J. Vis. Commun. Image R. 22 (2011) 1–8

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

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

A steganographic method for digital images with four-pixel differencing and modified LSB substitution

Xin Liao [a,*], Qiao-yan Wen [a], Jie Zhang [b]

[a] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[b] School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China

ARTICLE INFO

Article history:
Received 13 December 2009
Accepted 22 August 2010
Available online 27 August 2010

Keywords:
Steganography
Four-pixel differencing
Modified LSB substitution
Data hiding
Pixel-value differencing
Error block
Embedding capacity
Imperceptible

ABSTRACT

To improve the embedding capacity and provide an imperceptible visual quality, a novel steganographic method based on four-pixel differencing and modified least significant bit (LSB) substitution is presented. The average difference value of a four-pixel block is exploited to classify the block as a smooth area or an edge area. Secret data are hidden into each pixel by the k-bit modified LSB substitution method, where k is decided by the level which the average difference value falls into. Readjustment will be executed to guarantee the same level that the average difference value belongs to before and after embedding, and to minimize the perceptual distortion. By proving that the readjusting procedure works, a theoretical proof is given to justify our method succeeded in embedding and extracting. Our experimental results have shown that the proposed method not only has an acceptable image quality but also provides a large embedding capacity.

1. Introduction

Internet is a popular communication channel nowadays. Transmitted data are easy to be copied or destroyed by unauthorized persons. Therefore, how to transmit data secretly by internet becomes an important topic. Encryption may provide a safe way, which transforms data into a ciphertext via cipher algorithms [1]. However, it makes the messages unreadable and suspicious enough to attract eavesdroppers' attention. To overcome this problem, steganography offers different approaches to transmitting secret messages [2,3]. Steganography is a technique that imperceptibly hides secret data into cover media by altering its most insignificant components for covert communication, such that an unauthorized user will not be aware of the existence of secret data [4].

The most common and well-known steganographic method is called least significant bit (LSB) substitution, which embeds secret data by replacing k LSBs of a pixel with k secret bits directly [5]. Many optimized LSB methods have been proposed to improve this work [6–8]. The human perceptibility has a property that it is sensitive to some changes in the pixels of the smooth areas, while it is not sensitive to changes in the edge areas. Not all pixels in a cover image can tolerate equal amount of changes without causing noticeable distortion. Hence, to improve the quality of stego images, several adaptive methods have been proposed in which the amount of bits to be embedded in each pixel is variable [9–16]. In 2003, Wu and Tsai proposed a novel steganographic method that uses the difference value between two neighboring pixels to determine how many secret bits should be embedded [9]. Chang and Tseng proposed a side match approach to embed secret data, where the number of bits to be embedded in a pixel is decided by the difference between the pixel and its upper and left side pixels [10]. In 2005, Wu et al. presented a novel steganographic method, which combined pixel-value differencing and LSB substitution [11]. Park et al. proposed a new method based on the difference value between two pixels adjacent to the target pixel [12]. In 2006, Yang and Weng proposed a multi-pixel differencing method that uses three difference values in a four-pixel block to determine how many secret bits should be embedded [13]. The method in [14] provided the combination of multi-pixel differencing and LSB substitution to improve the work in [13], but the embedding capacity is far less than that of Wu et al.'s method. In 2008, Wang et al. presented a steganographic method that utilizes the remainder of two consecutive pixels to record the information of secret data [15]. Yang et al. proposed an adaptive LSB steganographic method using the difference value of two consecutive pixels to distinguish between edge areas and smooth areas [16]. All pixels are embedded by the k-bit modified LSB substitution method, where k is decided by the range which the difference value belongs to [16].

However, some of them seem not to consider the features of edge sufficiently [9,11,15,16]. The methods in [10,12] have overcome the drawback, but unfortunately they would result in the

* Corresponding author.
E-mail address: liaoxinbupt@gmail.com (X. Liao).

1047-3203/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2010.08.007

propagated error and lower embedding capacity. In order to provide better stego-image quality and larger embedding capacity, a novel steganographic method improving the multi-pixel differencing based on modified LSB substitution is presented in this paper. Similar to [13,14], a four-pixel block with three difference values is sufficiently considered. The average value of three difference values is exploited to distinguish between edge areas and smooth areas, and to estimate how many secret bits will be embedded into the block. Embed secret bits into each pixel in the block by modified LSB substitution method. Readjustment will be executed to extract secret data exactly and to minimize the perceptual distortion resulted from embedding. The experimental results show that our proposed method provides a large embedding capacity, and the quality of the stego image is improved as well.

The remainder of this paper is organized as follows. In Section 2, the embedding and extracting algorithms of the proposed method is presented. In the next section, we show that the proposed method succeed in embedding and extracting by proving that the readjusting procedure works. The experimental results will be in Section 4. Finally, conclusions are given in Section 5.

2. The proposed method

The proposed method conforms to the issues that are mentioned before. The pixels in edge areas can tolerate much more changes without making perceptible distortion than smooth areas. The range of average difference value is partitioned into two different levels, low level and high level. The division of smooth and edge areas is predefined by users. Pixels located in the block are embedded by the k-bit modified LSB substitution method, where k is decided by the level which the average difference value belongs to. Low level will use a lower value k_l, while high level uses k_h. The perceptual distortion can be minimized by readjustment, which, at the same time, guarantees the same level that the average difference value belongs to before and after embedding. The embedding and extracting algorithms are presented in the following subsections.

The concept of modified LSB substitution is to increase or decrease the most significant bit (MSB) part by 1 in order to improve the image quality [17,18]. For instance, secret data is $m = 000_{(2)}$, then a pixel $p = 1100111_{(2)}$ is embedded by the 3-bit common LSB substitution method and have a result $p' = 1100000$. The MSB part of p' is increased by 1, so the result of modified LSB substitution method is $p' = 1101000_{(2)}$, reducing the difference between p and p'.

2.1. The embedding algorithm

All the pixels in the cover image are 256 gray values. The cover image is partitioned into non-overlapping four-pixel blocks. For each block, there are four neighboring pixels $p_{i,j}$, $p_{i,j+1}$, $p_{i+1,j}$, $p_{i+1,j+1}$, and their corresponding gray values are y_0, y_1, y_2 and y_3, respectively. The detailed embedding steps are as follows.

Step 1: Calculate the average difference value D, which is given by

$$D = \frac{1}{3}\sum_{i=0}^{3}(y_i - y_{min}) \qquad y_{min} = min\{y_0, y_1, y_2, y_3\} \tag{1}$$

Step 2: Our proposed method adaptively embeds messages using two levels (lower-level and higher-level), and threshold value T is used to partition the range of D into two levels. If $D \leqslant T$, D belongs to "lower-level" (i.e., the block belongs to a smooth area), then $k = k_l$. Otherwise, D belongs to "higher-level" (i.e., the block belongs to an edge area), then $k = k_h$. In order to succeed in the readjusting procedure, we apply the restrictions $2^{k_l} \leqslant T \leqslant 2^{k_h}$ and $1 \leqslant k_l, k_h \leqslant 5$.

Step 3: Verify whether the block belongs to "Error Block". If not, continue to next step. Otherwise, restart from Step 1.

Definition 1. Assume $y_{max} = max\{y_0,y_1,y_2,y_3\}$, the block is called "Error Block" if and only if $D \leqslant T$ and $y_{max} - y_{min} > 2T + 2$.

For instance, assume $T = 6$. A block with four neighboring pixel values $(139, 140, 140, 154)$ belongs to "Error Block", because $D = \frac{17}{3} < 6$ and $154 - 139 = 15 > 2 \times 6 + 2 = 14$. "Error Block" is *NOT* used to embed secret bits, which will be explained in Section 3.

Step 4: Convert y_i to be y'_i by the k-bit common LSB substitution method $(0 \leqslant i \leqslant 3)$, respectively.

Step 5: Apply the k-bit modified LSB substitution method to y'_i, and let y''_i be the result $(0 \leqslant i \leqslant 3)$, respectively.

Step 6: This step is called "readjusting procedure". Let $\widehat{y_i} = y''_i + l \times 2^k, 0 \leqslant i \leqslant 3, l \in \{0, 1, -1\}$, and search $(\widehat{y_0}, \widehat{y_1}, \widehat{y_2}, \widehat{y_3})$ such that
(1) $\widehat{D}$ and D belong to the same level, where $\widehat{D} = \frac{1}{3}\sum_{i=0}^{3}(\widehat{y_i} - \hat{y}_{min})$, $\hat{y}_{min} = min\{\widehat{y_0}, \widehat{y_1}, \widehat{y_2}, \widehat{y_3}\}$.
(2) The final stego block $(\widehat{y_0}, \widehat{y_1}, \widehat{y_2}, \widehat{y_3})$ does not belong to "Error Block".
(3) The value of $\sum_{i=0}^{3}(\widehat{y_i} - y_i)^2$ is minimized.

After the replacement of (y_0,y_1,y_2,y_3) by $(\widehat{y_0}, \widehat{y_1}, \widehat{y_2}, \widehat{y_3})$ in the block, the purpose of $4k$-bit secret data hiding have been achieved. Repeat Steps 1–6 until all the secret data are embedded in the cover image, and the stego image is obtained.

For example, suppose we have a block with four neighboring pixel values $(139, 146, 137, 142)$, and the secret data are 000111111101. Assume $T = 5$, $k_l = 2$ and $k_h = 3$. Calculate the average difference value $D = \frac{16}{3} > T = 5$, then $y_i (0 \leqslant i \leqslant 3)$ are embedded by the 3-bit common LSB substitution method at first, $y'_0 = 136, y'_1 = 151, y'_2 = 143$ and $y'_3 = 141$. After applying the 3-bit modified LSB substitution method, $y''_0 = 136$, $y''_1 = 143$, $y''_2 = 135$ and $y''_3 = 141$. Readjustment is executed resulting in $\widehat{y_0} = 136$, $\widehat{y_1} = 151$, $\widehat{y_2} = 135$ and $\widehat{y_3} = 141$.

2.2. The extracting algorithm

In the extraction process, we can quickly extract secret data without the original image. Partition the stego image into four-pixel blocks, which is identical with the embedding algorithm. For each block $(p_{i,j}, p_{i,j+1}, p_{i+1,j}, p_{i+1,j+1})$, the following steps are executed to extract the secret data.

Step 1: Calculate the average difference value D by Eq. (1).

Step 2: Use the threshold value T to find out the level which D belongs to. If D belongs to the "lower-level", $k = k_l$, otherwise $k = k_h$.

Step 3: Verify whether the block belongs to "Error Block". If not, extract $4k$-bit secret data from the k-bit LSB of $y_i (0 \leqslant i \leqslant 3)$. Otherwise, restart from Step 1.
For instance, we extract the embedding example $(136, 151, 135, 141)$, which is shown in the above subsection. Since $T = 5$, $D = \frac{23}{3}$ belongs to "higher-level", then $k = k_h = 3$. We extract 3-bit LSB of $y_i (0 \leqslant i \leqslant 3)$, respectively. Secret data 000111111101 can be obtained.

3. Theoretic analyses and discussions

Secret data can be directly extracted as the least k bits of the pixel values, because Step 4 and Step 5 in the embedding algorithm do not change the embedded data. When applying the k-bit modified LSB substitution, pixel values may be modified at distance 2^k by increasing or decreasing the most significant bit part. The readjusting phase works by modifying the pixel value p as $p+2^k$ or $p-2^k$. Both of them do not affect the least k bits of the pixel values.

In the proposed method, adjustment procedure is the more important technique for ensuring that secret data can be extracted successfully. Some restrictions have been applied to ensure the correctness of the method. We will give a theoretical proof to explain or justify our strategy.

Lemma 1 [16]. *For the k-bit modified LSB substitution, if pixel value y_i dose not belong to the rang $[0,2^{k-1}]$ or $[255-2^{k-1},255]$, y_i is modifiable and the resulting value y_i'' such that $y_i-2^{k-1}\leqslant y_i''\leqslant y_i+2^{k-1}$.*

Lemma 2 [16]. *For the k-bit modified LSB substitution, if y_i belongs to the rang $[0,2^{k-1}]$, the embedded result has $y_i-2^{k-1}\leqslant y_i''\leqslant y_i+(2^k-1)$ and y_i'' belongs to the rang $[0,2^k-1]$; if y_i belongs to the rang $[255-2^{k-1},255]$, the embedded result has $y_i-(2^k-1)\leqslant y_i''\leqslant y_i+2^{k-1}$ and y_i'' belongs to the rang $[255-(2^k-1),255]$.*

Lemma 3 [16]. *Suppose that only one of p_1 and p_2 is not modifiable. For the k-bit modified LSB substitution, then $|p_1-p_2|-2^k-2^{k-1}+1\leqslant|p_1''-p_2''|\leqslant|p_1-p_2|+2^k$.*

Lemma 4. *Suppose that two pixels p_1 and p_2 belong to the range $[0,2^{k-1}]$. For the k-bit modified LSB substitution, then $0\leqslant|p_1''-p_2''|<2^k$.*

Proof. From Lemma 2, we have $0\leqslant p_1''\leqslant 2^k-1$ and $0\leqslant p_2''\leqslant 2^k-1$, so $0\leqslant|p_1''-p_2''|\leqslant 2^k-1<2^k$. □

Theorem 1. *Suppose that four pixel y_0, y_1, y_2 and y_3 are modifiable. For the k-bit modified LSB substitution, we have $|D''-D|\leqslant 2^k$, where $D''=\frac{1}{3}\sum_{i=0}^{3}(y_i''-y_{min}'')$, $y_{min}''=min\{y_0'',y_1'',y_2'',y_3''\}$*

Proof. Without loss of generality, let $y_0\leqslant y_1\leqslant y_2\leqslant y_3$, then $D=\frac{1}{3}(y_1+y_2+y_3-3y_0)$. If $y_0''=min\{y_0'',y_1'',y_2'',y_3''\}$, we have

$$\begin{aligned}|D''-D|&=\left|\frac{1}{3}(y_1''+y_2''+y_3''-3y_0'')-\frac{1}{3}(y_1+y_2+y_3-3y_0)\right|\\&=\left|\frac{1}{3}(y_1''-y_1)+\frac{1}{3}(y_2''-y_2)+\frac{1}{3}(y_3''-y_3)-(y_0''-y_0)\right|\\&\leqslant\frac{1}{3}|(y_1''-y_1)|+\frac{1}{3}|(y_2''-y_2)|+\frac{1}{3}|(y_3''-y_3)|\\&\quad+|(y_0''-y_0)|\\&\leqslant\frac{1}{3}\times 2^{k-1}+\frac{1}{3}\times 2^{k-1}+\frac{1}{3}\times 2^{k-1}+2^{k-1}=2^k\end{aligned}\tag{2}$$

If $y_0''\neq min\{y_0'',y_1'',y_2'',y_3''\}$, let's assume $y_1''=min\{y_0'',y_1'',y_2'',y_3''\}$. We know that

$$\begin{aligned}|D''-D|&=\left|\frac{1}{3}(y_0''+y_2''+y_3''-3y_1'')-\frac{1}{3}(y_1+y_2+y_3-3y_0)\right|\\&=\left|\frac{1}{3}(y_2''-y_2)+\frac{1}{3}(y_3''-y_3)+\frac{1}{3}(y_0''+3y_0)-\frac{1}{3}(3y_1''+y_1)\right|\\&\leqslant\frac{1}{3}|(y_2''-y_2)|+\frac{1}{3}|(y_3''-y_3)|\\&\quad+\frac{1}{3}|y_0''+3y_0-3y_1''-y_1|\end{aligned}\tag{3}$$

Assume $y_0''-y_0=\triangle y_0, y_1''-y_1=\triangle y_1$, we have $y_0\leqslant y_1, y_0''\geqslant y_1''$. Then, $0\leqslant y_1-y_0\leqslant\triangle y_0-\triangle y_1$.

$$\begin{aligned}y_0''+3y_0-3y_1''-y_1&=(y_0+\triangle y_0)+3y_0-3(y_1+\triangle y_1)-y_1\\&=-4(y_1-y_0)+\triangle y_0-3\triangle y_1\leqslant\triangle y_0-3\triangle y_1\\&\leqslant 2^{k+1}\end{aligned}$$

$$\begin{aligned}y_0''+3y_0-3y_1''-y_1&=-4(y_1-y_0)+\triangle y_0-3\triangle y_1\\&\geqslant-4(\triangle y_0-\triangle y_1)+\triangle y_0-3\triangle y_1\\&\geqslant-3\triangle y_0+\triangle y_1\geqslant-2^{k+1}\end{aligned}$$

Thus, we have $|y_0''+3y_0-3y_1''-y_1|\leqslant 2^{k+1}$. From Eq. (3) we know that $|D''-D|\leqslant\frac{1}{3}\times 2^{k-1}+\frac{1}{3}\times 2^{k-1}+\frac{1}{3}\times 2^{k+1}=2^k$. □

Theorem 2. *Suppose that four pixels of the block are modifiable and the cover block dose not belong to "Error Block". Then, the readjusting phase can work successfully when $2^{k_l}\leqslant T\leqslant 2^{k_h}$ and $1\leqslant k_l$, $k_h\leqslant 5$.*

Proof. There are eighty-one readjusting choices ($3^4=81$) in Step 6, and we will show that there exists at least one choice such that the former two conditions hold in Step 6; namely, (1) $\widehat{D}$ and D belong to the same level; (2) The final stego block does not belong to "Error Block". Then, there must exist the best one in the finite choices such that three conditions hold in Step 6. Thus, the readjusting phase works successfully.

Without loss of generality, suppose that the resulting pixel values are $w\leqslant z\leqslant y\leqslant x$ after the k-bit modified LSB substitution. From Theorem 1, we have $D''=\frac{1}{3}(x+y+z-3w)$ and $|D''-D|\leqslant 2^k$. Now we discuss the cases that D belongs to different levels, which can be divided into two categories:

(1) *Category A: $D>T$* belongs to high level. $k=k_h$.
Case 1: If $D''>T$, there already exists a choice (x,y,z,w) such that the former two conditions hold in Step 6.
Case 2: If $D''\leqslant T$, a choice $(x,y,z,w-2^k)$ such that $\widehat{D}=D''+2^k=D''+2^{k_h}\geqslant T$. Decrease of w by 2^k may not be allowed if the pixel value $w<2^k$. However, we can find another choice $(x+2^k,y+2^k,z+2^k,w)$ such that $\widehat{D}=D''+2^{k_h}\geqslant T$ if $z\leqslant y\leqslant x\leqslant 255-2^k$. Obviously, these choices don't belong to "Error Block".

(2) *Category B: $D\leqslant T$* belongs to low level. $k=k_l$. The cover block does not belong to "Error block", so we have $y_{max}-y_{min}\leqslant 2T+2$, then $x-w\leqslant(y_{max}+2^{k-1})-(y_{min}-2^{k-1})\leqslant 2T+2+2^k$.
Case 1: $D''\leqslant T$. If (x,y,z,w) does not belong to "Error block", it satisfies the former two conditions in Step 6. Otherwise, $x-w>2\times T+2$, we can find another choice $(x-2^k,y,z,w)$ such that the former two conditions hold.
Case 2: $D''>T$. We divide it into 10 subcases according to the relationship among four pixels x, y, z, and w, as follows:

subcase 1: $x\geqslant y\geqslant z\geqslant w+2^k\times 2>w$. Fig. 1(a) shows the possible location. There exists a choice $(x,y,z,w+2^k)$ such that $\widehat{D}=D''-2^k\leqslant D\leqslant T$ and $x-(w+2^k)\leqslant 2T+2$.
subcase 2: $x\geqslant y\geqslant w+2^k\times 2>z\geqslant w+2^k>w$. Fig. 1(b) shows the possible location. There exists a choice $(x,y,z,w+2^k)$ such that $\widehat{D}=D''-2^k\leqslant D\leqslant T$ and $x-(w+2^k)\leqslant 2T+2$.
subcase 3: $x\geqslant y\geqslant w+2^k\times 2>w+2^k>z\geqslant w$. Fig. 1(c) shows the possible location. There exists a choice $(x,y-2^k,z+2^k,w+2^k)$ such that $\widehat{D}=D''-2^k\leqslant D\leqslant T$ and $x-(w+2^k)\leqslant 2T+2$.
subcase 4: $x\geqslant w+2^k\times 2>y\geqslant z\geqslant w+2^k>w$. Fig. 1(d) shows the possible location. There exists a choice

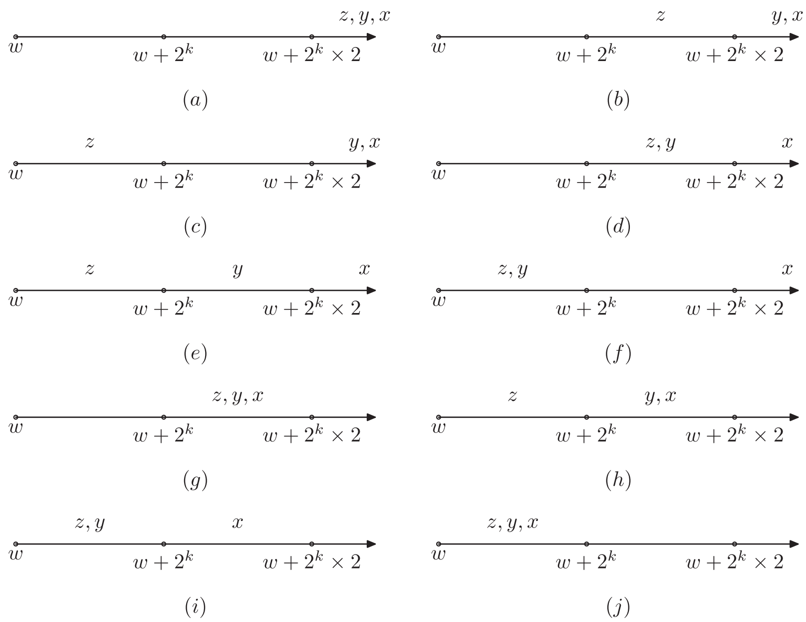

Fig. 1. 10 subcases about the possible locations of x, y, z, and w.

$(x,y,z,w+2^k)$ such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x-(w+2^k) \leqslant 2T+2$.

subcase 5: $x \geqslant w+2^k \times 2 > y \geqslant w+2^k > z \geqslant w$. Fig. 1(e) shows the possible location. There exists a choice $(x-2^k, y, z+2^k, w+2^k)$ such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $max\{x-2^k, y, z+2^k\} - (w+2^k) \leqslant 2T+2$.

subcase 6: $x \geqslant w+2^k \times 2 > w+2^k > y \geqslant z \geqslant w$. Fig. 1(f) shows the possible location. There exists a choice $(x-2^k, y+2^k, z+2^k, w+2^k)$ such that

$$\begin{aligned}
\widehat{D} &= \frac{1}{3} \times \Big[(x-2^k-(w+2^k)) + (y+2^k-(w+2^k)) \\
&\quad + (z+2^k-(w+2^k))\Big] = \frac{1}{3} \times [(x-w)+(y-w)+(z-w)] \\
&\quad - \frac{2}{3} \times 2^k \leqslant \frac{1}{3} \times \Big[(2T+2+2^k)+(2^k-1)+(2^k-1)\Big] \\
&\quad - \frac{2}{3} \times 2^k = \frac{1}{3} \times (2T+2^k) \leqslant T
\end{aligned}$$

and $max\{x-2^k, y+2^k, z+2^k\} - (w+2^k) \leqslant 2T+2$.

subcase 7: $w+2^k \times 2 > x \geqslant y \geqslant z \geqslant w+2^k > w$. Fig. 1(g) shows the possible location. There exists a choice $(x,y,z,w+2^k)$ such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x-(w+2^k) \leqslant 2T+2$.

subcase 8: $w+2^k \times 2 > x \geqslant y \geqslant w+2^k > z \geqslant w$. Fig. 1(h) shows the possible location. There exists a choice $(x-2^k, y-2^k, z, w)$ such that $\widehat{D} = D'' - \frac{2}{3} \times 2^k < \frac{1}{3} \times (2 \times 2^k + 2 \times 2^k + 2^k) - \frac{2}{3} \times 2^k = 2^k \leqslant T$ and $max\{x-2^k, z\} - w \leqslant 2T+2$.

subcase 9: $w+2^k \times 2 > x \geqslant w+2^k > y \geqslant z \geqslant w$. Fig. 1(i) shows the possible location. There exists a choice $(x-2^k, y, z, w)$ such that $\widehat{D} = D'' - \frac{1}{3} \times 2^k < \frac{1}{3} \times (2 \times 2^k + 2^k + 2^k) - \frac{1}{3} \times 2^k = 2^k \leqslant T$ and $max\{x-2^k, y, z\} - w \leqslant 2T+2$.

subcase 10: $w+2^k > x \geqslant y \geqslant z \geqslant w$. Fig. 1(j) shows the possible location. The subcase 10 does not belong to Case 2 because $D'' < \frac{1}{3} \times (2^k+2^k+2^k) = 2^k \leqslant T$, which contradicts the assumption imposed for Case 2. □

Theorem 2 has shown that our method is correct if all the four pixels are modifiable. Now, we discuss the cases that at least one of the four pixels is not modifiable.

Lemma 5. *If there exist a pixel belongs to the range $[0, 2^{k-1}]$ and another pixel belongs to the range $[255-2^{k-1}, 255]$, there already exists a choice such that the former two conditions hold in Step 6.*

Proof. Suppose the pixels $0 \leqslant p_1 \leqslant 2^{k-1}$ and $255 - 2^{k-1} \leqslant p_2 \leqslant 255$, then $D \geqslant \frac{1}{3} \times (255 - 2^{k-1} - 2^{k-1}) > 2^{k_h} \geqslant T$ and D belongs to high level. From Lemma 2, we have $0 \leqslant p_1'' \leqslant 2^k - 1$ and $255 - (2^k - 1) \leqslant p_2'' \leqslant 255$. $D'' \geqslant \frac{1}{3} \times [255 - (2^k-1) - (2^k-1)] > 2^{k_h} \geqslant T$ and D'' also belongs to high level. Obviously, the choice does not belong to "Error Block". □

Theorem 3. *Suppose that not all the four pixels of the block are modifiable and the cover block dose not belong to "Error Block". Then, the readjusting phase can work successfully when $2^{k_l} \leqslant T \leqslant 2^{k_h}$ and $1 \leqslant k_l, k_h \leqslant 5$.*

Proof. Similar to Theorem 2, we will show that there exists at least one choice such that the former two conditions hold in Step 6. It can be divided into three cases: (1) Only some pixels belong to $[0, 2^{k-1}]$; (2) Only some pixels belong to $[255-2^{k-1}, 255]$; (3) Some pixels belong to $[0, 2^{k-1}]$, and some belong to $[255-2^{k-1}, 255]$. We only need to discuss case 1, because the proof of case 2 is similar and case 3 is solved by Lemma 5. We divide case 1 into two categories:

(1) *Category A: $D > T$* belongs to high level. No matter how many pixels are not modifiable in the four-pixel block, the proof in Theorem 2 is still efficient and correct.
(2) *Category B: $D \leqslant T$* belongs to low level.

Case 1: $D'' \leqslant T$. The cover block is not an error block, so we have $y_{max} - y_{min} \leqslant 2T+2$. If (x,y,z,w) does not belong to "Error block", it satisfies the former two conditions in Step 6. Otherwise, $x - w > 2 \times T + 2$, we have to consider whether y_{max} is modifiable or not. If y_{max} is modifiable, then $x - w \leqslant (y_{max} + 2^{k-1}) - (y_{min} - 2^{k-1}) \leqslant 2T + 2 + 2^k$, so we can find another choice $(x-2^k, y, z, w)$ such that the former two conditions hold in Step 6. If y_{max} is not modifiable, i.e., y_{max} belongs to $[0, 2^{k-1}]$. Thus, all the cover four pixels are not modifiable.

From Lemma 4 we know that $x - w \leqslant 2^k < 2T + 2$, there exists a choice (x,y,z,w) such that the former two conditions hold.

Case 2: $D'' > T$. We also divide it into 10 subcases according to the relationship among four pixels x, y, z, and w.

Only some pixels in the four-pixel block belong to $[0, 2^{k-1}]$ and there are four conditions:

Condition 1: Only one pixel y_0 belongs to $[0, 2^{k-1}]$, and others are modifiable. All the subcases in Fig. 1 are possible and $y_0 = min\{y_0, y_1, y_2, y_3\}$. The block is not an error block, so we have $y_{max} - y_{min} \leqslant 2T + 2$, then $x - w \leqslant (y_{max} + 2^{k-1}) - (y_0 - 2^{k-1}) \leqslant 2T + 2 + 2^k$.

For the subcases 1, 2, 4, 7: From Lemma 2, we have $0 \leqslant y''_0 \leqslant 2^k - 1$. Thus, $y_0 = w$. From Lemma 3, we know that $D'' - D = \frac{1}{3}[(|y''_1 - y''_0| - |y_1 - y_0|) + (|y''_2 - y''_0| - |y_2 - y_0|) + (|y''_3 - y''_0| - |y_3 - y_0|)] \leqslant 2^k$. Then $(x,y,z,w+2^k)$ is a choice such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x - (w + 2^k) \leqslant 2T + 2$.

For the subcases 3, 5: If $y''_0 = w$, we also have $D'' - D \leqslant 2^k$ and $(x,y,z,w+2^k)$ is a choice such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x - z \leqslant 2T + 2$; If $y''_0 \neq w$, we know that $y''_0 = z$ according to Lemma 2. Suppose $y''_1 = w$, we have

$$\begin{aligned}
D'' - D &= \frac{1}{3}(y''_0 + y''_2 + y''_3 - 3y''_1) - \frac{1}{3}(y_1 + y_2 + y_3 - 3y_0) \\
&= \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) + \frac{1}{3}(y''_0 + 3y_0) - \frac{1}{3}(3y''_1 + y_1) \\
&= \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) \\
&\quad + \frac{1}{3}[(y''_0 - y''_1) + 2(y_1 - y''_1) - 3(y_1 - y_0)] \\
&\leqslant \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) + \frac{1}{3}[(y''_0 - y''_1) + 2(y_1 - y''_1)] \\
&= \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) + \frac{1}{3}[(z - w) + 2(y_1 - y''_1)] \\
&\leqslant \frac{1}{3} \times 2^{k-1} + \frac{1}{3} \times 2^{k-1} + \frac{1}{3} \times (2^k + 2 \times 2^{k-1}) = 2^k
\end{aligned}$$

$(x,y,z,w+2^k)$ is a choice such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x - z \leqslant 2T + 2$.

For the subcases 6, 8, 9, 10: The proof in Theorem 2 is still efficient and correct.

Condition 2: Two pixels y_0 and y_1 belong to $[0, 2^{k-1}]$, and others are modifiable. From Lemma 2, we know that $0 \leqslant y''_0, y''_1 \leqslant 2^k - 1$, so the subcases 1, 2, 4, 7 are impossible.

For the subcases 3, 5: Because $0 \leqslant y''_0, y''_1 \leqslant 2^k - 1$, $y''_0 = w$ or $y''_1 = w$. If $y''_0 = w$, from Lemmas 3 and 4 we have $D'' - D = \frac{1}{3}[(|y''_1 - y''_0| - |y_1 - y_0|) + (|y''_2 - y''_0| - |y_2 - y_0|) + (|y''_3 - y''_0| - |y_3 - y_0|)] \leqslant 2^k$. Then $(x,y,z,w+2^k)$ is a choice such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x - z \leqslant 2T + 2$; If $y''_1 = w$, we have

$$\begin{aligned}
D'' - D &= \frac{1}{3}(y''_0 + y''_2 + y''_3 - 3y''_1) - \frac{1}{3}(y_1 + y_2 + y_3 - 3y_0) \\
&= \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) \\
&\quad + \frac{1}{3}[(y''_0 - y''_1) + 2(y_1 - y''_1) - 3(y_1 - y_0)] \\
&\leqslant \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) + \frac{1}{3}[(y''_0 - y''_1) + 2(y_1 - y''_1)] \\
&= \frac{1}{3}(y''_2 - y_2) + \frac{1}{3}(y''_3 - y_3) + \frac{1}{3}[(z - w) + 2(y_1 - y''_1)] \\
&\leqslant \frac{1}{3} \times 2^{k-1} + \frac{1}{3} \times 2^{k-1} + \frac{1}{3} \times (2^k + 2 \times 2^{k-1}) = 2^k
\end{aligned}$$

$(x,y,z,w+2^k)$ is a choice such that $\widehat{D} = D'' - 2^k \leqslant D \leqslant T$ and $x - z \leqslant 2T + 2$.

For the subcases 6, 8, 9, 10: The proof in Theorem 2 is still efficient and correct.

Condition 3: Three pixels y_0, y_1 and y_2 belong to $[0, 2^{k-1}]$, and y_3 is modifiable. From Lemma 2, we know that since $0 \leqslant y''_0, y''_1, y''_2 \leqslant 2^k - 1$, only the subcases 6, 9, 10 are possible. The proof in Theorem 2 is still efficient and correct.

Condition 4: All the four pixels belong to $[0, 2^{k-1}]$. From Lemma 2, we know that since $0 \leqslant y''_0, y''_1, y''_2, y''_3 \leqslant 2^k - 1$, only the subcase 10 is possible. The proof in Theorem 2 is still efficient and correct. □

From Theorems 2 and 3, we have:

Theorem 4. *Suppose that four pixel values of the block are $y_i (0 \leqslant i \leqslant 3)$ and the cover block dose not belong to "Error Block". Then, the readjusting phase can work successfully when $2^{k_l} \leqslant T \leqslant 2^{k_h}$ and $1 \leqslant k_l, k_h \leqslant 5$.*

Here, we will give an example to show that the readjusting phase cannot work if the cover block belongs to "Error Block", i.e., $D \leqslant T$ and $y_{max} - y_{min} > 2T + 2$, here $y_{max} = max\{y_0, y_1, y_2, y_3\}$, $y_{min} = min\{y_0, y_1, y_2, y_3\}$ and $D = \frac{1}{3}\sum_{i=0}^{3}(y_i - y_{min})$. Suppose that we have a block with four pixel values (66, 66, 66, 81), and the secret data are 11001011. Assume $T = 5$, $k_l = 2$, and $k_h = 3$. Calculate the average difference value $D = \frac{15}{3} \leqslant T$, then $y_i (0 \leqslant i \leqslant 3)$ are embedded by the 2-bit common LSB substitution method at first, $y'_0 = 67$, $y'_1 = 64$, $y'_2 = 66$ and $y'_3 = 83$. After applying the 2-bit modified LSB substitution method, $y''_0 = 67$, $y''_1 = 64$, $y''_2 = 66$ and $y''_3 = 83$. There are 81 readjusting choice in Step 6, but the minimum of average difference values is $\widehat{D} = \frac{16}{3} > T$. The readjusting phase cannot work.

A good question would be: How many error blocks dose a digital image have? Table 1 shows the numbers *Num* and proportions *Pe* of error blocks that ten digital images have when $T = 2$ (lower boundary), $T = 15$ (middle value) and $T = 32$ (upper boundary). For an $M \times N$ grayscale image, $Pe = \frac{Num}{(M \times N)/4}$ and is shown in Fig. 2 for various threshold values T. These are significantly few error blocks in a cover image. So it will have a negligible effect on the capacity of our method, which can be almost ignored.

4. Experimental results

Several experiments are preformed to evaluate our proposed method. Ten grayscale images with size 512 × 512 are used in the experiments as cover images, and three of them are shown in Fig. 3. A series of pseudo-random numbers as the secret bit streams are embedded into the cover images. The peak signal to noise ratio (PSNR) is utilized to evaluate the quality of the stego image. For an $M \times N$ grayscale image, the PSNR value is defined as follows:

$$\mathrm{PSNR} = 10 \times \log_{10} \frac{255 \times 255 \times M \times N}{\sum_{i=1}^{M}\sum_{j=1}^{N}(p_{i,j} - q_{i,j})^2} (dB) \tag{4}$$

where $p_{i,j}$ and $q_{i,j}$ denote the pixel values in row i and column j of the cover image and the stego image, respectively.

Stego images created by our proposed method with various values of k_l and k_h are shown in Figs. 4–6. As the figures show, distortions resulted from embedding are imperceptible to human vision.

We have experimented using a series of "$k_l - k_h$" division with various threshold values. For instance, 2–3 division with $T = 7$ means that four-pixel block with average difference value falling into the low level and high level, will be embedded by the 2-bit and 3-bit modified LSB substitution method, respectively. Table 2

Table 1
The numbers and proportions of error blocks in ten digital images.

	Elaine	Lena	Baboon	Peppers	Toys	Girl	Gold	Brab	Zelda	Tiffany	Average
Num($T = 2$)	0	0	0	0	0	0	0	0	0	0	0
Pe($T = 2$)	0	0	0	0	0	0	0	0	0	0	0
Num($T = 15$)	32	12	130	15	12	1	29	84	1	25	34
Pe($T = 15$)	$\frac{4.88}{10^4}$	$\frac{1.83}{10^4}$	$\frac{19.84}{10^4}$	$\frac{2.29}{10^4}$	$\frac{1.83}{10^4}$	$\frac{0.15}{10^4}$	$\frac{4.43}{10^4}$	$\frac{12.82}{10^4}$	$\frac{0.15}{10^4}$	$\frac{3.81}{10^4}$	$\frac{5.19}{10^4}$
Num($T = 32$)	6	1	61	1	8	0	1	76	0	4	16
Pe($T = 32$)	$\frac{0.92}{10^4}$	$\frac{0.15}{10^4}$	$\frac{9.31}{10^4}$	$\frac{0.15}{10^4}$	$\frac{1.22}{10^4}$	0	$\frac{0.15}{10^4}$	$\frac{11.60}{10^4}$	0	$\frac{0.61}{10^4}$	$\frac{2.44}{10^4}$

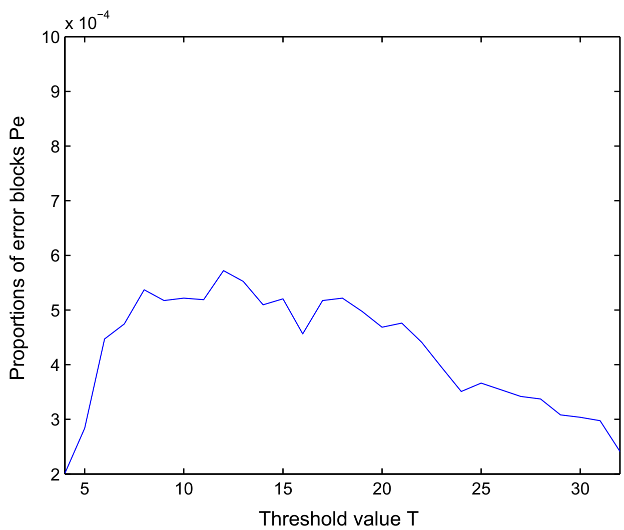

Fig. 2. The proportions of error blocks *Pe* for various threshold values *T*.

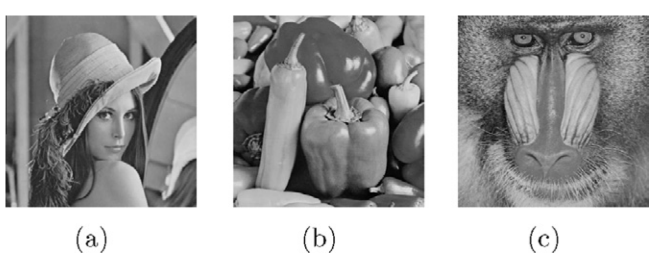

Fig. 3. Three cover images with size 512 × 512: (a) Lena (b) Peppers (c) Baboon.

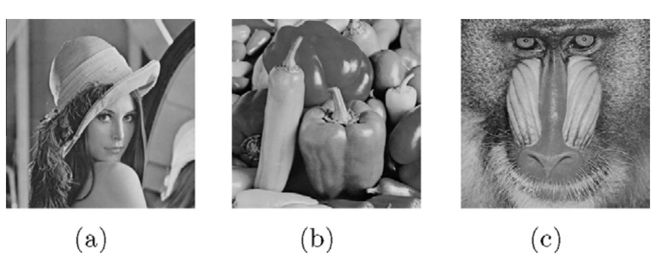

Fig. 4. Three stego images ($T = 7$, $k_l = 2$, $k_h = 3$) (a) Lena (embedded 587220 bits, PSNR = 44.05 dB) (b) Peppers (embedded 578624 bits, PSNR = 44.26 dB) (c) Baboon (embedded 714164 bits, PSNR = 41.58 dB).

shows the results of the proposed method in terms of embedding capacity and PSNR value. The embedding capacities (in bits) and the PSNR values are average values of the results executed by random bit streams many times.

Table 3 shows the comparisons of the results between Wu et al.'s [11] and ours in terms of embedding capacity and PSNR value, where the proposed method uses 3–4 division with $T = 15$. It is shown that our method provides both much larger embedding capacity and better image quality.

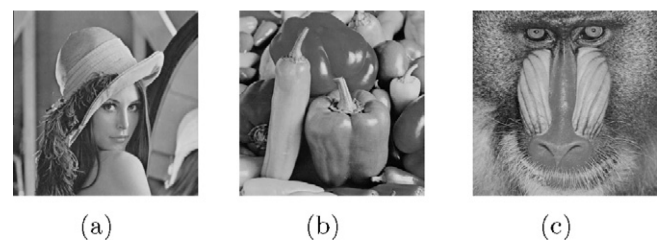

Fig. 5. Three stego images ($T = 15$, $k_l = 2$, $k_h = 4$) (a) Lena (embedded 572744 bits, PSNR = 42.68 dB) (b) Peppers (embedded 562648 bits, PSNR = 43.23 dB) (c) Baboon (embedded 760664 bits, PSNR = 37.71 dB).

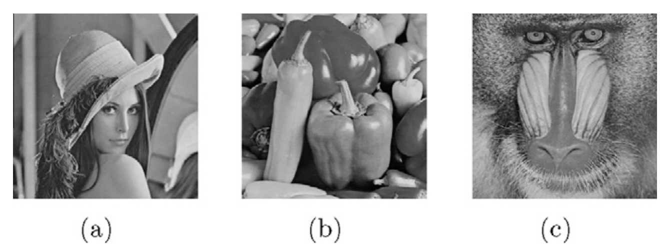

Fig. 6. Three stego images ($T = 15$, $k_l = 3$, $k_h = 4$) (a) Lena (embedded 810564 bits, PSNR = 39.58 dB) (b) Peppers (embedded 805492 bits, PSNR = 39.79 dB) (c) Baboon (embedded 903580 bits, PSNR = 36.90 dB).

A number of secret bits are embedded into every block using the modified LSB substitution method, which was used by Yang et al.'s [16]. Parameters $k_l - k_h$ both used in Yang et al.'s and ours represent the same condition. A steganographic scheme is preferred if it can provide higher PSNR values when concealing with the same embedding capacity. Thus, the comparisons of PSNR values based on the embedding capacities that are slightly larger than those provided by Yang et al. are given in Table 4. The statistics show that the proposed steganographic method is better than Yang et al.'s approach when using a two-level adaptive embedding strategy. A possible reason is that our proposed method considers non-overlapping 2 × 2 pixel blocks instead of two consecutive pixels, so the features of edge may be considered sufficiently and the pixels in edge areas can tolerate much more changes without making perceptible distortion.

5. Conclusions

In this paper, we have proposed a novel steganographic method based on four-pixel differencing and modified LSB substitution. Secret data are hidden into each pixel by the k-bit modified LSB substitution method, where k is decided by the average difference value of a four-pixel block. Readjustment has been executed to extract the secret data exactly and to minimize the perceptual distor-

Table 2
Experimental results with different parameters.

Covers	T = 7, 2–3		T = 12, 2–4		T = 15, 3–4		T = 18, 2–5		T = 18, 3–5		T = 21, 4–5	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
Elaine	668384	42.19	648040	39.71	821640	38.98	585144	38.51	826856	37.02	1060572	34.06
Lena	587220	44.05	590088	41.87	810564	39.57	579204	39.12	822996	37.45	1062512	34.04
Baboon	714164	41.57	803656	37.09	903580	36.90	825172	32.57	985988	32.27	1132012	31.58
Peppers	578624	44.28	576416	42.49	805492	39.79	568828	39.84	816032	37.89	1060176	34.16
Toys	584932	44.15	600488	41.54	816852	39.36	596892	38.18	834728	36.82	1067484	33.82
Girl	599736	43.69	591832	41.73	808584	39.61	567920	39.77	815500	37.85	1057860	34.24
Gold	624856	43.08	617696	40.80	816892	39.27	586928	38.63	828112	37.11	1062812	33.99
Barb	658560	42.52	724208	38.37	871184	37.67	741468	33.93	930572	33.47	1108808	32.27
Zelda	564248	44.73	557832	43.48	797268	40.16	546392	41.96	801168	39.08	1053524	34.51
Tiffany	579948	44.25	579592	42.32	805760	39.77	568008	39.86	815472	37.92	1059424	34.20
Average	616067	43.45	628985	40.94	825782	39.11	616596	38.24	847742	36.69	1072518	33.69

Table 3
Comparisons of the results between Wu et al.'s and ours.

Covers	Wu et al.[11]		Ours	
	Capacity	PSNR	Capacity	PSNR
Elaine	760182	37.28	821640	38.98
Lena	768612	37.35	810564	39.57
Baboon	729526	36.36	903580	36.90
Peppers	774985	37.48	805492	39.79
Toys	772678	37.18	816852	39.36
Girl	771137	37.48	808584	39.61
Gold	768850	37.42	816892	39.27
Barb	738908	35.43	871184	37.67
Zelda	778303	37.70	797268	40.16
Tiffany	772946	37.35	805760	39.77
Average	763613	37.10	825782	39.11

tion. The lemmas and theorems presented in Section 3 are of theoretical importance, because the correctness and feasibility of the proposed method are validated explicitly. The proposed method considers the features of edge sufficiently, so the pixels in edge areas can tolerate much more changes without making perceptible distortion. The experimental results show that the proposed method provides larger embedding capacity and better image quality.

Our proposed method majors in more significant promotions in the terms of capacity and imperceptivity. There is a trade-off between embedding capacity/quality and attack-resistance, and it would sacrifices attack-resistance a little for obtaining higher embedding capacity/quality. In the future, besides the merits achieved in this paper, we will attempt to improve and modify the propose method to achieve stronger security.

Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant Nos. 60873191, 60903152, 60821001), Beijing Natural Science Foundation (Grant No. 4072020).

Table 4
The same embedding capacity with better stego-image quality.

Covers	2–3				2–4				2–5			
	Yang et al.		Ours		Yang et al.		Ours		Yang et al.		Ours	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
Elaine	636044	42.34	647668	42.56	579792	41.54	582512	42.02	607544	34.90	611732	37.18
Lena	575188	44.12	578716	44.31	565936	42.63	568304	42.92	586760	36.73	590360	38.47
Baboon	695310	41.14	701580	41.76	741852	36.96	748056	37.92	727280	32.12	731124	34.05
Peppers	568856	44.21	569512	44.58	557496	43.14	559240	43.45	565496	38.22	565456	40.08
Toys	574656	43.99	577868	44.38	578756	41.79	580728	42.35	579944	37.69	581492	39.04
Girl	579686	43.89	588200	44.01	557680	42.94	562656	43.15	574376	37.07	574412	39.32
Gold	606488	42.87	609900	43.44	574304	42.05	578376	42.37	584288	36.75	586928	38.62
Barb	651008	42.17	650108	42.68	685736	38.55	686392	39.08	718832	32.76	718888	34.36
Zelda	556840	44.66	556592	45.01	542284	44.28	543160	44.52	551282	39.54	552596	41.26
Tiffany	566992	44.38	571608	44.53	556808	43.21	559552	43.40	573068	37.67	572132	39.57
Average	601107	43.38	605175	43.73	594064	41.71	596898	42.12	606887	36.35	608512	38.19
	3–4				3–5				4–5			
Elaine	814184	38.91	815328	39.25	841936	34.17	844548	36.06	1076328	31.96	1077364	33.16
Lena	807256	39.49	808344	39.67	828080	35.69	830400	37.02	1069400	32.91	1070440	33.66
Baboon	895214	36.13	897452	37.04	921760	31.79	923596	33.57	1116240	30.57	1116068	32.02
Peppers	803036	39.68	803804	39.88	813904	36.79	813844	38.06	1062312	33.41	1062232	34.06
Toys	813666	39.06	814548	39.45	823536	36.43	824488	37.39	1067128	33.29	1067484	33.82
Girl	803128	39.59	805608	39.74	819824	35.93	819828	37.56	1065272	32.95	1065244	33.85
Gold	811440	39.19	813308	39.43	826432	35.71	828112	37.11	1068576	32.88	1069296	33.67
Barb	867156	37.33	866836	37.78	916128	32.39	915532	33.84	1113424	31.06	1112176	32.18
Zelda	795430	40.11	795868	40.23	804428	37.67	805304	38.71	1057574	33.74	1058012	34.25
Tiffany	802692	39.72	803880	39.85	818952	36.39	818188	37.75	1064836	33.25	1064244	33.94
Average	821320	38.92	822498	39.23	841498	35.30	842384	36.71	1076109	32.60	1076256	33.46

References

[1] H.J. Highland, Data encryption: a non-mathematical approach, Comput. Secur. 16 (1997) 369–386.
[2] R.J. Anderson, F.A.P. Petitcolas, On the limits of steganography, IEEE J. Sel. Areas Commun. 16 (1998) 474–481.
[3] F.A.P. Petitcolas, R.J. Anderson, M.G. Kuhn, Information hidinga survey, Proc. IEEE Spec. Issue Prot. Multimedia Content 87 (7) (1999) 1062–1078.
[4] H. Wang, S. Wang, Cyber warfare: steganography vs. steganalysis, Commun. ACM 47 (10) (2004) 76–82.
[5] D.W. Bender, N.M. Gruhl, A. Lu, Techniques for data hiding, IBM Syst. J. 35 (1996) 313–316.
[6] R.Z. Wang, C.F. Lin, J.C. Lin, Image hiding by optimal LSB substitution and genetic algorithm, Pattern Recognit. 34 (3) (2001) 671–683.
[7] C.K. Chan, L.M. Chen, Hiding data in images by simple LSB substitution, Pattern Recognit. 37 (3) (2004) 469–474.
[8] I.C. Lin, Y.B. Lin, C.M. Wang, Hiding data in spatial domain images with distortion tolerance, Comput. Stand. Inter. 31 (2) (2009) 458–464.
[9] D.C. Wu, W.H. Tsai, A steganographic method for images by pixel-value differencing, Pattern Recognit. Lett. 24 (9–10) (2003) 1613–1626.
[10] C.C. Chang, H.W. Tseng, A steganographic method for digital images using side match, Pattern Recognit. Lett. 25 (12) (2004) 1431–1437.
[11] H.C. Wu, N.I. Wu, C.S. Tsai, M.S. Hwang, Image steganographic scheme based on pixel-value differencing and LSB replacement methods, Proc. Inst. Elect. Eng., Vis. Images Signal Process 152 (5) (2005) 611–615.
[12] Y.R. Park, H.H. Kang, S.U. Shin, K.R. Kwon, A Steganographic Scheme in Digital Images Using Information of Neighboring Pixels, vol. 3612, Springer-Verlag, Berlin, Germany, 2005. pp. 962–967.
[13] C.H. Yang, C.Y. Weng, A steganographic method for digital images by multi-pixel differencing, in: Proceedings of International Computer Symposium, Taipei, Taiwan, R.O.C., 2006, pp. 831–836.
[14] K.H. Jung, K.J. Ha, K.Y. Yoo, Image data hiding method based on multi-pixel differencing and LSB substitution methods, in: International Conference on Convergence and Hybrid Information Technology, 2008, pp. 355–358.
[15] C.M. Wang, N.I. Wu, C.S. Tsai, M.S. Hwang, A high quality steganography method with pixel-value differencing and modulus function, J. Syst. Softw. 81 (2008) 150–158.
[16] C.H. Yang, C.Y. Weng, S.J. Wang, H.M. Sun, Adaptive data hiding in edge areas of images with spatial LSB domain systems, IEEE Trans. Inf. Forensics Secur. 3 (3) (2008) 488–497.
[17] Y.K. Lee, L.H. Chen, High capacity image steganography model, Proc. Inst. Elect. Eng. Vis. Image, Signal Processing 147 (3) (2000) 288–294.
[18] S.J. Wang, Steganography of capacity required using modulo operator for embedding secret images, Appl. Math. Comput. 164 (2005) 99–116.