International Journal of Applied Science and Engineering

Special issue: The 10th International Conference on Awareness Science and Technology (iCAST 2019)

An effective lossless dual-images based hiding method using block complexity and adopted thresholds

Tzu-Chuen Lu[1], Jau-Ji Shen[2*], Ting-Chi Chang[2]

[1] *Department of Information Management, Chaoyang University of Technology, Taichung, Taiwan*
[2] *Department of Information Management, National Chung Hsing University, Taichung, Taiwan*

ABSTRACT

Dual-images based hiding method is one kind of the hiding scheme which uses two duplicated images to conceal the secret message to generate the stego-images. Only a legal person who owns two stego-images can extract the correct message from the image. This kind of schemes been paid a lot of attention recently because it has better image quality and embedding capacity and realizes the concept of secret sharing. Lu et al. (2017) proposed a reversible dual-image based hiding scheme that used a block folding strategy to reduce the distortion between the cover image and the stego-image. They divided the secret message into two parts and encoded them into smaller digits to enhance the image quality of the stego-image. But their scheme needs an extra pixel to record the section numbers that limit the capacity of the secret message. This paper considers the complexity of the pixel to analysis how many bits can be embedded in the pixel to solve the problem. Furthermore, the proposed scheme applies an adaptive thresholds strategy to decide the thresholds for controlling the image quality. Experimental results show that the proposed scheme indeed improves the hiding performance.

Keywords: Dual-images; Center folding strategy; Block folding strategy; Reversible data hiding.

OPEN ACCESS

Received: March 4, 2020
Revised: March 24, 2020
Accepted: March 29, 2020

Corresponding Author:
Jau-Ji Shen
jjshen@nchu.edu.tw

Publisher:
Chaoyang University of Technology
ISSN: 1727-2394 (Print)
ISSN: 1727-7841 (Online)

1. INTRODUCTION

Information security has been paid more and more attention. One of the issues of information security is how to transmit secret message safely on Internet or mobile environment. Information hiding is a method that can effectively share secret message on cloud environment. One of the popular hiding schemes is dual-images based hiding scheme which divides the secret information into two parts and hides them in two identical cover images to produce stego-images. The receiver must obtain two stego-images at the same time in order to extract the complete secret information.

Chang et al. (2007) proposed the first reversible dual-images reversible data hiding method. They combined the dual-images technology with the Exploiting Modification Direction (EMD) by using the modulus function to generate a 256 × 256 matrix and converted the 2 secret bits into a 5-ary secret symbol. The scheme transformed the two original pixels with a pair and mapped the pair to the matrix to find a mapping value which has the same value with the secret symbol in the diagonal direction. Chang et al. (2009) changed the diagonal direction to horizontal and vertical direction to improve embedding capability. Lee et al. (2009) proposed an asymmetric dual image technique combined with EMD. The first stego-image is hidden by traditional EMD method, and the second stego-image is modified by referring to the first stego-image. In the study by Lee et al. (2009), 4 secret bits correspond to a group of four directions. In order to achieve reversibility, the latter 2 secret bits must correspond to the opposite or clockwise direction of the former2 secret bits before they can be hidden. In the study by Lee and Huang. (2013), the authors define 25 embedding rules, which correspond to 5-ary secret symbols, and the embedding capacity is effectively increased. The rules defined by Lee and Huang (2013) are modified by modular function to obtain better embedding capacity. In the study of Lu (2015), the least significant bit (LSB) matching method is applied to dual images. The reversible pixels are unchanged and the irreversible pixels are changed by using the rule table.

In the study by Lu et al. (2015), the center folding strategy is proposed, and secret symbols are encoded into smaller digits to reduce distortion. In the study by Lu et al. (2017), they extended the center folding strategy and encoded the secret symbols according to its occurrence frequency to improve the image quality. In the study by Yao et al. (2017), the authors find that the center folding strategy has the problem of insufficient utilization. They increase the modification cases and control the embedding rate by expanding the number of embedding bits K. In the study by Lu and Leng (2017) a dual-images technique based on block folding is proposed, which folds secret symbols twice to reduce distortion greatly. In the study by Yao et al. (2018), the best length of K is selected adaptively according to the required embedding rate.

The method proposed by Lu and Leng (2017) decomposes the secret symbols into two segments and encodes each segment using the center folding strategy to improve the image quality. However, in order to know what segment the folded code belongs to, an additional pixel is needed to record the segment number that will reduce the embeddable space.

In this study, we modify the scheme by Lu and Leng (2017) by adjusting the length of K according to the complexity of the block to solve the problem. The proposed scheme does not need the additional pixel such that can enhance the embedding capacity and image quality.

2. RELATED WORKS

2.1 Center folding strategy and Frequency-based encoding strategy

The center folding strategy (Lu et al., 2015) folds the value in the value range by subtracting the middle value from each value in the range table. The scheme takes K secret bits as a group to convert into a decimal secret symbol d. The value range of the secret symbol is $d \in [0,\ 2^K - 1]$. The value 2^{K-1} is the middle value. The middle value is subtracted from each value in the range table to generate the folded values. The center value is 0.

Take $K = 3$ as an example, the value range is $d = \{0,1,2,3,4,5,6,7\}$. The value range after folding is $\hat{d} = \{-4,-3,-2,-1,0,1,2,3\}$. Then, the folded secret symbol is split into two parts to be hidden in two identical stego-images.

Lu et al. (2017) extended the center folding strategy. They encoded secret symbols according to the frequency of secret symbols such that the most frequent secret symbols can be encoded to the smallest digits. The method makes effectively improve the image quality.

2.2 Reversible dual-images based hiding scheme using block folding technique

Lu et al. divided the original image into many blocks. Each block has B pixels $BK = \{BK_1, BK_2, \ldots, BK_B\}$. They divided the value range into two sections, and the secret symbols are encoded into a pair of codes $(I, \hat{d})$ according to the frequency of occurrence, where I represents the segment number of the secret message and $\hat{d}$ represent the folded value. Each block BK can be used to hide $(B - 1)$ pairs of codes $(I_1, \hat{d}_1)$、$(I_2, \hat{d}_2)$、 … 、$(I_{B-1}, \hat{d}_{B-1})$.

The scheme collects the I of each code pair and converts it into decimal digits $(ID)_{10} = (I_1, I_2, \ldots, I_{B-1})_2$. The number strategy is split into two parts to hide in the last pixel BK_B . The folded values $\hat{d}$ are concealed into pixel $BK_1, BK_2, \ldots, BK_{B-1}$ by using the same method.

The scheme of Lu and Leng (2017) can effectively encode secret symbols to smaller digits. However, each block needs an additional pixel to record the segment number that wastes the hiding space. If the block is large, then the value of ID is large that will seriously affect the image quality. Furthermore, the number of bits to be embedded in each pixel is fixed.

3. PROPOSED METHOD

Different from the previous methods, the proposed scheme does not fix the concealed secret bit for each pixel. The scheme uses the complexity of the block to decide how many bit can be embedded into the pixel. A threshold controlling strategy is used to flexibly adjust the length of the secret message. The scheme analysis the complexity of the whole cover image and divides the image into three different kinds of complexities, smooth, middle, and complexity. Two thresholds is used to split the complexity. For the smooth area, less secret message is embedded in it for minimalized the sensitive by the human eyes. For the complexity area, more message is concealed to maximum the hiding capacity.

The scheme divides the cover image into several non-overlapping blocks, and categories the block into three different zones $(\boldsymbol{Z_1}, \boldsymbol{Z_2}, \boldsymbol{Z_3})$ according to its complexity. Each zone set with different embedded bits. The secret bits are re-encoded by the center-folding strategy to further improve the image quality.

Compared with the study by Lu and Leng (2017), the proposed scheme considers the image texture features to adjust the hiding payload. More detail processes are shown following:

3.1 Pre-Embedding process

- Divides the cover image into several blocks with block size $\boldsymbol{B}$.
- Categories the blocks into three different zones, smooth, middle and complexity. Each zones has $\boldsymbol{MN\_EZ}$ blocks.
- Calculate the complexity of each block by the sum of the distance of two adjacent pixels in the block. The diagram is shown in Fig. 1. The equation is shown in Equations (1).

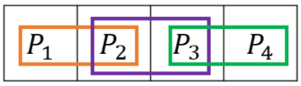


Fig. 1. The diagram of calculating the complexity of each block

$$BC = \sum_{i=1}^{B-1} |P_i - P_{i+1}|. \quad (1)$$

- Count the frequency $f(BC_i)$ of each complexity BC.
- Generate a histogram of the frequency and split the histogram into three parts by using two thresholds T_1 and T_2.

Assume that the block size is $B = 3$ and the size of a cover image is 512×512. The total number of blocks is $512 \times 512/3 = 87381$. The number of blocks in each zone is $MN\_EZ = 87381/3 = 29127$. The frequency $f(BC_i)$ of each complexity is shown in Table 1.

Table 1. The frequencies of the block complexities

BC	0	1	2	3	4	5	6
$f(BC_i)$	874	3,114	5,499	6,879	7,196	6,910	6,401
BC	7	8	9	10	11	12	…
$f(BC_i)$	5,661	4,836	4,314	3,665	3,062	2,678	…

The histogram of the frequencies is shown in Fig. 2. The scheme divides the histogram into three parts with two thresholds $T_1 = 5$ and $T_2 = 12$ such that the number of blocks in each zones are $Z_1 = 30472$, $Z_2 = 30617$ and $Z_3 = 26292$, respectively.

- The scheme conceals K_{max}, $K_{max} - 1$, and $K_{max} - 2$ bits for the complex, middle, and smooth zone, respectively. Equation (2) used to determine how many bit can be concealed in the block. Fig. 3 shows how many bits can be concealled into different type of the blocks.
- Generate three sets Zd_1, Zd_2 and Zd_3 to collect the secret symbols which belongs to three different zones.
- Generate three coding tables, which are recorded each set using frequency-based center folding strategy (Lu et al., 2017). The formulas are shown in Equations (3). In the equation, $O(d)$ is the order of the occurrence frequency of d.

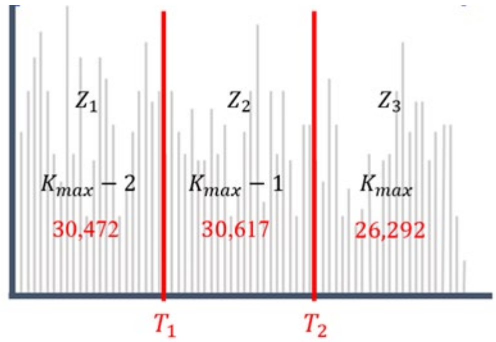


Fig. 2. Adaptive thresholds T_1 and T_2.

$$K = \begin{cases} K_{max} - 2, & if\ BC < T_1 \\ K_{max}, & if\ BC > T \\ K_{max} - 1, & otherwise. \end{cases} \quad (2)$$

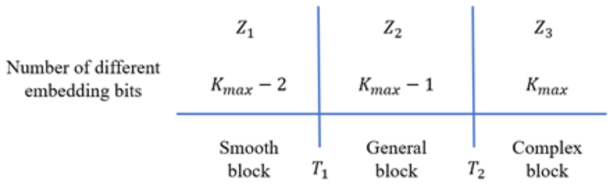


Fig. 3. The diagram of each zone with different length of secret bits

$$\hat{d}' = \begin{cases} \left\lceil \frac{O(d)}{2} \right\rceil, & if\ O(d)\ is\ an\ odd\ number, \\ -\left\lceil \frac{O(d)}{2} \right\rceil, & otherwise. \end{cases} \quad (3)$$

3.2 Embedding process

- Divides the cover image into several blocks with block size B.
- Compute the complexity of each block.
- Determine the length of the secret bit K embedded in the block by using Equation (2).
- Extract K secret bits from the secret message and converts the bit string into the decimal secret symbol d.
- Store the secret symbol d into the zone set $Zd_j = \{d_1, d_2, \dots, d_{max}\}$, where $1 \le j \le 3$ and d_{max} is the largest symbol in Zd_j.
- Obtain a new secret symbol $\hat{d}'$ according to the corresponding encoding table.
- Split the folded secret symbols $\hat{d}'$ into two parts and hides into two cover pixels. The equations are Equations (4) and (5).

$$\begin{cases} Sd_1 = \left\lfloor \frac{\hat{d}'}{2} \right\rfloor, \\ Sd_2 = \left\lceil \frac{\hat{d}'}{2} \right\rceil. \end{cases} \quad (4)$$

$$\begin{cases} P_i' = P_i + Sd_1, \\ P_i'' = P_i - Sd_2. \end{cases} \quad (5)$$

P_i' is the i^{th} stego pixel of the first image and P_i'' is the i^{th} stego pixel of the second image.

3.3 Examples of the embedding process

Fig. 4 is an example used to demonstrate the embedding processing of the proposed scheme. Suppose that the secret message is 110011111011101, block size is $B = 3$, $K_{max} = 3$, and the frequencies of the block complexities are shown in Table 1. The two thresholds $T_1 = 5$ and $T_2 = 12$, and the numbers of the embedded bits K for Z_3, Z_2, and Z_1 is 3, 2, 1, respectively. The detailed embedding steps are as follows:

- Take three original pixels as a block $\{162, 165, 170\}$.
- Calculate the complexity of the block by Equations (1), $BC = |162 - 165| + |165 - 170| = 8$. The block is belongs to Z_2, and each pixel can embed 2 bits.
- Converts the secret bit into the decimal secret symbol d to obtain three secret symbols $d = \{3, 0, 3\}$.
- Add $d = \{3, 0, 3\}$ to the set Zd_2.
- Analysis next block {140,153,152}. The complexity of the block is $BC = 14$ that is belongs to Z_3, and $K = 3$. The secret symbols $d = \{7, 3, 5\}$ are add to the set Zd_3.
- Repeat steps 2 to 4 until whole cover image is finished.
- Generate three coding tables using frequency-based central folding strategy (Lu et al., 2017) for three sets (Zd_1, Zd_2, Zd_3).

162	165	170
140	153	152

Cover Image P

162	165	170
140	154	150

Stego Image 1 P'

161	165	169
140	152	153

Stego Image 2 P''

Fig. 4. Examples of the embedding process

- Assuming the final code tables is shown in Fig. 5. The first block is belonged to Z_2, so that the secret symbols $d = \{3, 0, 3\}$ are recoded by coding table Fig. 5 (b) to obtain $\hat{d}' = \{1,0,1\}$.

d	0	1
$O(d)$	0	1
$\hat{d}'$	0	1

(a)

d	0	3	2	1
$O(d)$	0	1	2	3
$\hat{d}'$	0	1	-1	2

(b)

d	7	4	2	3	0	1	5	6
$O(d)$	0	1	2	3	4	5	6	7
$\hat{d}'$	0	1	-1	2	-2	3	-3	4

(c)

Fig. 5. Three coding tables

- Copy the block into two identical stego-blocks, and split the $\hat{d}'$ into two parts and hides in two stego-blocks {162,165,170} and{161,165,169}.
- Analysis the next block, the secret symbols $d = \{7, 3, 5\}$ are recoded by the coding table Fig. 5 (c) to obtain $\hat{d}' = \{0,2,-3\}$. Using the same method, the two stego-blocks after hiding are {140,154,150} and {140,152,153}.
- The final stego-images are shown in Fig. 4.

3.4 Extraction and Recover

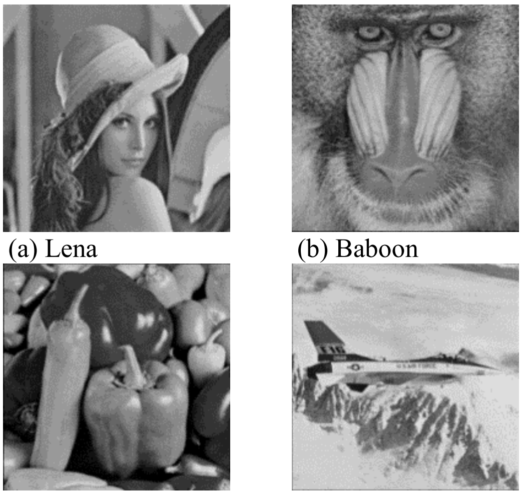
- Obtain two stego-images, block size B, T_1, T_2, the center folding table and the maximum number of hidden bits K_{max}.
- Restores the original pixel by using the equation
$$P_i = \left\lceil \frac{P_i' + P_i''}{2} \right\rceil. \quad (6)$$
- Divides the restored image into several blocks with block size B.
- Calculate the complexity BC of the block.
- Use Equation (2) to obtain the number of embedded bits K.
- Compute the distance between two stego-pixels to get the secret symbol $\hat{d}'$. The equation is
$$\hat{d}' = P_i' - P_i''. \quad (7)$$
- Map $\hat{d}'$ to the center folding table to get the corresponding d.
- Covert d into a binary secret string with K bits.
- Repeat the procedure until whole pixels have been processed.
- Obtain the original image and the secret message.

3.5 Overflow and Underflow

In embedding process, if the original pixel $P_i < \lfloor \hat{d}'_{max}/2 \rfloor$ or $P_i > 255 - \lfloor \hat{d}'_{max}/2 \rfloor$, then it will cause the overflow or underflow problem, then the pixel is non-embeddable and the scheme skips it and sets $P_i' = P_i'' = P_i$. Where $\hat{d}'_{max}$ is the maximum value in each group Zd_1, Zd_2, Zd_3. In the extraction process, if P_i' is equal to P_i'' and both of them satisfy the rule $P_i < \lfloor \hat{d}'_{max}/2 \rfloor$ or $P_i > 255 - \lfloor \hat{d}'_{max}/2 \rfloor$, then the pixel is non-embeddable.

4. EXPERIMENTAL RESULTS

In the experiment, we use four 512 × 512 gray-scale images for experiments, as shown in Fig. 6, including Lena, Baboon, Peppers, and Airplane. The experiment is mainly divided into two parts. In the first part, we set different block sizes B and maximum embedded bits K_{max} for the proposed method. In the second part, we use different settings to compare with the block folding technique (Lu and Leng, 2017).



(a) Lena (b) Baboon

(c) Peppers (d) Airplane

Fig. 6. Four 512 × 512 greyscale experimental images.

The embedding capacity and image quality are measured by Bits Per Pixel (bpp) and Peak Signal to Noise Ratio (PSNR), respectively. The formulas are

$$\text{bpp} = \frac{\text{capacity}}{2 \times h \times w}, \quad (8)$$

$$\text{MSE} = \frac{\sum_{i=1}^{h}\sum_{j=1}^{w}(P_i - P_i')^2}{h \times w}, \quad (9)$$

$$\text{PSNR} = 10 \times log_{10}\frac{255^2}{\text{MSE}} \text{ (dB)}. \quad (10)$$

In the equation, $h \times w$ is the size of the cover image and capacity is the number of bits embedded in the stego-image. P is the original image, P' is the stego-image, MSE is the mean square error between the cover image and the stego-image, and PSNR is used to judge the stego-image quality. If the PSNR value is higher than 35 dB that means image distortion is non-visible by the human eyes.

The first part is mainly for different block sizes B and the maximum number of hidden bits K_{max}. Table 2 shows the results of embedding “Peppers” in “Lena.” Stego1 and Stego2 are the PSNR value of the first and the second stego-images, and the average is the average of the two PSNR values. The experimental results show that under the same K_{max} setting, there is no significant difference in the capacity and image quality, because the block size B only affects the calculation complexity. On different K_{max}, the result will have a large change because it will directly affect the embedding capacity.

The second part, it is divided into two experiments to compare with the block folding technique (Lu and Leng, 2017).

First, we set the same block size B and different embedding bits to compare with the block folding technique and try to use different complexity of cover images. The experiment set the block size $B = 5$ and K_{max} set to 3 and 4. In block folding, we set K to 2 and 3, which is the number of hidden bits in the middle zone in the proposed method. The experimental results are shown in Table 3, where cover is the cover image and PSNR is the average of the PSNR value of the two stego-images.

Experimental results show that the proposed method has better embedding ability and image quality. Because considering the texture of the cover image, the proposed method has different effects in the different complexity of the cover image, like smooth image “Lena”, which can have better embedding ability, and in the complex image “Baboon”, we can achieve better image quality. In contrast, there is no difference in block folding technology.

Second, we set the maximum number of hidden bits K_{max} is 4 and the block size B is set to be 3 to 7. In the block folding technique (Lu and Leng, 2017), the number of hidden bits K is 3, and the number of pixels per block is set to be 3 to 7.

As shown in Fig. 7 to Fig. 9, the experimental results show that the proposed method has better embedding ability, and maintains the image quality at a certain level in the different block sizes. In contrast, as the block size increases, the image quality of the block fold technique is significantly affected, because the extra message to be recorded is very large.

5. CONCLUSIONS

In this paper, we use the complexity of the block to determine the hiding bit. More complex block owns more hiding capacity. The strategy solves Lu et al.’s scheme (Lu and Leng, 2017) which needs extra recording bit.

The proposed scheme encodes the secret symbols into smaller numbers more effectively. The experimental results show that the hiding performance of the proposed method is higher than that of the other method such as block folding technique (Lu and Leng, 2017) in terms of embedding capacity and image quality.

Lu et al., International Journal of Applied Science and Engineering, 17(3), 249–256

Table 2. Experimental results of the different maximum number of hidden bits K_{max} and different block sizes B.

B	K_{max}	Stego1	Stego2	average	Capacity	bpp
3	3	49.60	50.72	50.16	511,746	0.98
3	4	45.20	45.23	45.21	773,889	1.48
3	5	39.61	39.64	39.63	1,036,032	1.98
4	3	49.53	50.71	50.12	514,368	0.98
4	4	45.45	45.59	45.52	776,512	1.48
4	5	39.69	39.74	39.72	1,038,656	1.98
5	3	49.64	50.84	50.24	504,665	0.96
5	4	45.30	45.33	45.32	766,805	1.46
5	5	39.72	39.74	39.73	1,028,945	1.96
6	3	49.51	50.75	50.13	511,656	0.98
6	4	45.22	45.31	45.27	773,796	1.48
6	5	39.64	39.66	39.65	1,035,936	1.98
7	3	49.50	50.54	50.02	519,610	0.99
7	4	45.05	45.08	45.06	781,753	1.49
7	5	39.46	39.49	39.48	1,043,896	1.99

Table 3. Same block size B and different embedded bit compare with block folding

Method	cover	PSNR	Capacity	bpp
Propose(B=5, K_{max}=4)	Lena	45.32	766,805	1.46
Lu 2017(B=5, K=3)		43.52	632,832	1.21
Propose(B=5, K_{max}=3)	Lena	50.24	504,665	0.96
Lu 2017(B=5, K=2)		48.62	421,888	0.8
Method	**cover**	**PSNR**	**Capacity**	**bpp**
Propose(B=5, K_{max}=4)	Airplane	45.42	760,430	1.45
Lu 2017(B=5, K =3)		43.52	632,832	1.21
Propose(B=5, K_{max}=3)	Airplane	50.33	498,290	0.95
Lu 2017(B=5, K =2)		48.62	421,888	0.8
Method	**cover**	**PSNR**	**Capacity**	**bpp**
Propose(B=5, K_{max}=4)	Baboon	45.07	783,987	1.5
Lu 2017(B=5, K =3)		43.52	632,688	1.21
Propose(B=5, K_{max}=3)	Baboon	49.99	521,886	1
Lu 2017(B=5, K =2)		48.62	421,824	0.8

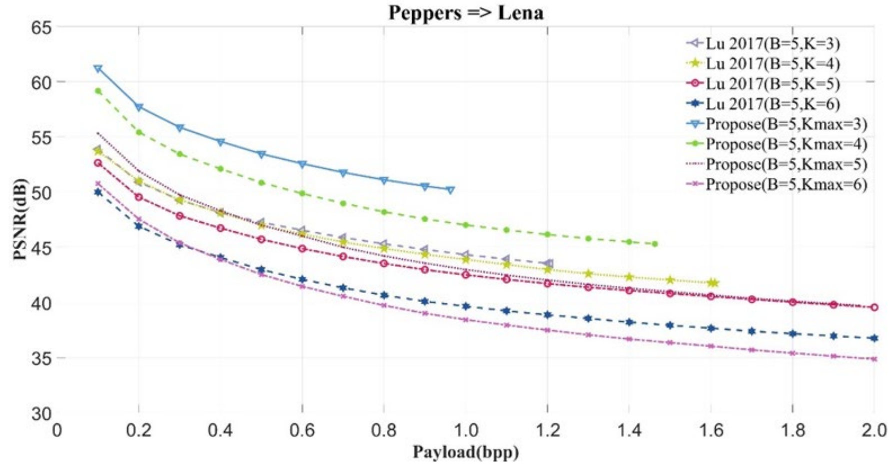


Fig. 7. The experimental results of secret message "Peppers" embedded in "Lena."

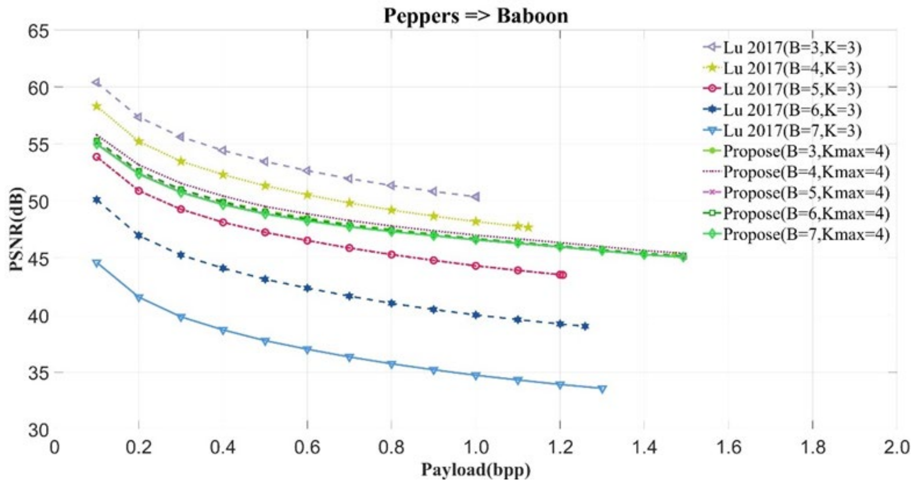


Fig. 8. The experimental results of secret message "Peppers" embedded in "Baboon."

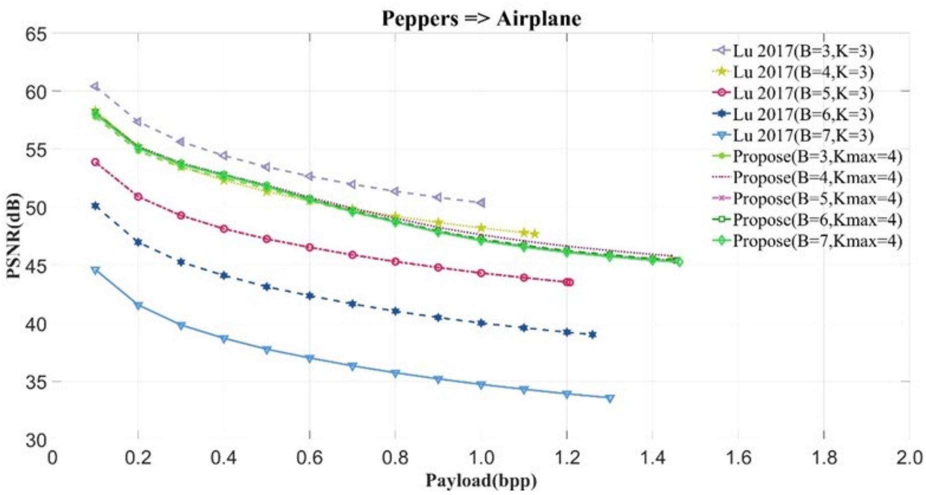


Fig. 9. The experimental results of secret message "Peppers" embedded in "Airplane.

REFERENCES

Chang, C.C., Chou, Y.C., Kieu, T.D. 2009. Information hiding in dual images with reversibility, proceedings of 2009 Third International Conference on Multimedia and Ubiquitous Engineering, 145–152.

Chang, C.C., Kieu, T.D., Chou, Y.C. 2007. Reversible data hiding scheme using two steganographic images, proceedings of TENCON 2007 - 2007 IEEE region 10 conference, 1–4.

Chang, C.C., Lu, T.C. Horng, G., Huang, Y.H., Hsu, Y.M. 2013. A high payload data embedding scheme using dual stego-images with reversibility, proceedings of 2013 9th International Conference on Information, Communications & Signal Processing, 1–5.

Lee, C.F., Huang, Y.L. 2013. Reversible data hiding scheme based on dual stegano-images using orientation combinations, Telecommunication Systems, 52, 2237–2247.

Lee, C.F., Wang, K.H., Chang, C.C., Huang, Y.L. 2009. A reversible data hiding scheme based on dual steganographic images, proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication - ICUIMC ’09, 228–237.

Lu, T.C. Tseng, C.Y., Wu, J.H. 2015. Dual imaging-based reversible hiding technique using LSB matching, Signal Processing, 108, 77–89.

Lu, T.C. Wu, J.H., Huang, C.C. 2015. Dual-image-based reversible data hiding method using center folding Technology, Signal Processing, 115, 195–213.

Lu, T.C., Chi, L.P., Wu, C.H., Chang, H.P. 2017. Reversible data hiding in dual stego-images using frequency-based encoding Technology, Multimedia Tools and Applications, 76, 23903–23929.

Lu, T.C., Leng, H.S. 2017. Reversible dual-image-based hiding scheme using block folding technique, Symmetry (Basel), 9, 10.

Qin, C., Chang, C.C., Hsu, T. 2015. Reversible data hiding scheme based on exploiting modification direction with two steganographic images, Multimedia Tools and Applications, 74, 5861–5872.

Yao, H., Qin, C., Tang, Z., Tian, Y. 2017. Improved dual-image reversible data hiding method using the selection Technology of shiftable pixels’ coordinates with minimum distortion, Signal Processing, 135, 26–35.

Yao, H., Qin, C., Tang, Z., Zhang, X. 2018. A general framework for shiftable position- based dual-image reversible data hiding, Eurasip Journal on Image and Video Processing, 1.