (n) binary shares with dimensions of $m_1K_1 \times m_2K_2$ pixels. Each noise-like binary share is distributed to one of two (n) participants. The secret image is visually revealed only if two (k) recipients stack their shares. The stacked four black pixels will be visually decoded as a black pixel. The $m_1 \times m_2$ stacked block consisting of two black and two white pixels will be decoded as a white pixel, when viewed from a distance.

The advantage of such a traditional visual cryptography scheme is that the recognition of secret message from overlapping shares doesn't require additional computations or any knowledge of cryptographic keys. Quite a few different schemes were thus proposed [31]. Nevertheless, there may exist two problems in traditional visual cryptography; the first problem is the risk of transmission since these meaningless share images appearing in a communication channel will raise suspicion or interest of deciphering by potential eavesdroppers. The second problem is the difficulty of managing shares. The noise-like images may not be differentiated easily by the human eyes so extra care has to be taken to ensure that correct users are given the corresponding shares. Visual secret sharing using meaningful images thus draws research attention. Each share is a viewable image so the transmission of them looks like a regular process and the problem of confusing the recipients of certain shares can also be avoided. Ateniese et al. [3] proposed an extended visual cryptography scheme ($EVCS$) based on hypergraph colourings. Some images are encoded to generate

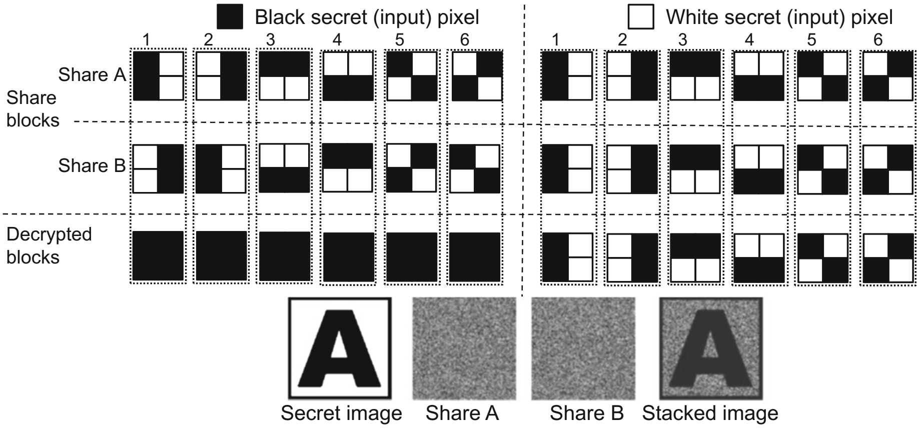


Fig. 1 Visual cryptography concept demonstrated using a (2,2)-threshold scheme [9]

shares that still contain meaningful contents. $EVCS$ has an interesting property that stacking together the shares associated to certain sets can reveal the secret message without traces of the original images. Figure 2 shows one example, in which the secret image S is carried by the shares A, B and C. The shares in this example are constructed in such a way that stacking A, B and C can reveal the letter S while stacking only A and C will not.

Nevertheless, the hypergraph colourings in [3] are still constructed by randomly distributed pixels so the resultant binary shares contain strong white noises, which consequently lead to poorer results, and suffer from loss in contrast. Using more natural images to serve as shares would be a better idea and halftoning [26] is a pretty good choice. Halftoning is one of the earliest image processing techniques and still used nowadays in printing and displaying processes. Although represented with only two colors, halftone images seem to contain various shades of gray when viewed from a distance due to the low-pass characteristics of the human visual system. Several image halftoning methodologies exist, such as ordered dithering [27], dot diffusion [17], error diffusion [15], multi-scale error diffusion (MED) [16] and direct binary search [2], etc. Each method has unique features or advantages and can be applied in different situations. The intriguing appearance of dot patterns makes halftone images suitable to such applications of visual cryptography [34]. An example of a halftone visual secret sharing scheme based on error diffusion by Wang et al. [29] is shown in Fig. 3. Three 512 × 512 images, “Baboon”, “Man” and “Tank”, are used to carry a logo with the original size 128 × 128. By stacking the three images using “OR” operation with black being Boolean “1”, the secret logo can be revealed.

It is worth noting that, to carry the secret message in halftone visual cryptography, flipping black/white pixels to generate halftone shares is necessary so the resultant shares are more or less different from the images processed by the halftoning process only. From this point of view, visual secret sharing in halftone images is closely related to information hiding, in which the data values of host media are modified in an imperceptible way to insert additional information. Such applications as covert communication, copyright protection, ownership declaration and content authentication, etc. can be achieved by information hiding techniques. Secret sharing in halftone images can be viewed one type of covert communication, which conveys messages on a seemingly innocuous host. Quite a

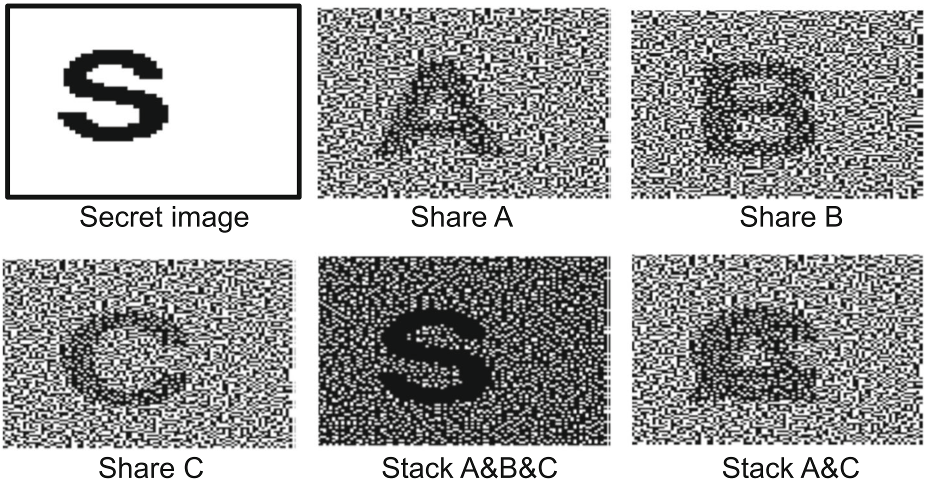


Fig. 2 An example of extended visual cryptography [3]

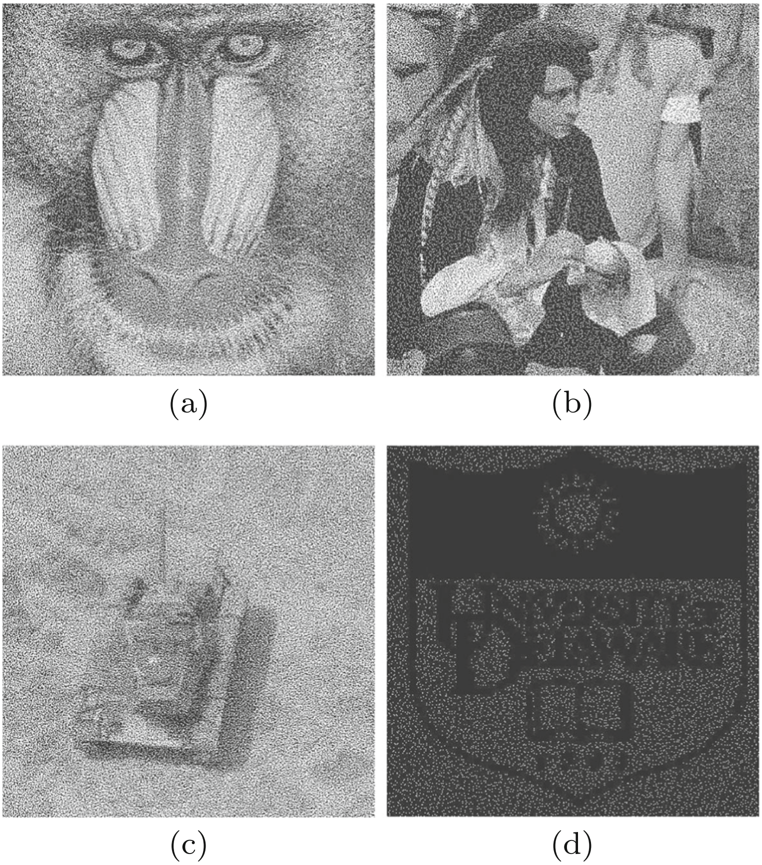

(a) (b) (c) (d)

Fig. 3 An illustrated example of halftone visual cryptography [29] with the three shares showing (**a**)"Baboon", (**b**) "Man" and (**c**) "Tank". **d** is the revealed logo by stacking the three halftone shares

few visual-cryptography-based information hiding schemes have been proposed [6]. In such schemes, besides the related requirements of secret sharing, the quality of distributed images has to be well maintained such that the functions of these images will not be seriously affected. In addition, the hidden information is usually required to be perfectly reconstructed in the extraction phase.

This research investigates effective visual secret sharing in halftone images. Maintaining the quality of shares and ensuring error-free transmission of hidden information are the major research issues. Instead of pursuing recognition of secret message without any additional computations or knowledge of cryptographic keys like such traditional visual sharing schemes as [29], the proposed scheme requires minimum additional operations of parity-check (by examining the number of white points at a position when stacking) and scrambling/descrambling of secret information. These extra steps certainly take more operations but can make the scheme more secure. A major advantage is that the hidden message can be another halftone image have the same resolution with the shares and will be perfectly decoded/revealed. To be more specific, several gray-level images are selected from an image database and transferred to halftone ones acting as camouflages or carriers for transmitting another secret halftone image. All the images participating in the process of secret sharing will have the same resolution. Each processed host image is given to a different user so the secret message is shared by a group of users. The hidden image can only be extracted perfectly after all the host images are available. Fewer shares should not reveal any secret information. MED is employed as the underlying halftoning mechanism since it possesses several interesting characteristics, which are appropriate to the design of visual secret sharing and make the quality of shares well maintained. The rest of the paper is organized as follows. Section 2 reviews the related work, including some information hiding schemes in halftone images and the process of MED. Section 3 describes the proposed scheme in detail. The experimental results in Section 4 will show the feasibility of the proposed scheme, followed by the conclusive remarks in Section 5.

2 Related work

2.1 Information hiding in halftone images

As mentioned before, since halftone visual sharing is closely related to halftone information hiding, several information hiding methods in halftone images are reviewed first. Pan et al. [21] proposed an authentication scheme by exploiting an image hash as a fragile watermark to embed in halftone images. Guo et al. [10, 11, 13, 14] have proposed several interesting halftone watermarking schemes. An information hiding scheme to embed a binary pattern into two or more halftone images with adaptive noise-balanced error diffusion was proposed [10]. They further developed a high-capacity watermarking method using the block truncation code [11] and a halftoning-based approach using direct binary search [14]. They also proposed to embed a multi-tone watermark in halftone images [13]. Son et al. [23] proposed a watermarking method for clustered halftone dots, in which the embedded binary data are extracted using dictionary learning. They also proposed to recover the color information from halftone images by inserting extra information [24, 25]. Wong [32] proposed the idea of multi-resolution halftoning and defined “constrained halftoning,” in which the embedded information is cast first, followed by the halftoning process. “Constrained halftoning” is quite related to our proposed work since the main objectives of both are maintaining the quality of halftone images and carrying error-free hidden message. He [33] then improved the previous scheme by an adaptive error diffusion filter. To further reduce artifacts and directional hysteresis caused by the causal nature of error diffusion filters, Fung et al. [8] employed MED to achieve multi-resolution halftoning by producing a set of halftoning results of different resolutions and embedding those lower-resolution images into the full-scale one such that they can be extracted by direct down-sampling.

For visual secret sharing in halftone images, Guo et al. [12] presented a low-complexity method using dot diffusion. The embedded information can be decoded visually by overlaying the original dot diffusion image and the secret-share dot diffusion copy. Zhou et al. [34] presented a visual cryptography scheme in halftone images to improve $EVCS$ [3] by making use of the complementary images to cover the visual information of the shares. The characteristics of blue-noise halftoning were employed in the share construction mechanism and the security properties of visual cryptography are still maintained. Myodo et al. [18, 19] proposed a visual cryptography scheme based on void-and-cluster halftoning and the other scheme based on error diffusion. Wang et al. [28] proposed a visual cryptographic scheme based on error diffusion and they [29] further presented a constrained iterative halftoning algorithm to generate better halftone shares.

2.2 Multi-scale error diffusion (MED)

The proposed scheme is developed based on MED. MED is a frame-based halftoning method, which can remove directional hysteresis by avoiding the predetermined sequential order of processing. The local average intensity of the output halftone image will resemble that of the input gray-level image by the principle of "maximum intensity guidance." Let X, B, and E be the input gray-level image, output halftone image and error image respectively. Without loss of generality, it is assumed that they are all of the size, $2^r \times 2^r$, where r is a positive number. The values of X are within [0, 1] so the white pixel with the luminance value 255 corresponds to 1 here. The elements in B are initially set as 0 so $E = X - B$ is equal to X. An image pyramid structure is employed with the image arrays, X_k with the size $2^k \times 2^k$, $0 \leq k \leq r$, being the down-scaled versions of the input image X or X_r. The elements of X_k are formed by

$$X_k(i_k, j_k) = \sum_{i=0}^{1}\sum_{j=0}^{1} X_{k+1}(2i_k + i, 2j_k + j), \tag{1}$$

where $0 \leq i_k, j_k \leq 2^k - 1$ and $0 \leq k \leq r - 1$. The error arrays E_k are initialized accordingly. The algorithm begins with the lowest resolution image or the top of the image pyramid, and proceeds by always selecting the quadrant with the highest average intensity. This procedure ends when a pixel of the original image has been reached and this pixel will be quantized by assigning a white dot, i.e., $B(i_r, j_r) = 1$. The quantization error is calculated by

$$e_q = E_r(i_r, j_r) - 1. \tag{2}$$

We then reset $E_r(i_r, j_r)$ as e_q and diffuse the error to the neighbors of (i_r, j_r) using a noncausal diffusion filter, followed by an update of the image pyramid. In other words, after the quantization and error distribution for a given pixel, the error-image quadtree E_k is updated so that the values at all resolutions are in accordance with the new error-diffused values at the finest resolution. The procedures are repeated until the sum of elements in E_r is bounded in absolute value by 0.5. Figure 4 shows two examples of the resultant images by MED.

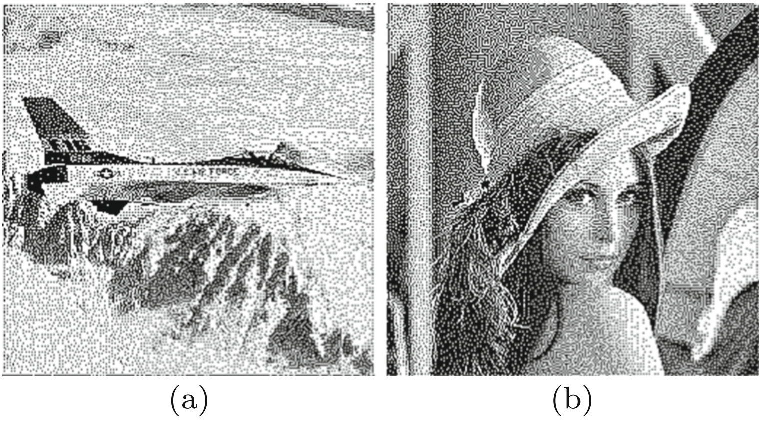


Fig. 4 Two halftone images generated by MED

3 Secret sharing by multi-scale error diffusion

A straightforward implementation of information hiding in a halftone image is to substitute a subset of pixels with a binary code. If the extractor knows the exact order of chosen pixels, the embedded binary code can be retrieved correctly. Two ways of applying information hiding in halftone images can be realized; one is to change the pixels of a halftone image directly and the other is to spread the secret binary code on a blank image, followed by the halftoning process. Although acquiring the same amount of hidden information, the constrained halftoning approach [32, 33] should yield better image quality. It is worth noting that the constrained halftoning can work with any conventional error diffusion method. However, since pixels are processed according to a predefined sequential order, a constrained pixel will not be taken into account until it is really encountered. The value assigned to this constrained pixel may thus cause abrupt changes in a local area and degrade the quality of an output image.

The proposed visual secret sharing scheme employs MED, which is considered more suitable to information hiding because of its frame-based nature to compensate the disturbance caused by the mismatch in the dot assignment of constrained pixels when processing the unconstrained ones. A group of continuous-tone gray-level images with the same size $2^r \times 2^r$, $\mathbf{X} = \{X^{(1)}, X^{(2)}, \ldots, X^{(M)}\}$, are selected and the hidden secret image, which is also a halftone one, is $B^{(S)}$. M halftone host images, $\mathbf{B} = \{B^{(1)}, B^{(2)}, \ldots, B^{(M)}\}$, will be used to carry $B^{(S)}$. A simple rule can be set as follows. For the position (i, j) of all the host halftone images, an even number of white pixels will be decoded as "0" while an odd number will generate a "1". The decoded binary data will be equal to $B^{(S)}$ or

$$\{B^{(1)} + B^{(2)} + \ldots + B^{(M)}\}\%2 = B^{(S)}, \tag{3}$$

where "%" is the modular operator. Although the secret message may not be revealed by directly stacking the shares, examining the parity of sums is straightforward and can be applied very efficiently.

3.1 Initial host images

All the output images $\mathbf{B}$ are initially blank and we will form the initial host images according to $B^{(S)}$ only. To be more specific, if $B^{(S)}(i, j) = 1$, one image out of $\mathbf{B}$ will be chosen and a white dot is assigned at (i, j). A reasonable strategy is to select $B^{(m)}$, $1 \leq m \leq M$, if $X^{(m)}(i, j)$ has the largest value among the M gray level images at this position. Figure 5 shows an example with four host images, "Lena", "Peppers", "Baboon" and "Boat" to carry the secret image "F-16". Since each position of the four images has at most one white dot, adding up these images will demonstrate the secret image perfectly. The processing will follow the raster scan. As in MED, after a white dot is assigned, the error diffusion is applied and the error quadtree is also updated. For a host image, $X^{(m)}$, its binary initial image is stored as $F^{(m)}$, in which "1" indicates that a white dot is assigned by only considering $B^{(S)}$ and the largest value in $\mathbf{X}$ at the same position.

3.2 "Painting" on dirty papers

At the beginning, the output images $\mathbf{B}$ are equal to $\mathbf{F} = \{F^{(1)}, F^{(2)}, \ldots, F^{(M)}\}$. We will employ MED to assign white dots on $\mathbf{B}$ such that the processed halftone images will look similar to the input images $\mathbf{X}$. The error quadtrees $\mathbf{E} = \{E^{(1)}, E^{(2)}, \ldots, E^{(M)}\}$ are used to select the image to be processed and decide the position to assign white dots. To be more

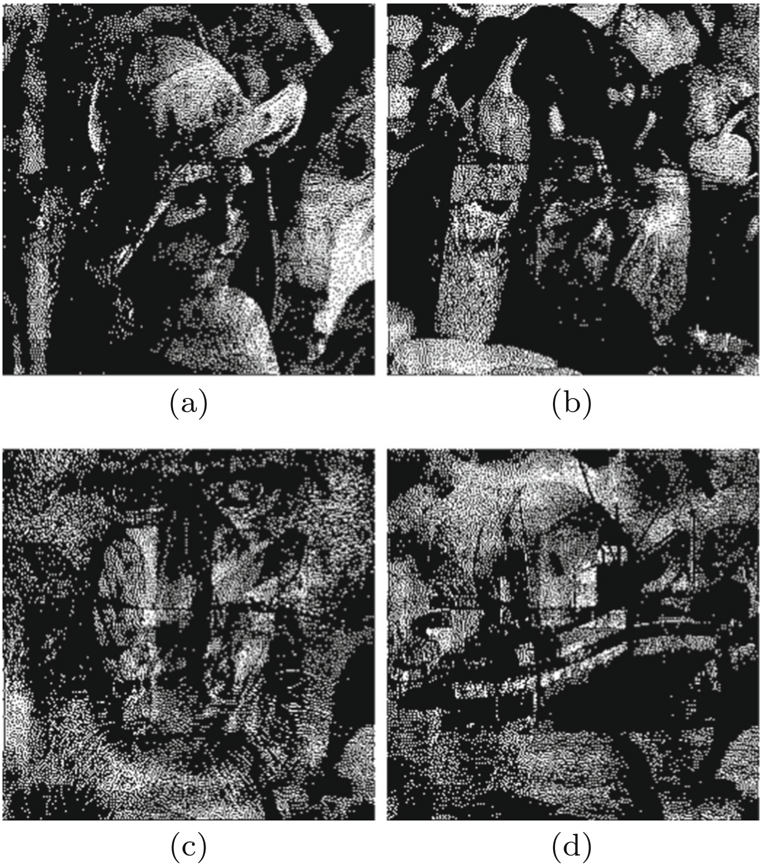


Fig. 5 Assigning white dots according to the secret image and luminance values of cover images to form the initial binary images

specific, $\{E_0^{(1)}, E_0^{(2)}, \ldots, E_0^{(M)}\}$ are examined and the image with the largest value will be selected. The subsequent procedure is the same as MED; the error quadtree is checked from root to leave to find the largest intensity. When the finest level is reached, a white dot will be assigned. It should be noted that, as in the original MED, the position already assigned with a white dot will not be selected again since (2) will generate a negative value. Nevertheless, any added white dot in this step will violate the constraint of (3). If $B^{(n)}(i, j)$ is chosen and will be changed from "0" to "1", one of the following three options will be taken to ensure that $B^{(S)}$ can be perfectly retrieved at any time.

Option 1: If $\sum_{m=1}^{M} F^{(m)}(i, j) = 1$, or more specifically, only one of $\mathbf{F}(i, j)$ is equal to 1, say $F^{(p)}(i, j) = 1$, and others are 0, i.e., $F^{(m)}(i, j) = 0$, $1 \leq m \leq M$, $m \neq p$, then $B^{(p)}(i, j)$ will be modified from "1" to "0" and $F^{(p)}(i, j)$ is set to "0" too. We can see that, at any position (i, j), there will be at most one image with $F(i, j) = 1$. In other words, there is at most one white dot assigned at a position without using MED. By modifying $B^{(p)}(i, j)$, the error should be adjusted by

$$e_q = E_r(i, j) + 1. \tag{4}$$

The error is then diffused to the neighboring areas and the error quadtree also has to be updated accordingly.

Option 2: If $\sum_{m=1}^{M} F^{(m)}(i,j) = 0$ and a white dot is now assigned by MED, we need to add another white dot since all the other white dots at this position are processed by MED too and cannot be changed. Therefore, two white dots will be added in two images simultaneously. Again, we check the values of remaining $\mathbf{X}(i,j)$ and select the output image with the largest luminance value, say $B^{(p)}(i,j)$, to flip it from "0" to "1". This process is similar to the step of initialization and $F^{(p)}(i,j)$ is set as "1", which can be changed back to "0" in the future if the condition of Option 1 is met.

Option 3: As shown in Option 1 and 2, when a white dot is assigned at (i,j) by MED, we either intentionally add one more white dot in one of the other images at the same position or remove a white dot associated with a nonzero $F(i,j)$. The third option is applied when $\sum_{m=1}^{M} F^{(m)}(i,j) = 0$ and $\sum_{m=1}^{M} B^{(m)}(i,j) = M-1$. In this case, the pixel values at (i,j) in $M-1$ out of M output images are white and now MED chooses the same position of the remaining one image, say $B^{(p)}(i,j)$, to assign a white dot. To avoid retrieving a wrong pixel value in the extraction process or to maintain the constraint in (3), the assignment of this white dot has to be skipped, i.e., $B^{(p)}(i,j)$ will remain "0". However, the update of error quadtree is still necessary as if $B^{(p)}(i,j) = 1$ to prevent selecting $B^{(p)}(i,j)$ again.

3.3 Generating the final host halftone images

The iterative procedure of MED will stop when the sum of all elements of E_r is bounded in absolute value by 0.5. The other way is to determine the number of white dots that should be assigned in images, $R^{(m)}$, $1 \leq m \leq M$, by

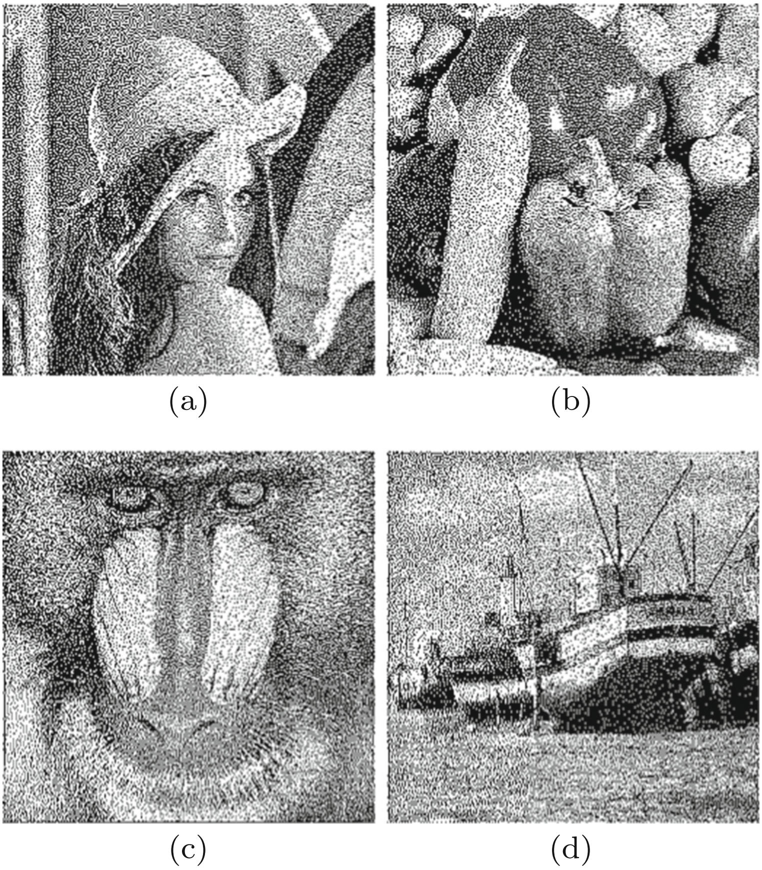
$$R^{(m)} = \lfloor \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X^{(m)}(i,j) + 0.5 \rfloor. \tag{5}$$

$R^{(m)}$ is equal to the nearest integer of luminance sum in the gray-level host image. The proposed scheme thus keeps track of the number of white dots in the halftoning process. If the number of assigned white dots reaches its limit, this image will be pulled out of consideration of data modification temporarily, unless its number of white dots is reduced because of Option 1 mentioned in Section 3.2.

Figure 6 shows the resultant four shares of the previous example. White dots are spread on Fig. 5 to form the final host images. To access the quality of the processed images, the weighted $SSIM$ [30] is employed and its setting is the same as the one used in [22]. Given two windows x and y with the size $K \times K$ $(K = 11)$ in the images X and Y respectively, the $SSIM$ index is calculated by

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{({\mu_x}^2 + {\mu_y}^2 + c_1)({\sigma_x}^2 + {\sigma_y}^2 + c_2)}, \tag{6}$$

where the mean intensity, μ_x, μ_y, the variance, ${\sigma_x}^2$, ${\sigma_y}^2$ and the covariance, σ_{xy}, are used to compute the differences of luminance, contrast and structure. The parameters c_1, c_2 stabilize the division with weak denominator. The $SSIM$ indices are then averaged in the entire image to acquire the mean $SSIM$ or $MSSIM$. To evaluate the quality of halftone images, $K \times K$ Gaussian filtering is applied to both the gray level original image X and the tested halftone image Y before the calculation of $MSSIM$. Table 1 shows the quality evaluation on the images participating the secret sharing. The first and second rows list the $MSSIM$ of



(a) (b)

(c) (d)

Fig. 6 Four resultant shares: (**a**) “Lena”, (**b**) “Peppers”, (**c**) “Baboon” and (**d**) “Boat” to embed “F-16”

applying MED and the resultant shares after information embedding respectively. Since the secret image “F-16” is perfectly extracted, a “-” mark is added. The reduction of $MSSIM$ is around 0.09 in all the four images of this example and considered acceptable by examining the resultant images demonstrated in Fig. 6.

3.4 Selecting host images

It should be noted that the selection of host images is an important issue in the proposed secret sharing scheme. Let’s check another example of using “Lena”, “Cameraman”, “Tiffany” and “Man” to embed the same secret image “F-16”. The resultant shares are

Table 1 $MSSIM$ of images participating in an example of secret sharing

Images	F-16	Lena	Peppers	Baboon	Boat
MED	0.7852	0.8273	0.8207	0.8407	0.8170
Share	–	0.7323	0.7389	0.7422	0.7237
Reduction	–	0.0950	0.0818	0.0985	0.0933

shown in Fig. 7, in which the introduced noises are more visible than those in Fig. 6. The $MSSIM$ values of resultant shares, which are listed in Table 2, are much lower and the reduction of $MSSIM$ when compared with the images processed by MED only is also more obvious. This negative effect is mutual so it can be observed that the quality of “Lena” shown in Fig. 7 also becomes worse. Among the four images, “Tiffany” is affected most seriously probably because it is the brightest image and lacks discriminating textures for generating a good halftone image. Besides, the lack of balance in average luminance in the host images is also responsible for the deteriorated performance in “Tiffany”. In the initial images shown in Fig. 8, the average luminance values of “Lena”, “Cameraman”, “Tiffany” and “Man” are 124, 167, 211 and 89 respectively. The darker contents make the initial images of “Lena” and “Man” almost blank and the entire contour of the plane is drawn on “Tiffany”. As mentioned in Section 3.2, the secret shares are painted on the four initial images so the initially assigned white dots have an impact on the image quality. The numbers of white dots in the initial images also provide some information. Compared with the numbers of white dots in Fig. 8 (1486, 15123, 28926 and 390), those in Fig. 5 (10962, 11443, 11727 and 11793) are much more similar in these four initial images so the quality of all the resultant shares can be maintained in a better way. The quality degradation will be

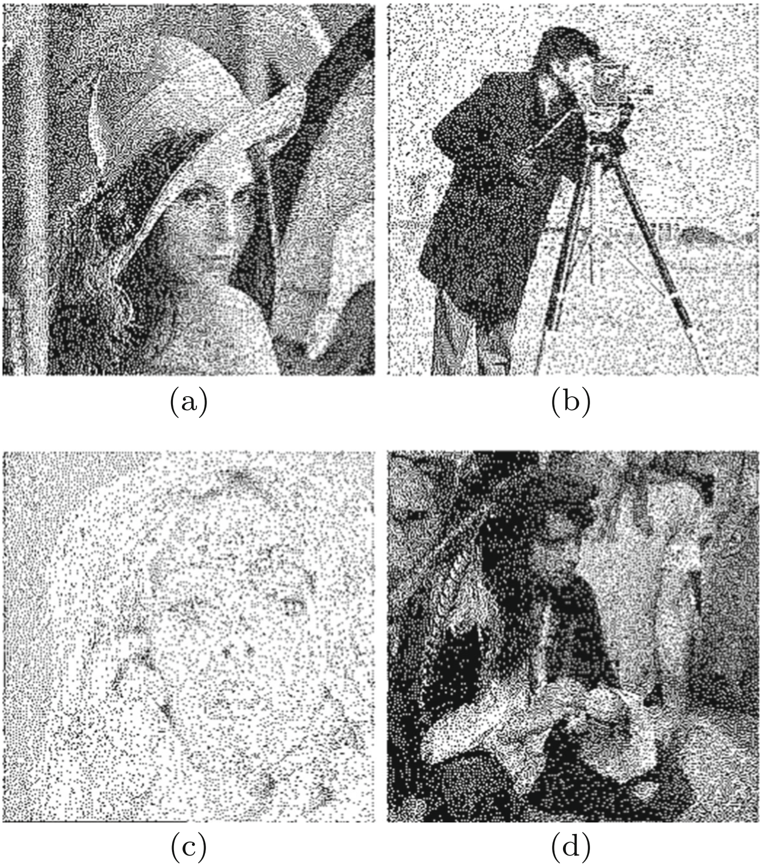

(a)
(b)
(c)
(d)

Fig. 7 Another example of four resultant shares: (**a**) “Lena”, (**b**) “Cameraman”, (**c**) “Tiffany” and (**d**) “Man”, to embed “F-16”

Table 2 $MSSIM$ of images in another example of secret sharing

Images	F-16	Lena	Cameraman	Tiffany	Man
MED	0.7852	0.8273	0.7520	0.7491	0.8064
Share	–	0.6916	0.6376	0.5678	0.6966
Reduction	–	0.1357	0.1144	0.1813	0.1098

more serious if many white dots are assigned initially, as in the case of "Tiffany" in Fig. 8. A halftone image should have a better quality when a larger portion or area is formed by MED, instead of the initialization process described in Section 3.1.

Furthermore, if a large uniform area appears in an image, the white dots placed by MED tend to be evenly distributed in these regions. Any disturbance on these evenly spread dots may cause unpleasant quality degradation and decrease $MSSIM$ as well. In this example, we can see that there exist very few white dots in the black areas of the MED-processed halftone images "Cameraman" and "Man". Due to the constraint enforced by the secret image to be embedded, the proposed scheme has to cast some white dots in these black areas. The sporadic white dots seem to be more noticeable as shown in Fig. 7d and the

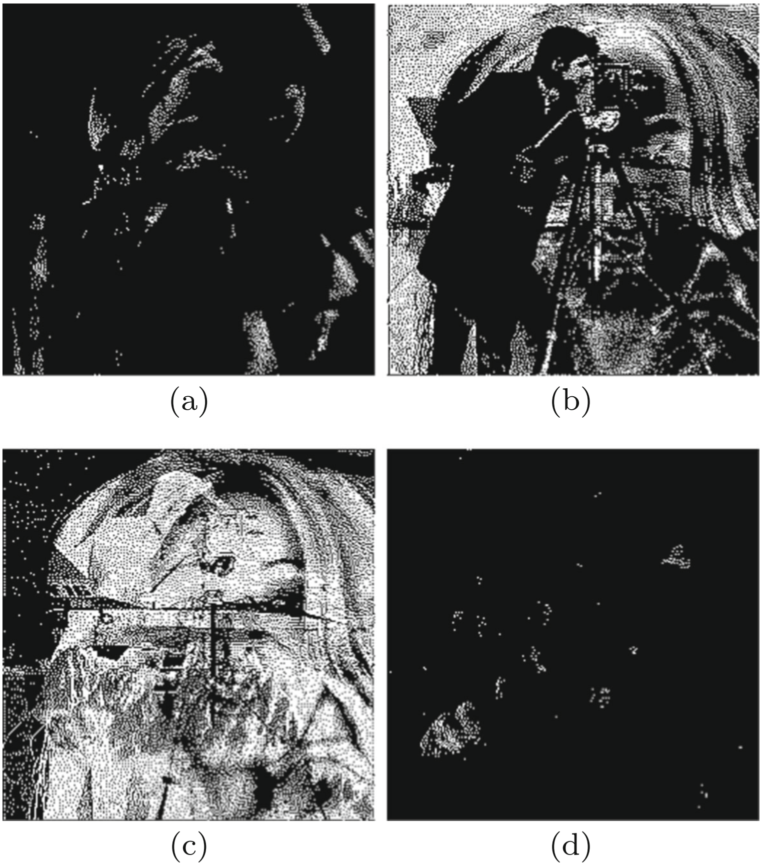


Fig. 8 The initial images when the average luminance values are less balanced

perceptual quality will be affected. Large white areas seem to be affected less by the secret sharing process probably because of the "maximum intensity guidance" adopted in MED.

It is also worth noting that the content structure of secret image itself may also play a certain role. When a large white or black area exists in the secret image, e.g. a handwritten note, it is challenging to maintain the security and quality of resultant shares. To reduce such a concern, the secret image should always be scrambled beforehand by, for example, computing XOR of the secret image with a random binary pattern. This additional step helps to avoid the trouble of dealing with various kinds of secret images since the resultant secret image will usually have a mean value close to 0.5 with almost equal numbers of black and white pixels. This random binary pattern is certainly required in the extraction process to successfully recover the secret image.

Based on the above discussions, the selection of images mainly relies on examining the $MSSIM$ values of applying MED only. First, a low $MSSIM$ indicates that the process of transferring an image to its halftone version has already affected the picture quality so this image should not be taken into consideration. Besides, the images with higher $MSSIM$ values usually contain more varying contents and will be more suitable to carry the secret

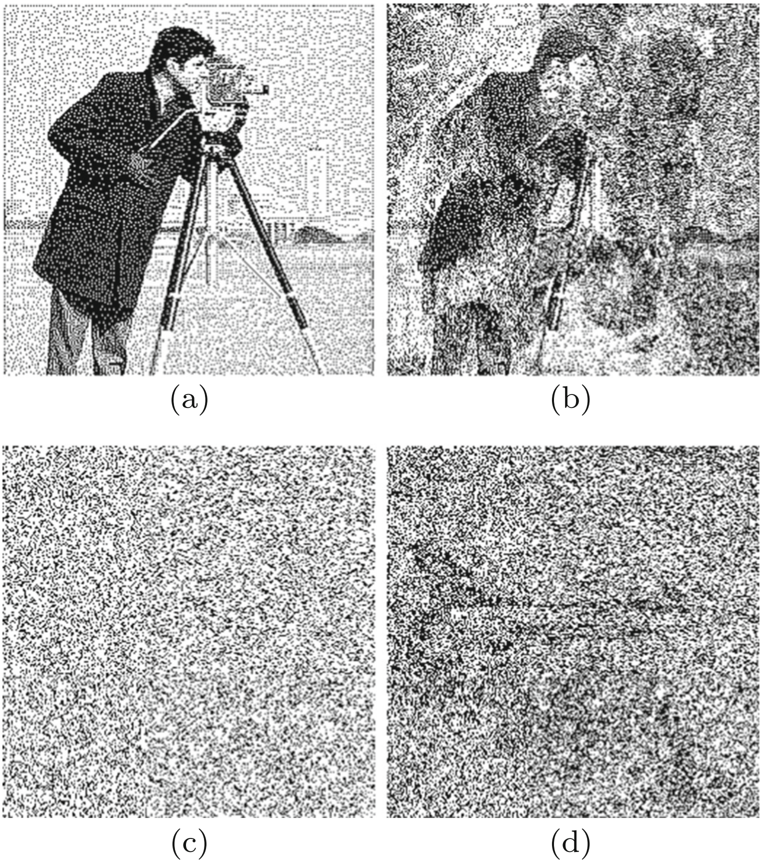


Fig. 9 Dealing with the problem of extracting the secret image with fewer shares: (**a**) the secret image "Cameraman", (**b**) extracting the secret images with three images only without "Man", (**c**) the scrambled secret image and (**d**) the extracted image from the three shares

image, which is also noise-alike as the scrambling process is applied in advance. Moreover, it is observed that the decrease of $MSSIM$ after the secret sharing process is less serious for the images with high $MSSIM$ values. Assuming that an image database is available and any suitable image can be selected as a share or host, we simply select M images with the highest $MSSIM$ values when comparing its MED halftone version with the original image. It should be noted that the candidate image may be further excluded if it contains large black areas. A simple method to determine the existence of black areas is to process the MED halftone image by a 3 × 3 all-one filter. A pixel will be identified as a white dot if

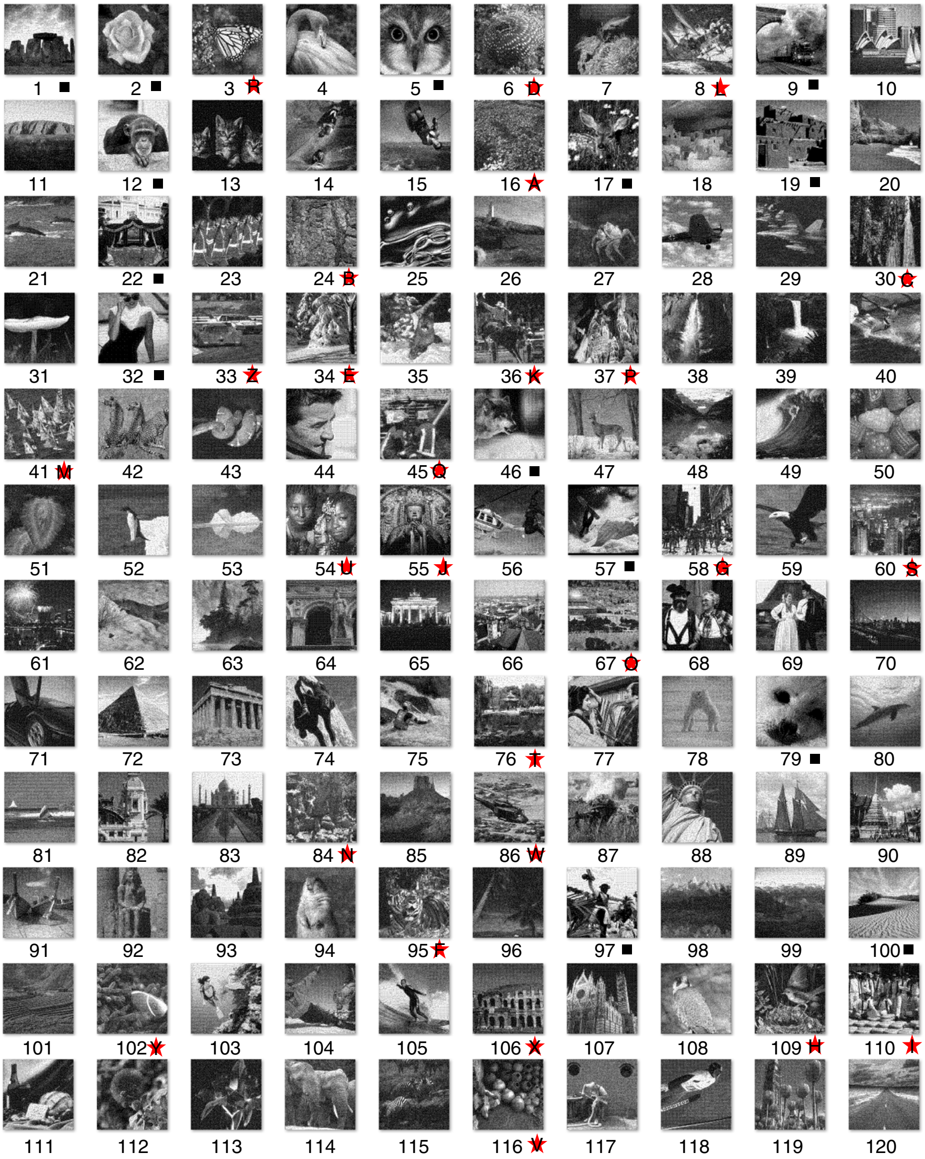


Fig. 10 The image set and the candidates

its filtered value is larger than 1, and as a black pixel otherwise. The connected components with black pixels are calculated. An image will not be taken into account if its largest black connected component occupies a certain percentage of the image.

3.5 Scrambling secret halftone image

The extraction of secret halftone image is simple. After collecting all the host halftone images, the number of the white dots in each position of the host halftone images will be recorded. If the number of white dots is odd, the secret halftone image has a white pixel at this position; otherwise, a black pixel is decoded. One requirement of the proposed scheme is that the message can only be perfectly extracted when all the participating shares are available. Since the proposed scheme always maintains the required parity of the number of white dots in each position during the embedding process, perfect reconstruction of secret image is certainly achievable. Nevertheless, the proposed secret sharing also requires that the extraction based on fewer shares cannot reveal any imagery information. Here we show a serious case, in which "Lena", "Man", "F-16", and "Boat" are used to carry "Cameraman" as shown in Fig. 9a. The extraction is applied using three images only, i.e., "Lena", "F-16" and "Boat". Figure 9b shows the extracted image, from which we can see that the contour of

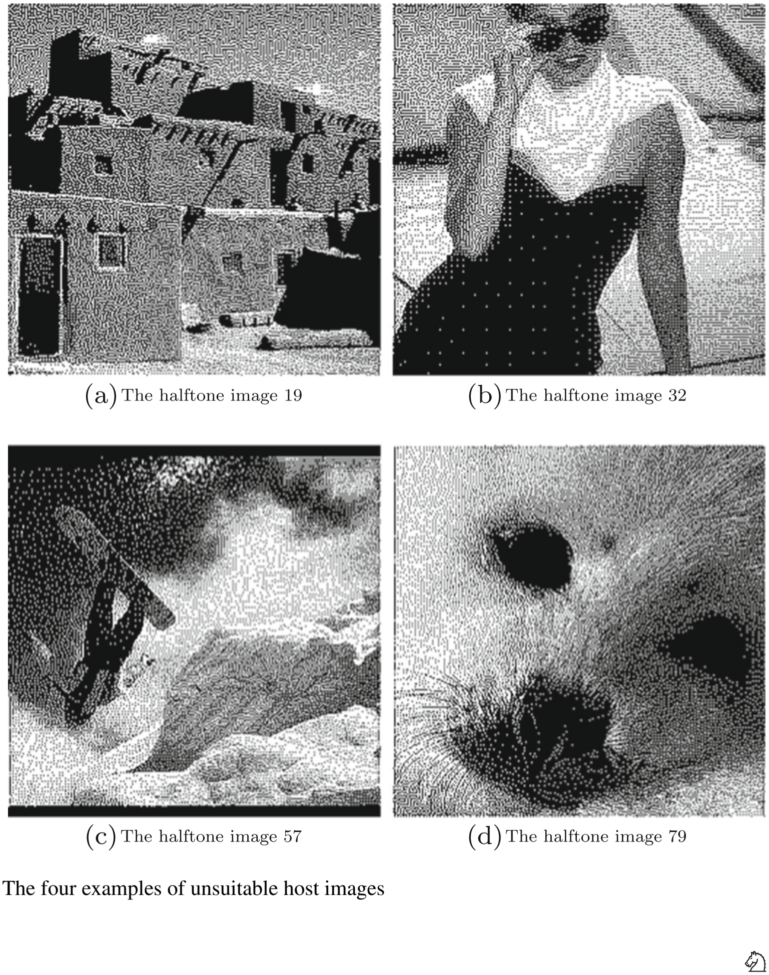
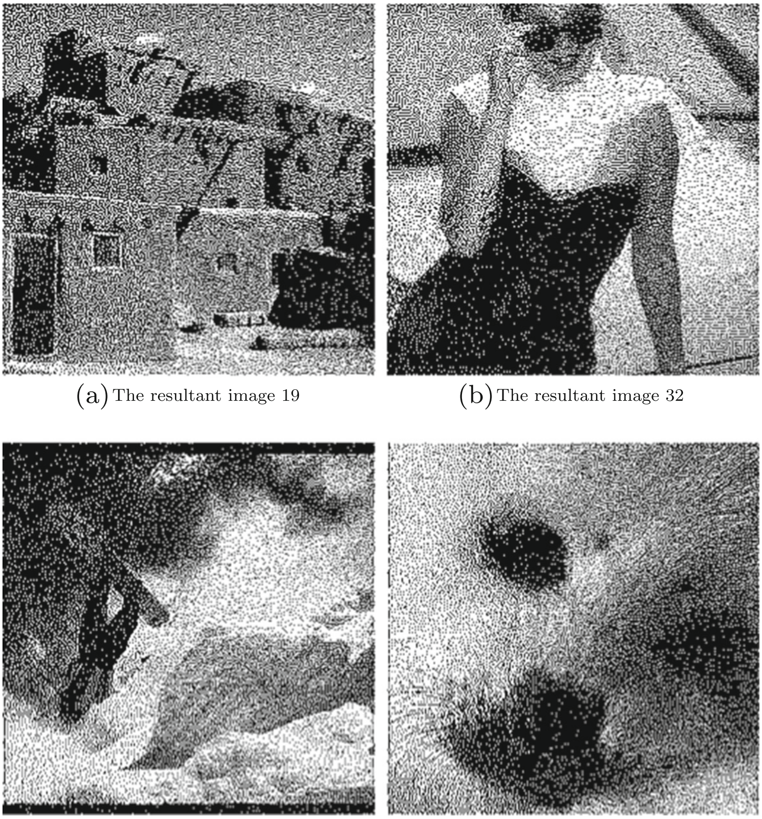

(a) The halftone image 19
(b) The halftone image 32
(c) The halftone image 57
(d) The halftone image 79

Fig. 11 The four examples of unsuitable host images

"Cameraman" can be partially observed. Although the scrambling of secret image by using XOR with a random pattern or changing the mapping of the numbers of white dots and the representative pixels in the secret image may help a bit, this problem still exists if an unauthorized user also acquires the scrambling binary pattern or secret mapping. A feasible solution is to ensure that the hidden image can be viewed only if all of its parts are correctly extracted such that partial revealing is not possible. A preprocessing step is thus adopted to relate the pixel values of the secret image. Assume that $B^{(S)}$ is the secret image and its shuffled version, $\tilde{B}^{(S)}$, is formed by scrambling the positions of pixels. $\tilde{B}^{(S')}$ is acquired by calculating XOR with adjacent pixels, i.e.,

$$\tilde{B}^{(S')}(\lfloor\frac{k}{2^r}\rfloor, k\%2^r) = \tilde{B}^{(S)}(\lfloor\frac{k}{2^r}\rfloor, k\%2^r) \otimes \tilde{B}^{(S')}(\lfloor\frac{k-1}{2^r}\rfloor, (k-1)\%2^r), \tag{7}$$

where $0 < k < 2^r \times 2^r$ and is equal to $i \times 2^r + j$. "%" and "⊗" are the modular and XOR operators respectively. $\tilde{B}^{(S')}$ is then reordered back to form $B^{(S')}$, which is the secret image to be embedded. $B^{(S)}$ can be recovered by (7) from the retrieved $\tilde{B}^{(S')}$. Figure 9c shows the scrambled "Cameraman" and Fig. 9d is the extracted image from the three shares. The incorrect extraction of some pixels will make the entire image obtained from fewer shares less intelligible.


(a) The resultant image 19

(b) The resultant image 32

(c) The resultant image 57

(d) The resultant image 79

Fig. 12 The resultant shares of using unsuitable images

3.6 Mutual embedding

Instead of specifying a halftone image that will be carried by M host images, an interesting practice termed "mutual embedding" allows any $M-1$ participants to decode the remaining one image. In other words, any share or halftone image can be the hidden image and extracted after all the other shares are available. The subtle difference from the above-mentioned scheme is that all the halftone images are processed by MED. Since no secret halftone image is determined in advance, the procedure described in Section 3.1 is not necessary. The initialization can also be viewed as embedding an all zero or black image into all of host images so the output image $B^{(m)}$ does not need to be assigned with any white dots beforehand. The embedding process is then the same as what we mentioned in Sections 3.2 and 3.3. If a halftone image $B^{(n)}$ is to be extracted from the other carriers, the number of white dots at each position in $M-1$ halftone images are recorded and its parity is checked as the number of white dots at each position is always even in this case. Nevertheless, since there is no specific or fixed secret image in this mutual embedding scenario, scrambling will not be applied to any image to ensure that all the halftone images can be viewable. Therefore, large uniform areas in the halftone images participating in the secret sharing should be avoided to reduce the risk of using fewer images (less than $M-1$) to reveal the intelligibility of a "missing" one.

4 Experimental results

The experiments are demonstrated in three parts; 1) selecting host images, 2) evaluating the effects from the numbers of host images, and 3) mutual embedding. 120 images from Corel

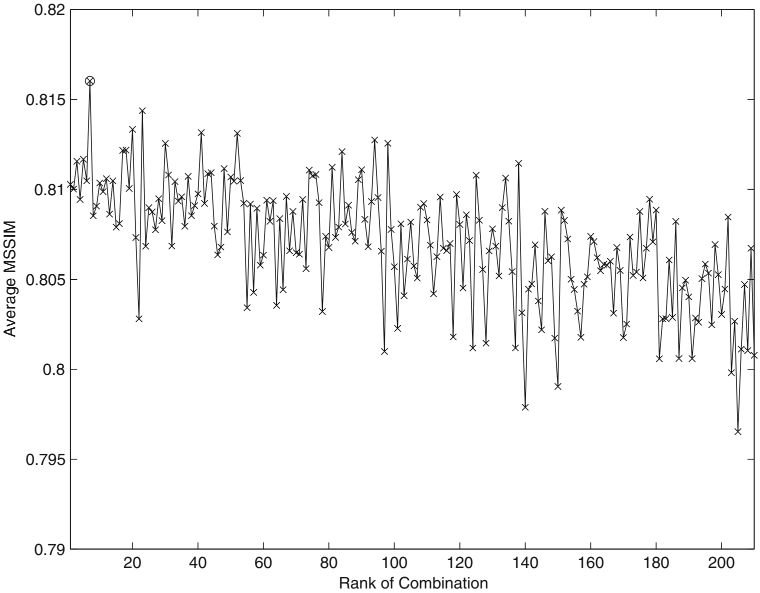
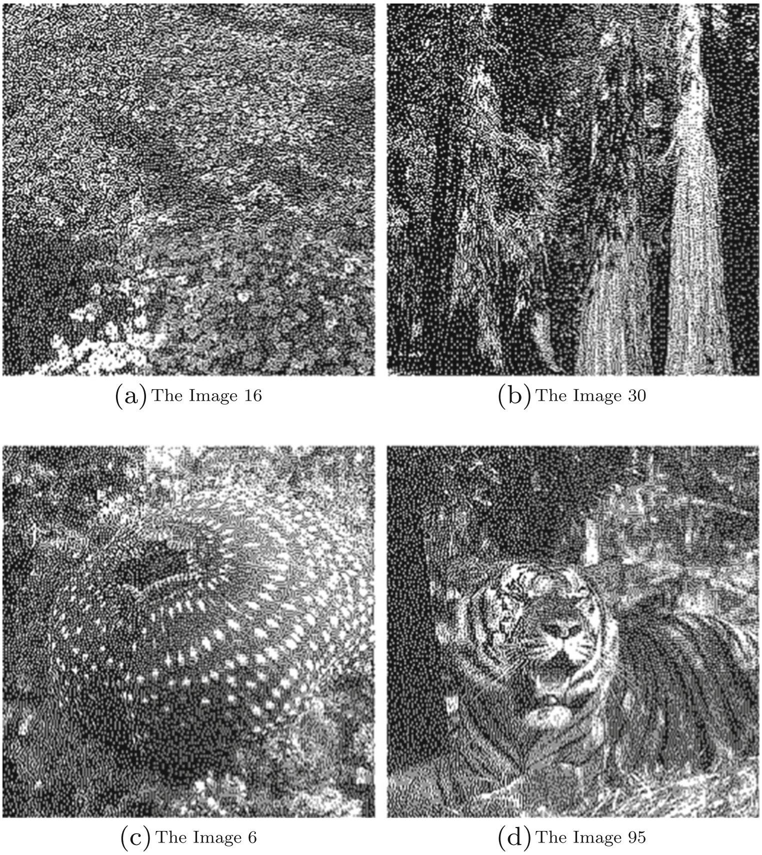


Fig. 13 The resultant $MSSIM$ of 210 ($C(10,4)=210$) image combinations from ten candidates in the secret sharing with four images

database [7] are used in this test and all of them are of the size 256 × 256. Figure 10 shows the halftone versions of these tested images processed by MED, which makes use of 3 × 3 noncausal diffusion filters to diffuse the errors.

We first follow the image selection method mentioned before to remove unsuitable host images. The halftone images containing large black areas will not be selected and the excluded images are shown with ■ symbols in Fig. 10. Four of them are demonstrated in Fig. 11. A quick example is using these four images as the hosts to generate secret shares and the resultant halftone images are shown in Fig. 12. By comparing Figs. 11 and 12, we can find that the process of information embedding generates sporadic white dots in black areas and may cause unpleasant distortions. It should be noted that this is a precautious step since some images (e.g., the images 32 and 79) are affected more but some images (e.g., the images 19 and 57) may be affected less. The objective here is to help select more suitable images and remove the images that could have poorer results.

The $MSSIM$ values of the remaining images compared with the original gray-level images are then calculated and sorted. The best 26 images are identified and shown in the order from A to Z in Fig. 10. The image 16 has the highest $MSSIM$ equal to 0.8831 and the image 33 is ranked 26^{th}, whose $MSSIM$ is 0.8440. High $MSSIM$ values indicate that


(a) The Image 16

(b) The Image 30

(c) The Image 6

(d) The Image 95

Fig. 14 The selected four host images

the contents of these images are more vivid and suitable to information embedding. Besides, the resultant $MSSIM$ after the secret sharing process should be higher too.

Considering the high $MSSIM$ values of these test images, a simple strategy is to select the images with highest $MSSIM$ values, e.g. the images 16, 24, 30 and 6 (A to D in Fig. 10) in this image set if four host images are required. The other strategy is to select the top K images measured by $MSSIM$, try all the combinations to embed a secret halftone image, and then select the one with the highest resultant average $MSSIM$. We adopt this strategy and an example is shown in Fig. 13, in which 10 candidates (A to J in Fig. 10) are available for choosing the four host images. The scrambled secret halftone image "F-16" will be embedded. The X-axis shows the 210 ($C(10, 4) = 210$) combinations ranked according to the original $MSSIM$ while the Y-axis is the resultant $MSSIM$ after the secret sharing process. The descending tendency indicates that the images with higher original $MSSIM$ will certainly have better chances to yield higher resultant $MSSIM$ after processing. The highest $MSSIM$ is marked with "$\circ$" and this combination is composed of the images 16, 30, 6 and 95, which are further demonstrated in Fig. 14 and will be selected as the four host images for testing. The resultant four shares are shown in Fig. 15. The iterative algorithm

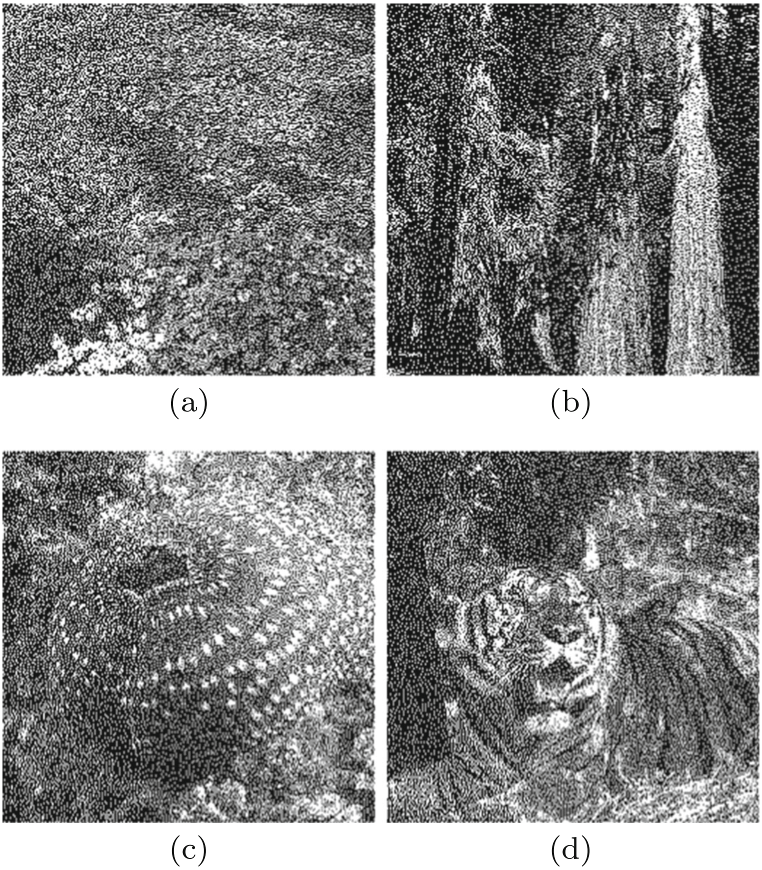
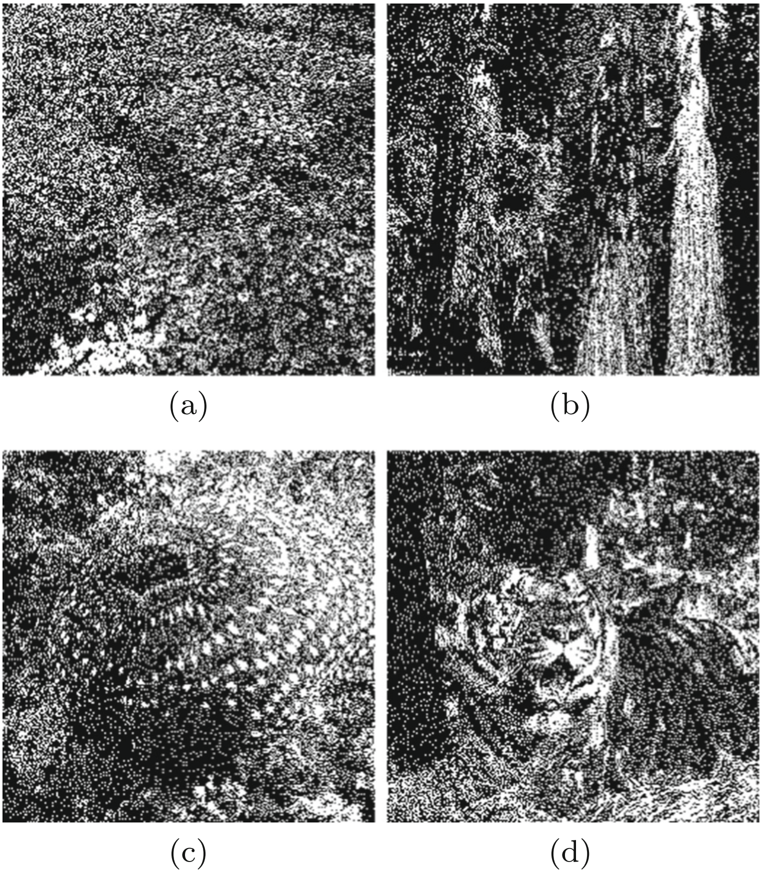


Fig. 15 The resultant halftone images

updates the error image quadtree after the quantization and error distribution for a given pixel and make the quality acceptable.

An intuitive information embedding method that modifies the pixels of halftone images directly is implemented as the comparison to show that the information sharing based on MED is a more appropriate approach. In this method, both the host images and secret image can be generated by any halftoning approach. To be more specific, we first measure the numbers of white dots in the host halftone images at all the positions. For the position (i, j) of the secret halftone image, the value of pixel "1" will result in an odd number of white dots in the same position of the host halftone images. If the actual number of white dots is different from the required number, we need to make a change by flipping a dot at this position in one of the halftone host images. The modification is based on the value of $X^{(m)}(i, j)$. When assigning a new white dot, we select the pixel with $B^{(m)}(x, y) = 0$ and having the largest intensity or $X^{(m)}(i, j)$. Similarly, if removing a white dot is required, we select the pixel with $B^{(m)} = 1$ and having the lowest intensity or $X^{(m)}(i, j)$. The decision of inserting or removing a white dot depends on the introduced error after the modification or quantization to ensure that the introduced error is the minimum. The images processed by this intuitive method are shown in Fig. 16 and we can see that there exist unpleasant binary


(a)

(b)

(c)

(d)

Fig. 16 The compared halftone images processed by the intuitive information embedding method

Table 3 The $MSSIM$ of four halftone images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8172(0.0624)	0.6113
30	0.8741	0.8203(0.0538)	0.6472
6	0.8727	0.8146(0.0580)	0.5916
95	0.8694	0.8120(0.0575)	0.5767
Average	0.8740	0.8160(0.0580)	0.6069

patterns that make the images blurred. Although this intuitive method can embed the secret halftone image into the carriers successfully, the poor image quality, which further raises the risk of being noticed during transmission, is the major concern.

The $MSSIM$ values of three cases, i.e., the original halftone images processed by MED only, the proposed method, and the compared/intuitive method, are listed in Table 3. The parentheses in the table show the $MSSIM$ decreases of the proposed method when compared with the original $MSSIM$. We can find that the $MSSIM$ difference of the image 30 is the lowest and that of the image 16 is the largest. The average $MSSIM$ values of the original halftone images, the resultant host halftone images of proposed scheme, and the intuitive scheme, are 0.874, 0.816 and 0.607 respectively. The $MSSIM$ decrease of the proposed scheme is quite limited as the resultant host halftone images better retain the detailed information. The good quality comes from the fact that MED can adjust the spreading of dots according to the constraints. The intuitive method generates low quality because this method does not consider the image structure and select the dots with the largest or lowest intensity directly. The much higher $MSSIM$ values of the proposed methods than the

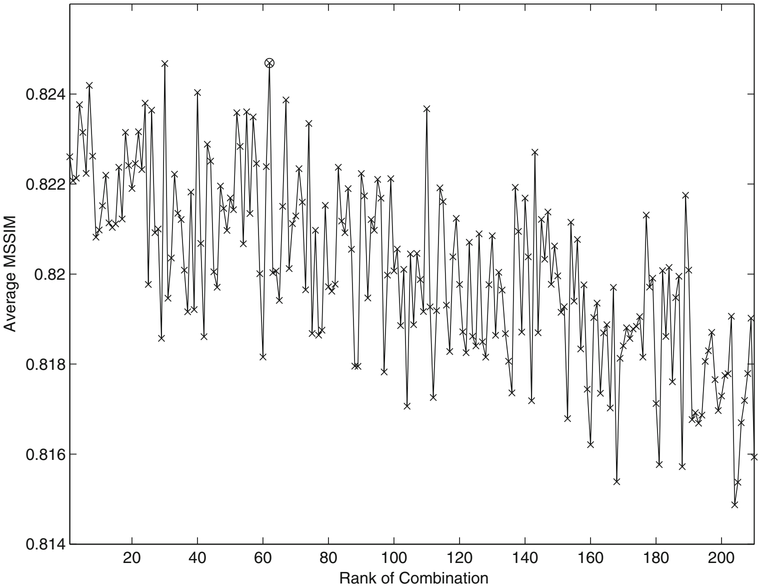


Fig. 17 The resultant $MSSIM$ of 210 ($C(10, 6) = 210$) image combinations from ten candidates in the secret sharing with six images

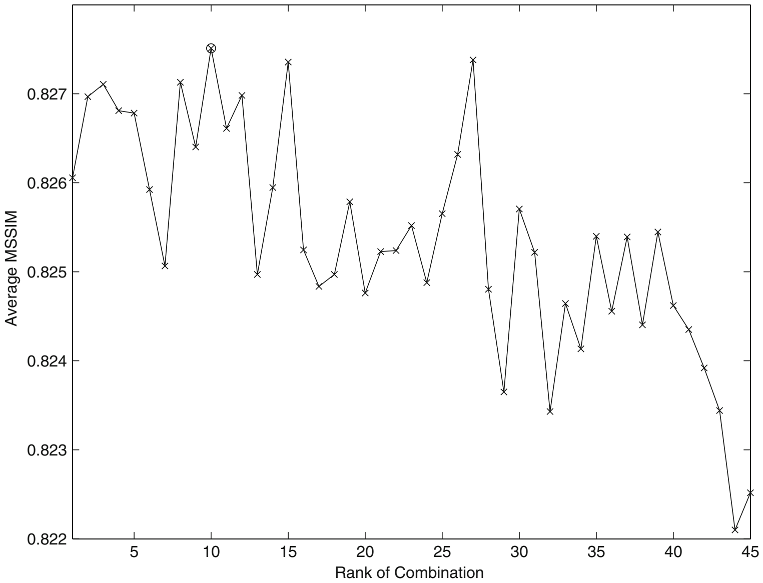


Fig. 18 The resultant $MSSIM$ of 45 ($C(10, 8) = 45$) image combinations from ten candidates in the secret sharing with eight images

intuitive method also indicate that $MSSIM$ is effective in reflecting the quality of halftone images.

When more images participate in this secret sharing scheme, the negative effects on quality in individual images will be further reduced since the proposed method can effectively disperse the noises in host images. We test the uses of six and eight host halftone images in the proposed scheme. The selections of six and eight host images are based on Figs. 17 and 18 respectively. In the case of using six host images, the images 16, 30, 6, 34, 110 and 55 in Fig. 10 are chosen. In the case of using eight host images, the images 16, 24, 30, 6, 34, 58, 110 and 55 are selected. Tables 4 and 5 show the image quality measured by $MSSIM$. We can see that the average $MSSIM$ decrease is smaller when more images are involved.

In the case of "mutual embedding", the secret halftone image is not assigned. When using four host images, any three images can be used to reveal the remaining one. The

Table 4 The $MSSIM$ of six halftone images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8289(0.0501)	0.6630
30	0.8741	0.8249(0.0492)	0.6718
6	0.8727	0.8262(0.0465)	0.6367
34	0.8711	0.8247(0.0464)	0.6607
110	0.8642	0.8243(0.0399)	0.6672
55	0.8628	0.8190(0.0438)	0.6775
Average	0.8708	0.8247(0.0461)	0.6628

Table 5 The $MSSIM$ of eight halftone images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8380(0.0416)	0.7012
24	0.8748	0.8261(0.0487)	0.7080
30	0.8741	0.8307(0.0434)	0.6989
6	0.8727	0.8306(0.0421)	0.6705
34	0.8711	0.8263(0.0448)	0.6927
58	0.8652	0.8213(0.0439)	0.6726
110	0.8642	0.8265(0.0377)	0.6864
55	0.8628	0.8206(0.0423)	0.7014
Average	0.8706	0.8275(0.0431)	0.6914

resultant images using the images 16, 30, 6 and 95 as the host images are shown in Fig. 19 and the $MSSIM$ values are listed in Table 6. Compared with Table 3, the $MSSIM$ values are slightly lower probably because the scenario of mutual embedding is equivalent to

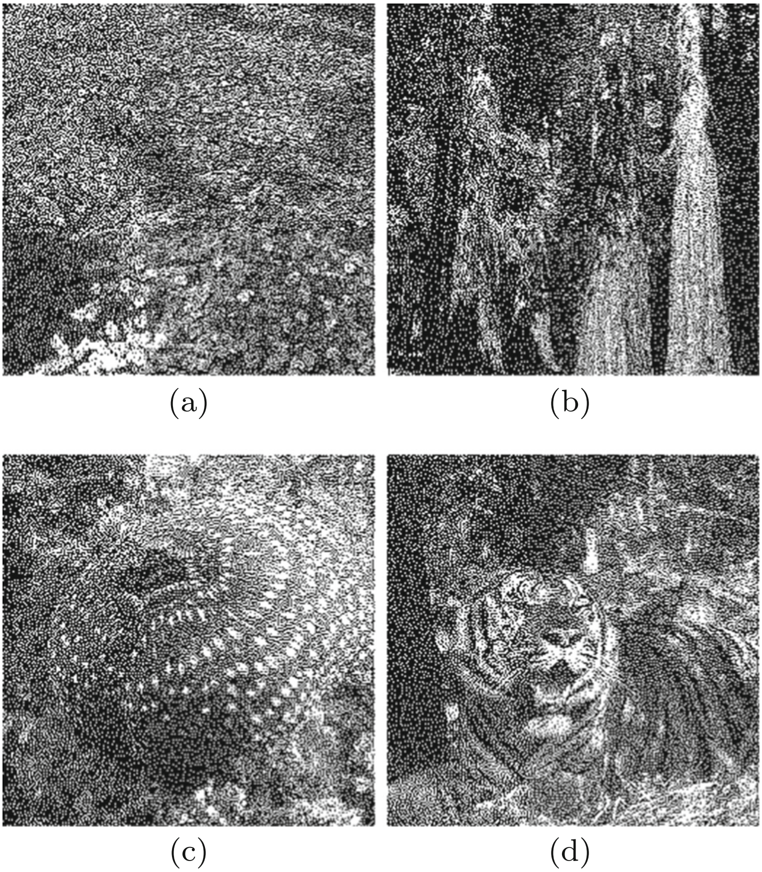


Fig. 19 The resultant halftone images in mutual sharing using four images

Table 6 The $MSSIM$ of mutual embedding with four host images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8082(0.0714)	0.5550
30	0.8741	0.8133(0.0608)	0.6063
6	0.8727	0.8070(0.0657)	0.5597
95	0.8694	0.8036(0.0658)	0.5355
Average	0.8740	0.8080(0.0659)	0.5641

embedding a blank or black "secret" image without any content. Nevertheless, the difference is still limited and the images in Fig. 19 look acceptable. The scenario also demonstrates the feasibility of the proposed scheme based on MED since any type of secret images will yield pretty good results. The $MSSIM$ of the cases using six and eight images are shown in Tables 7 and 8 respectively. As expected, when the number of images increases, the quality of images will become much better.

Finally, we would like to compare the proposed scheme with the state-of-the-art halftone visual secret sharing scheme [29]. As in Fig. 3, 512 × 512 "Baboon", "Man" and "Tank" are generated as the secret shares and demonstrated in Fig. 20. The hidden secret image is the scrambled "F-16". By comparing Figs. 3 and 20, we can say that both schemes do a good job in preserving the quality of halftone images. The characteristics of the resultant halftone images do differ as error diffusion was employed in [29] while MED is adopted in the proposed scheme. Error diffusion tends to spread dots more evenly while MED puts more white(black) dots in light(dark) regions. The major benefit of the proposed scheme over [29] is that the 512 × 512 halftone secret image, i.e. Fig. 20d, is perfectly transmitted, instead of a smaller 128 × 128 logo used in [29]. Although recognition of a logo without any computation in [29] can be viewed as one advantage, the proposed scheme further secures the hidden secret image by additional scrambling/descrambling as described before. It should be noted that the proposed scheme may also be modified to achieve such a functionality as direct visual decoding after stacking. We use a similar idea of Fig. 1 as an example. Assume that M shares will be generated for visual cryptography. To visually decode a 2×2 block as a black block, the scheme ensures that, for each of the four positions in this 2×2 block, at least one share has a black pixel. In other words, we have to prevent the M^{th} share to put a white dot at a position if the other $M - 1$ shares have done so. After stacking using "OR" operation with black being Boolean "1", the four positions will all be decoded as

Table 7 The $MSSIM$ of mutual embedding with six host images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8261(0.0535)	0.6421
30	0.8741	0.8254(0.0487)	0.6586
6	0.8727	0.8163(0.0564)	0.6308
34	0.8711	0.8243(0.0468)	0.6516
110	0.8642	0.8198(0.0444)	0.6522
55	0.8628	0.8135(0.0493)	0.6573
Average	0.8708	0.8209(0.0499)	0.6488

Table 8 The $MSSIM$ of mutual embedding with eight host images

Halftone Images	MED	Proposed Scheme(Decrease)	Compared Scheme
16	0.8796	0.8380(0.0416)	0.7012
24	0.8748	0.8261(0.0487)	0.7080
30	0.8741	0.8307(0.0434)	0.6989
6	0.8727	0.8306(0.0421)	0.6705
34	0.8711	0.8263(0.0448)	0.6927
58	0.8652	0.8213(0.0439)	0.6726
110	0.8642	0.8265(0.0377)	0.6864
55	0.8628	0.8206(0.0423)	0.7014
Average	0.8706	0.8255(0.0451)	0.6904

black pixels and a black block can be formed. On the other hand, to visually decode a 2 × 2 block as a white block, similar to Fig. 1, at least two positions of a 2 × 2 block have to be white. Therefore, all the M shares have to put white dots at these positions to make

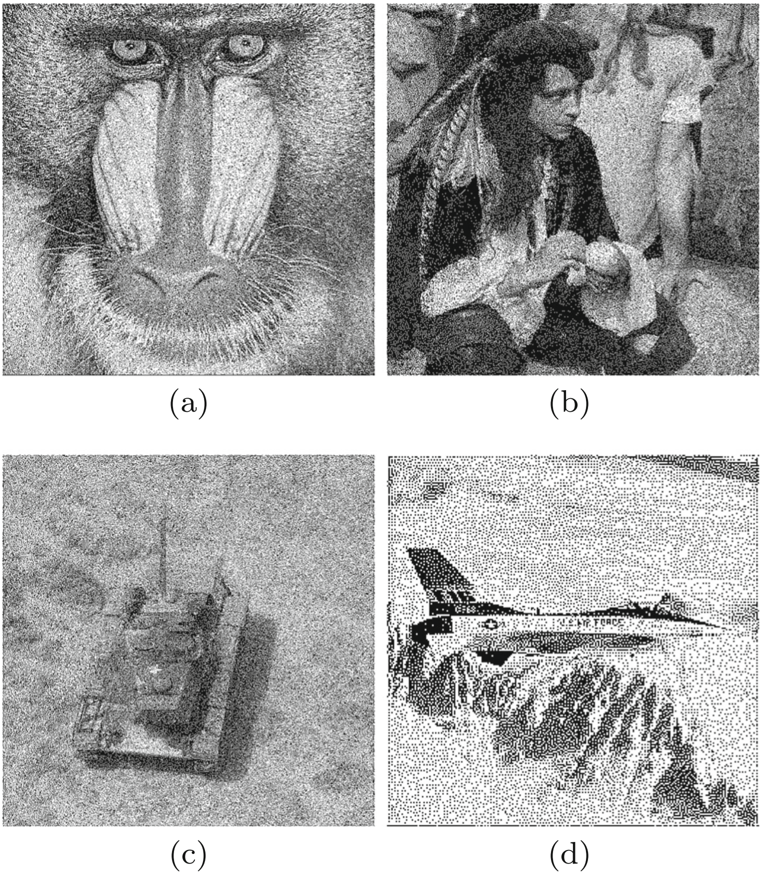

(a) (b) (c) (d)

Fig. 20 The 512 × 512 resultant host images of (**a**) "Baboon" (**b**) "Man" (**c**) "Tank" generated by the proposed scheme for embedding scrambled "F-16". **d** is the decoded image, which is the same as Fig. 4(a)

the 2×2 block look "brighter" after stacking. To facilitate this design, the logo had better use a black background painted with fewer white dots depicting the logo shape or outlines so that fewer positions have to become white forcibly. Nevertheless, the performance by doing so may not be as good as that of [29] because of the usage of MED in the proposed scheme, which tends to leave larger white/black areas than error diffusion. Imagine that all the M shares are forced to cast a white dot at a dark position but such a case happens at an inadequate (dark) region. A worse quality of a resultant halftone image may thus be expected. This is also a reason that we suggest to exclude the images with large dark areas from being secret shares, as mentioned before, to avoid generating possibly poor halftone images.

5 Conclusion

In this research, an innovative secret sharing scheme in halftone images based on multiscale error diffusion is proposed. During the transformation of several gray-level images to halftone ones, a secret halftone image is embedded in these images by controlling the number of white dots in each position. The experimental results show that grouping all the host halftone images can reveal the secret halftone image successfully. The main contributions of the research is to select suitable images, maintain the quality of halftone host images and achieve error-free transmission of a secret halftone image with the same resolution as that of the host images.

References

1. Adi S (1979) How to share a secret. Commun ACM 22(11):612–613
2. Analoui M, Rogowitz BE, Allebach JP (1992) Model based halftoning using direct binary search. In: Human vision, visual processing, and digital display III, vol 1666, pp 96–108
3. Ateniese G, Blundo C, De Santis A, Stinson DR (2001) Extended capabilities for visual cryptography. Theor Comput Sci 250(1-2):134–161
4. Beimel A (2011) Secret-sharing schemes: A survey. In: Proceedings of the third international conference on coding and cryptology, pp 11–46
5. Blakley GR (1979) Safeguarding cryptographic keys. In: Proceedings of the national computer conference, vol 48, pp 313–317
6. Cimato S, Yang JCN, Wu C-C (2014) Visual cryptography based watermarking. Transactions on Data Hiding and Multimedia Security IX. Lect Notes Comput Sci 8363:91–109
7. (1994). Corel Stock Photo Library. Corel Corporation
8. Fung YH, Chan YH (2006) Embedding halftones of different resolutions in a full-scale halftone. IEEE Signal Process Lett 13(3):153–156
9. Furht B (2008) Encyclopedia of multimedia springer
10. Guo JM, Tsai JJ (2009) Data hiding in halftone images using adaptive noise-balanced error diffusion and quality-noise look up table. In: IEEE International symposium on circuits and systems, vol 1, pp 201–204
11. Guo J-M, Liu Y-F (2012) High capacity data hiding for error-diffused block truncation coding. IEEE Trans on Image Processing 21(12):4808–4818
12. Guo JM, Huang JH (2010) Data hiding in halftone images with secret-shared dot diffusion. In: IEEE International symposium on circuits and systems, pp 1133–1136
13. Guo J-M, Liu Y-F (2010) Hiding multitone watermarks in halftone images. IEEE Multimed 17(1):34–43
14. Guo J-M, Su C-C, Liu Y-F, Lee H, Lee J-D (2012) Oriented modulation for watermarking in direct binary search halftone images. IEEE Trans Image Process 21(9):4117–4127
15. Jarvis JF, Judice CN, Ninke WH (1976) A survey of techniques for the display of continuous-tone pictures on bilevel displays. Comput Graphics Image Process 5(1):13–40
16. Katsavounidis I, Jay Kuo C-C (1997) A multiscale error diffusion technique for digital halftoning. IEEE Trans Image Process 6(3):483–490

17. Knuth DE (1987) Digital halftones by dot diffusion. ACM Trans Graph 6(4):245–273
18. Myodo E, Sakazawa S, Takishima Y (2006) Visual cryptography based on void-and-cluster halftoning technique. In: IEEE International conference on image processing, pp 97–100
19. Myodo E, Takagi K, Miyaji S, Takishima Y (2007) Halftone visual cryptography embedding a natural grayscale image based on error diffusion technique. In: IEEE International conference on image processing, pp 2114–2117
20. Naor M, Shamir A (1995) Visual cryptography. In: Advances in cryptograhy: EUROCRYPT, LNCS, vol 950, pp 1–12
21. Pan J-S, Luo H, Lu Z-M (2006) A lossless watermarking scheme for halftone image authentication. Int J Comput Sci Netw Secur 6(2):147–151
22. Pang W-M, Yingge Q, Wong T-T, Cohen-Or D, Heng P-A (2008) Structure-aware halftoning. ACM Trans Graph 27(3):89:1–89:8
23. Son C-H, Choo H (2014) Watermark detection from clustered halftone dots via learned dictionary. Signal Process 102(12):77–84
24. Son C-H, Choo H (2014) Color recovery of black-and-white halftoned images via categorized color-embedding look-up tables. Digital Signal Process 28:93–105
25. Son C-H, Lee K, Choo H (2015) Inverse color to black-and-white halftone conversion via dictionary learning and color mapping. Inf Sci 299:1–19
26. Ulichney R (1987) Digital halftoning. MIT Press, Cambridge
27. Ulichney RA (1993) The void-and-cluster method for dither array generation. In: Proc. SPIE, Human Vis. Visual Process., Digit. Displays IV, vol 1913, pp 332–343
28. Wang Z, Arce GR (2006) Halftone visual cryptography through error diffusion. In: IEEE International conference on image processing, pp 109–112
29. Wang Z, Arce GR (2010) Halftone visual cryptography by iterative halftoning. In: IEEE Int. Conf. acoustics, Speech and Signal Processing, pp 1822–1825
30. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: From error visibility to structural similarity. IEEE Trans on Image Process 13(4):600–612
31. Weir J, Yan W (2010) A comprehensive study of visual cryptography. Transactions on Data Hiding and Multimedia Security V. Lect Notes Comput Sci 6010:70–105
32. Wong PW (1996) Adaptive error diffusion and its application in multiresolution rendering. IEEE Trans on Image Process 5(7):1184–1196
33. Wong PW (2003) Multi-resolution binary image embedding. In: Proc. SPIE security and watermarking of multimedia contents v, vol 5020, pp 423–429
34. Zhou Z, Arce GR, Di Crescenzo G (2006) Halftone visual cryptography. IEEE Trans on Image Process 15(8):2441–2453

Po-Chyi Su was born in Taipei, Taiwan in 1973. He received the B.S. degree from the National Taiwan University, Taipei, Taiwan, in 1995 and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 1998 and 2003, respectively, all in Electrical Engineering. He then joined Industrial Technology Research Institute, HsinChu, Taiwan, as an engineer. Since August 2004, he has been with the Department of Computer Science and Information Engineering, National Central University, Taiwan. He is now an Associate Professor. His research interests include multimedia security, visual surveillance, digital image/video processing and compression.

Tzung-Fu Tsai was born in Taipei, Taiwan in 1989. He received the B.S. degree from the Ming Chuan University, Taoyuan, Taiwan, in 2012 and the M.S. degrees from the National Central University, Jhongli, Taiwan, in 2015, both in Computer Science and Information Engineering. His research interests include multimedia security, digital image/video processing and compression.

Yu-Chien Chien was born in Taichung, Taiwan in 1992. He received the B.S. degree from the National Taichung University of Education, Taichung, Taiwan, in 2015, majoring in Computer Science and Information Engineering. He is now working on the M.S. degree at the Department of Computer Science and Information Engineering, National Central University, Jhongli, Taiwan. His research interests include digital image/video forensics and information security.