An adaptive high-fidelity steganographic scheme using edge detection and hybrid hamming codes☆

Chin-Feng Leea, Chin-Cheng Changb,⁎, Xiaozhu Xieb, Ke Maoc, Run-Hua Shic

a Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan
b Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan
c School of Computer Science and Technology, Anhui University, Hefei 230601, China

ARTICLE INFO

Keywords:
Data-hiding
Edge detection
Hamming code
Matrix embedding

ABSTRACT

In this paper, we propose a novel image steganography technique with an adaptive embedding scheme that combines the methods of edge detection and hybrid Hamming codes to conceal a secret message in a cover image. We use the Canny edge detection algorithm to identify the sharpness degree of a block because the human visual system is more sensitive to the smooth regions than the sharp regions in the image; therefore, embedding confidential information in the cover image according to the sharpness degree of image regions will provide the consequent stego image with superior visual quality. Inspired by this, the proposed scheme identifies the sharpness degree of each block when embedding a secret message into a cover image, and the volume of data embedded in each block depends on the block sharpness; i.e., the sharper the block is, the more the data embedded into it. To embed a secret message into a cover image, the proposed scheme first uses the sharpest regions of the image and then gradually proceeds to the less sharp regions. Therefore, the proposed scheme can effectively reduce the distortion of the stego image, making it imperceptible. The experimental results show that the proposed scheme achieves better image quality than the previously reported related steganographic techniques.

1. Introduction

In recent years, with the rapid development of multimedia technologies and computer networks, the Internet has become more popular all over the world, and large amounts of valuable and sensitive data are being transmitted daily. Therefore, extensive research in the field of information security is being conducted to ensure the security of these valuable data during the transfer process and prevent interception from illegal third parties. Steganography is one of the most popular technologies used to tackle this issue because it can secure the transmitted data by embedding the secret data into a carrier, such as an audio, image, video, and text, and so on. Image steganography involves embedding the secret data into an image, which is called a cover image, and after the data are embedded in the image, it is referred to as a stego image, which displays almost the same as the cover image.

The most popular criteria for measuring the performance of image steganography are embedding capacity and peak signal-to-noise ratio (PSNR). The PSNR is used to measure the degree of distortion between the cover image and stego image. When the PSNR is low, the image can be identified easily, making it vulnerable to visual and statistical attacks. Therefore, it is important for image steganography methods to provide the highest possible values of PSNR. Embedding capacity refers to the maximum number of secret bits that can be embedded per pixel (bpp) in a cover image. In general, under the premise of image quality, the higher the embedding capacity, the better the steganography method.

Currently, there are three major image steganography methodologies: (1) the least significant bit (LSB) replacement method, in which the LSBs of each pixel in the cover image are replaced by message bits. This method is the most popular type of steganography method because it is easy to implement and it has a high embedding rate [1–5]; (2) the pixel value differencing method, which was proposed by Wu et al. [6] in 2003, and involves embedding the secret message according to the mean value characteristic of two adjacent pixel values [7]; (3) the image steganography technique based on an encoding function, in which the secret message is embedded using a special equation or distinctive function [8–10].

To reduce the possibility of visual attacks and statistical analyses,
some researchers have applied an edge detection mechanism in the LSB replacement method [11–17]. Luo et al. [11] proposed an edge adaptive image steganography based on LSB matching revisited. This method adaptively selects the embedded region according to the size of the secret information and the difference between two adjacent pixel values in a cover image. Chen et al. [14] combined a fuzzy edge detector and Canny edge detector to exploit more edge pixels for embedding data. This method can achieve a higher embedding capacity as well as resist image steganalysis. Kaur et al. [15] improved Chen et al.’s scheme by applying $2^k$ correction method to obtain better imperceptibility. Tseng et al. [16] extended Chen et al.’s scheme to block-based and achieved minimal distortion for better image quality. Bai et al. [17] determined the edge information by most significant bits to avoid side information, thereby obtaining a higher embedding capacity. Subsequently, Nguyen et al. [12] proposed an adaptive multi bit-plane image steganography using block data-hiding; this method improved the image quality by selecting the complex regions of a cover image depending on an adaptive complexity threshold computation. To obtain a high embedding capacity while maintaining appreciable visual quality, Al-Dmour et al. [13] proposed a steganography embedding method based on edge identification and exclusive OR (XOR) coding. In this scheme, a cover image is divided into non-overlapping blocks of size $3 \times 3$, and the four corner pixels of a block are fixed, which are used to estimate the number of secret bits can be embedded, and the residual pixels except the central pixel are employed to embed secret bits. Using this scheme, three secret bits can be embedded into one block, with only 1.25 bits being flipped. This method also considers the human vision sensitivity and classifies the edge regions of the cover image into five different groups depending on the degree of sharpness. In this method, when embedding a secret message into a cover image, the sharpest regions of the image are first used, and then the succeeding less sharp regions. Many scholars have proposed different methods to improve the embedding capacity and image quality. However, improving embedding capacity and image quality are mutually exclusive and involve a trade-off.

To achieve a high embedding efficiency, Crandall et al. [18] originally proposed the matrix encoding, which effectively reduced the modifications when embedding a message into a cover image. Subsequently, this scheme was used in a steganographic scheme known as FS [19]. Accordingly, the proposed scheme uses matrix embedding to obtain a high embedding efficiency.

Hamming Codes invented by Richard Hamming in 1950 are perfect codes, which add additional check bits to data bits of a message to correct one bit error of a message. In addition, for a given bit block, Hamming code requires a minimum occurrence of redundancy. The $(n, k)$ Hamming code is used for matrix embedding; here, $(n - k)$ secret bits can be embedded into $n$ cover bits, and only one bit in the cover vector is flipped. Because of this property, many scholars have proposed different methods to improve embedding capacity and image quality based on Hamming code. Bai et al. [20] inserted a secret message into an absolute moment block truncation coding-compressed bit-stream and used matrix embedding with Hamming code to achieve a high payload.

Several researches have aimed to obtain embedding capacities as large as possible, although at the expense of the image quality distortion [4,14,16,17,19,20]. This motivation is not suitable for some practical applications, such as medical images and forensic evidence, which require a high image quality at a given embedding capacity requirement. To improve the stego image quality, a novel image steganography scheme with an adaptive embedding capacity that combines edge detection and hybrid Hamming codes is proposed in this paper. This scheme makes the embedding capacity adjustable via rectifying the threshold. Using this scheme, varying quantities of secret message can be embedded according to sharpness degrees of image regions; logically, more data is embedded in the sharper areas and less data in the smooth areas. In the proposed scheme, when a secret message is embedded into a cover image, the sharpest regions of the image are first used, and then the succeeding less sharp regions. The experimental results show that the proposed scheme can effectively reduce the stego image distortion and achieve imperceptibility.

2. Related work

In this section, we first introduce the edge detection method and then matrix embedding using the Hamming code method.

2.1. Edge detection

The human visual system is more sensitive to the smooth regions than the sharp regions of an image. Therefore, embedding confidential information into the cover image gradually from the sharpest to the least sharp regions results in stego images with better visual quality.

The term “edge” refers to the boundary between the pixels of two regions in an image. Edge detection is an important research field in image processing and computer vision, especially with respect to feature detection and feature extraction; this is because the edge of the image, where the image attributes change notably, usually reflects the important information and changes of the image characteristics. Edge detection in stego image processing significantly reduces the volume of information to be analyzed, as it eliminates irrelevant information, thereby preserving the important structural attributes of the image. The Canny edge detection algorithm was proposed in 1986 as a multi-stage edge detection algorithm [21], and it is still used extensively. For optimal edge detection, the Canny edge detection algorithm works effectively using these three principles: (1) optimal inspection, (2) optimal location selection criteria, and (3) checking of points corresponding with the peripheral points. Since the Canny edge detection algorithm is a popular and convenient method, it is adopted in this study to detect a wide range of edges in images.

Since a cover image is modified in the process of embedding, the generated stego image differs from the cover image. In LSB substitution method, the lower order bit planes are manipulated. When detecting the cover image and stego image directly, the obtained edge pixel values are not identical; thus, the recipient of the message may extract the wrong confidential information. The modification of the low level bit planes has little influence on the image; therefore, in the embedding process, before detecting the edge of the cover image, we first set the pixel values at the lower order bit planes of the cover image to 0. Similarly, we set the pixel values at the same lower order bit planes of the stego image to 0 before detecting the edge of the stego image.

2.2. Matrix embedding using Hamming code

Matrix embedding first encodes the cover image and secret message with an error correction code; then, the cover image is modified according to the encoding results to obtain the stego image. The $(n, k)$ Hamming code is used to embed $n - k$ secret bits into $n$ cover bits, where only one bit is flipped. The equations associate with $n$ and $k$ are as follows:

$$n = 2^r - 1, \quad r \geq 2.$$  

(1)

$$k = 2^r - r - 1, \quad r \geq 2.$$  

(2)

Mao et al. [22] introduced a fast algorithm for matrix embedding steganography to modify the method proposed by Crandall et al. [18]. To reduce the computational complexity, all the columns in the original parity check matrix $H^{(n)}$, in Hamming code are arranged in ascending order from left to right, and then, Mao’s method can obtain a new corresponding parity check matrix $H'^{(n)}$. For example, in (7, 4) Hamming code where $n = 7$ and $k = 4$, all columns in $H^{(n)}$ are changed to ascending ordered columns and presented as $H'^{(n)}$(Fig. 1). There is no need to find the cost leader when embedding, so it greatly reduces computational complexity.
In addition, a function, $D(V)$, is used to convert a column vector $V$ into a decimal digit $d$. For example, given a column vector $V = (0000101_2)$, we can get a decimal digit $d = 5$. Furthermore, a function $T(V', d)$ is used to set the $d^{th}$ bit of the vector $V'$ as 1, while the rest of the bits of the vector are 0s. For example, given a column vector $V = (0000101_2)$ and $d = 5$, we get a vector $T(V, d) = (0000101_2)$.

Next, we introduce their fast matrix embedding and extraction processes based on Hamming code. To facilitate the understanding of the reader, we use a vector to demonstrate the embedding and extraction processes as follows.

Specifically, we assume that the $n$ cover bits are represented as a vector $\mathbf{V} = (v_1, v_2, \ldots, v_n)$ and that the stego bits are represented as a vector $\mathbf{V'} = (v_1', v_2', \ldots, v_n')$. The binary secret bits are $S = (s_1, s_2, \ldots, s_{n-k})$.

To embed the secret bits, we can obtain the stego vector of $n$-bit using the following equation.

$$V' = V + T(V, D(S - H^{(n)}V) \mod 2),$$

where $H^{(n)}$ is the corresponding parity check matrix with size $(n-k) \times n$.

The process of extracting the secret bits from the stego vector of $n$-bit is as follows:

$$S' = H^{(n)} \times V',$$

where $H^{(n)}$ is the corresponding parity check matrix with size $(n-k) \times n$; $V'$ is the vector of stego bits; and $S'$ is the vector of binary secret bits that we extract. When $S'$ is the same as $S$, it means that the secret bits have been extracted correctly.

For instance, given a length-7 block of cover image $V = (0000101_2)$, and a 3-bit message vector $S = (1 1 0)$. The stego vector of $n$-bit can be obtained by using embedding function and the secret embedding (as shown in Fig. 2) can be hidden into the cover vector $V$.

The computation using Mao’s method takes only 0.0339 s, while Filler et al.’s proposed scheme [23] takes 0.2624 s when $10^4$ bits are embedded. It is obvious that Mao’s method can achieve a lower computation complexity when the embedding rates are the same as in Filler et al.’s method. Therefore, in this paper, we use Mao’s method to achieve highly effective steganographic results.

3. Proposed scheme

In this section, we present the details of our proposed scheme. This section consists of three parts: (1) Image preprocessing, (2) Message embedding, and (3) Message extraction.

3.1. Preprocessing phase

Since the human eye is more sensitive to changes in the smooth areas than in the sharp areas of an image, our proposed scheme identifies the sharpness degree of each block when embedding a secret message into a cover image. Then according to the sharpness degree, we embed a different volume of data into the block. The sharper the block is, the more data we embed into it. This strategy can achieve a higher visual quality with the specified amount of embedding data.

First, the Canny edge method has been used for detecting the edges of images. Then, varying quantities of secret message is designed to hidden according to the block degrees of sharpness; logically, more data is embedded in the sharper areas and less data in the smooth areas. Besides, when embedding a secret message into a cover image, the sharpest regions of the image are first used, and then the succeeding less sharp regions. The sharpness degree $\omega$ in each image block is defined before the embedding and extraction processes to obtain the maximum number ($N$) of information embedded (or extracted) into (from) each block. After Canny edge method is applied, the number of non-zero elements in a block can be calculated. Let $\delta$ stand for the number of non-zero elements in a block. Since by changing one bit, the $(n, k)$ Hamming code can embed $(n - k)$ bits into $n$ bits, $N = (n - k) \times \omega$ (bits) can be embedded into the cover stream of each image block.

To better understand the details of the proposed scheme, we use an example to illustrate the generation of a cover stream from a given image block. We assume the image block $B$ with a size of $m_1 \times m_2 = 3 \times 3$ as Fig. 3, and the threshold $\delta$ is set to be 3. Then, a cover stream of 27 bits, $011010000011010110001110100$, can be obtained by lining up the xLSB planes, $x \in \{1, 2, 3\}$, of the block in a zigzag scanning order.

Compared with changing the most significant bits, changing the LSBs generates less impact on image quality, so we choose the xLSB plane ($x = 1, 2, \ldots, \theta$) of the image to embed information. Since the values at the xLSB planes ($x = 1, 2, \ldots, \theta$) of the cover image will be changed, we first set the xLSB planes ($x = 1, 2, \ldots, \theta$) of cover image and stego image to 0s before the Canny edge detection process is applied to ensure the sharpness degree of each block in the cover image and the stego image is the same, thereby guaranteeing that the extracted information is the same as the embedded information.

Let us consider an example to illustrate how to obtain the sharpness degree of a block. First, we set the 3 LSB planes of an image block B to 0s and get a residual block $B'$, as shown in Fig. 4, where $x = 1, 2, \ldots, \theta$. Then, we apply the Canny edge detection onto the image block $B'$ to get an edge block $EB$, as shown in Fig. 5. From Fig. 5, we obtain $\delta = 3$; i.e., the number of non-zero elements in the block is 3.

According to the value of $\delta$, we design an equation, Eq. (5), to define the sharpness degree of a block as shown in the third step of the image.
preprocessing procedure. The details are as follows:

**Image preprocessing procedure**

**Inputs:** An image (I) and a predetermined threshold \( \theta \).

**Output:** The sharp degree \( \omega \) of each image block.

**Step 1:** Set the \( x \)LSB plane of image \( I \) to 0s and get a residual image \( \tilde{I} \), where \( x = 1, 2, ..., \theta \). Then, an edged image \( E \) is obtained through the image \( R \) by performing the Canny edge detection method.

**Step 2:** Divide the edged image \( E \) into non-overlapping blocks of size \( m_1 \times m_2 \).

**Step 3:** Calculate the sharpness degree \( \omega \) of each image block according to the value of \( \delta \) obtained from the corresponding edged image block as shown in Eq. (5):

\[
\omega = \begin{cases} 
1, & 0 \leq \delta < \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor \\
2, & \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor + 1 \leq \delta < 2 \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor \\
3, & 2 \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor + 1 \leq \delta < 3 \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor \\
\vdots & \\
M, & (M-1) \left\lfloor \frac{m_1 \times m_2}{M} \right\rfloor + 1 \leq \delta \leq m_1 \times m_2
\end{cases}
\]

(5)

where \( M \) is the sharpest degree among all the block, that is \( M = 4 \times \delta \), \( \lfloor x \rfloor \) is the floor function which means the greatest integer less than or equal to \( x \).

### 3.2. Message embedding

The two important evaluation criteria of image steganography include embedding rate and image quality. Assuming that we only use the simple \((n, k)\) Hamming code for matrix embedding, i.e., we embed \((n - k)\) secret message into a \(n\)-dimensional cover vector by changing only one bit. Let \( n = 2^r - 1 \). Then, the embedding rate \( ER \) of a cover block can be represented as follows:

\[
ER = \frac{\theta \times m_1 \times m_2}{n} \times (n-k)
\]

The variance between the cover block and the stego block can be represented as follows:

\[
VAR = \sum_{x=1}^{\theta} \frac{\theta \times m_1 \times m_2}{2^r-1} \times 2^x
\]

To go further, the \( ER \) and \( VAR \) described above can be represented as follows:

\[
ER = \frac{\theta \times m_1 \times m_2}{2^r-1} \times r, \ r \geq 2.
\]

\[
VAR = \sum_{x=1}^{\theta} \frac{\theta \times m_1 \times m_2}{2^r-1} \times 2^x, \ r \geq 2
\]

From Eq. (6), we can deduce that when the block is constant, \( ER \) and \( r \) are inversely proportional, i.e., the smaller the \( r \), the higher the \( ER \). Similarly, from Eq. (7), we can deduce that when the block is constant, \( VAR \) and \( r \) are inversely proportional; in other words, the higher the \( r \), the smaller the variance, and we can get a higher image quality.

To achieve a balance between image quality and embedding rate, we use a hybrid of \((n^{(1)}, k^{(1)})\) Hamming code and \((n^{(2)}, k^{(2)})\) Hamming code, in which \( k^{(1)} < k^{(2)} \). Furthermore, since the human eye is more sensitive to the smooth regions than the sharp regions in an image, we use a hybrid of Hamming codes to embed information into the \( x \)th layer of the bit-plane, i.e., the \( x \)LSB plane, of an image block. When \( x \) is small, \((n^{(1)}, k^{(1)})\) Hamming code is used, and when \( x \) is large, we use \((n^{(2)}, k^{(2)})\) Hamming code. Furthermore, when embedding information into the cover image, we apply \((n^{(3)}, k^{(1)})\) Hamming code in sharper areas and \((n^{(3)}, k^{(2)})\) Hamming code in less sharp areas.

When embedding a secret message, first, we choose the blocks with the most sharpness degree in the cover image, and then, the proposed scheme gradually proceeds to hide secret data to the less sharp areas. In other words, suppose the current proceeded block with a sharpness degree is represented as \( G \), then let \( G = M \); after embedding in all blocks with \( G \) sharpness degree, we proceed to \( G = G - 1 \), until \( G = 1 \). Accordingly, we present a flowchart (Fig. 6) to introduce the embedding process and present the embedding procedure as follows.

Proceed each block and obtain the sharpness degree \( \omega \) of each block in the image \( I \) by preprocessing procedure, then reserve \( P \) blocks in the image \( I \) to store the position information \((F, e, l)\) and additional information (such as the key used to decrypt the secret message), and we assume that the length of the additional information is \( \phi \), so we have

\[
P = \left\lceil \frac{\log_2[(N + m_1 m_2 + \theta + \phi)]}{m_1 m_2} \right\rceil
\]

Here, \( F \) represents the position of the block that contains the last bit of secret data in the cover image, \( e \) represents the last position of the pixel which has data embedded within, and the pixel is denoted as \( p, l \) represents the last position of the bit that is embedded in \( p \). We set the initial value of \((F, e, l)\) as \((1, 1, 1)\) and use the LSB method to embed them.
Embedding procedure:

Inputs: Cover Image \(I\) (size \(W \times H\)), threshold: \(th\), initial \(G = 2th\), secret message \(S\), secret message length \(L\), block size \(m_1 \times m_2\), and the block number \(N\).

Output: Stego Image \(I’\).

Step 1:

Step 1: Divide the image \(I\) into non-overlapping blocks of size \(m_1 \times m_2\).

Step 2: Starting from the \((P + 1)\)th block to the \(N\)th block, where \(N = \frac{W \times H}{m_1 \times m_2}\), we successively do two operations for each block.

(1) Extract the \(x_{LSB}\) \((x = 1, 2, ..., th)\) planes of the block in ascending order to constitute the cover stream, \(C\), the length of which are \(ρ = th \times m_1 \times m_2\).

(2) Determine the sharpness degree \(ω\) of the block, if \(ω \neq G\), we keep the block unchanged, and turn to the next block; otherwise, do operations on the block as follows:

(a) If \(ω\) is even, embed information into \(x_{LSB}\) plane \((x = 1, 2, ..., \lfloor \frac{ω}{2}\rfloor\) \) of the block by using \([\frac{m_1 \times m_2}{n^{(1)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(1)}, k^{(1)})\) Hamming code.

(b) If \(ω\) is odd, embed information into \(x_{LSB}\) plane \((x = 1, 2, ..., \lfloor \frac{ω}{2}\rfloor + 1\) \) of the block by using \([\frac{m_1 \times m_2}{n^{(2)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(2)}, k^{(2)})\) Hamming code, and embed information into the \(x_{LSB}\) plane of the block where \(x = \lfloor \frac{ω}{2}\rfloor + 1\) by using \([\frac{m_1 \times m_2}{n^{(2)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(2)}, k^{(2)})\) Hamming code.

(c) Update the value of \((F, e, l)\).

Step 4: Embed the last secret information \((F, e, l)\) and additional information into the first \(P\) blocks by using the simple LSB method.

More details are illustrated by an example using a hybrid of \((3, 1)\) Hamming code and \((7, 4)\) Hamming code (Fig. 7). We set the block size \(m_1 \times m_2\) to \(7 \times 3\), determine \(th = 2\), and assume that the sharpness degree \(ω\) of the block is 3. For a secret message \(S = (1011000001001101)\), the embedding process is manipulated by embedding \(S\) into the 1 LSB plane using 5 times \((3, 1)\) Hamming code and the 2 LSB plane using 2 times \((7, 4)\) Hamming code.

3.3. Message extraction

In this section, we introduce the extraction process, and we present a flowchart (Fig. 8). To extract the correct secret message, the order of extraction is the same as in the embedding process. We extract the secret message by beginning with the sharpness degree areas in the Stego image, and then proceed to the less sharp areas. In other words, after completing \(M\) sharpness degree areas, we proceed to \(M - 1\), and then \(M - 2\), until the sharpness degree is equal to 1. For convenience, the current processing sharpness degree is represented as \(G\).

Extraction procedure:

Inputs: Stego Image \(I’\) with size \(W \times H\), block number: \(N\), initial sharpness degree \(G = 2 \times th\), block size \(m_1 \times m_2\).

Output: Secret Message \(S\).

Step 1: Divide the image \(I’\) into non-overlapping blocks of size \(m_1 \times m_2\).

Step 2: Starting from the first block of the stego image \(I’\), we successively extract \((F, e, l)\) and additional information, then record the block number as \(P\).

Step 3: Get the sharp degree of each block in the image \(I’\) in the order of processing the image \(I’\).

Step 4: Starting from the \((P + 1)\)th block to the \(N\)th block, we successively extract two operations for each block as follows until the current position is equal to \((F, e, l)\), where \(N = \frac{W \times H}{m_1 \times m_2}\).

(1) We take out the \(x_{LSB}\) planes \((x \in\{1, 2, ..., th\})\) of the block in an ascending order to constitute the cover stream \(C\), the dimensions of which are \(ρ = th \times m_1 \times m_2\).

(2) To determine the sharp degree \(ω\) of the block, if \(ω \neq G\), we keep the block unchanged, and turn to the next block; otherwise, we will do operations on the block as follows:

(a) If \(ω\) is even, extract information from \(x_{LSB}\) plane \((x = 1, 2, ..., \lfloor \frac{ω}{2}\rfloor\) \) of the block by using \([\frac{m_1 \times m_2}{n^{(1)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(1)}, k^{(1)})\) Hamming code.

(b) If \(ω\) is odd, extract information from \(x_{LSB}\) plane \((x = 1, 2, ..., \lfloor \frac{ω}{2}\rfloor + 1\) \) of the block by using \([\frac{m_1 \times m_2}{n^{(2)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(2)}, k^{(2)})\) Hamming code, and extract information from the \(x_{LSB}\) plane of the block where \(x = \lfloor \frac{ω}{2}\rfloor + 1\) by using \([\frac{m_1 \times m_2}{n^{(2)}}] \times \lfloor \frac{ω}{2}\rfloor\) times \((n^{(2)}, k^{(2)})\) Hamming code.

To facilitate understanding, we give an example using the hybrid of \((n^{(1)}, k^{(1)})\) Hamming code and \((n^{(2)}, k^{(2)})\) Hamming code \((l_n^{(1)} = 3, k^{(1)} = 1, n^{(2)} = 7, k^{(2)} = 4)\) to illustrate the message extraction process as shown in Fig. 9. The to-be-processed block has size \(m_1 \times m_2 = 7 \times 3\). We assume that the sharpness degree \(ω\) of the block is 3 and \(th = 2\).

4. Experiments and analyses

The experimental results obtained by the method we proposed are
explained below and the results of the payload and image quality are presented. The experimental environment, such as software and hardware, cover image, block size, Hamming code type, and embedded message, are as follows:

The experimental hardware is Lenovo T61, CPU:Intel(R) Core(TM) i7-3770 @3.40 GHz, RAM: 8.00 GB and the software consists of Windows 7 Professional Service Pack 1. The application program is MATLAB R2009a. We applied the proposed scheme based on (3, 1) Hamming code and (7, 4) Hamming code onto 10,000 natural grayscale images with size of $125 	imes 125$, from the BOWS2 database [24]. Moreover, eight standard test grayscale images (Lena, Lake, Baboon, Airplane, Barbara, House, Peppers, Boat) are as shown in Fig. 9 were used to determine how the difference images influenced the performance of the proposed scheme. The confidential message is generated using a binary random generation function.

There are two measurements to evaluate the steganography methods, i.e., (1) the execution efficiency of the embedding method, which is measured by information load (bits per pixel, bpp) and (2) the quality of the image after steganography, which is measured by PSNR and compare the quality of the cover image and the stego image by the human eye.

4.1. Evaluation of embedding capacity

Embedding capacity is an important indicator to measure steganography schemes. It refers to the quotient between the quantum of secret information embedded in the cover image and the number of pixels in the cover image. We refer to it as embedding rate or payload. The equation for calculating information payload is as follows:

$$ER = \frac{|S|}{W \times H} \text{ (bpp)},$$

where $|S|$ is the amount of secret information embedded into the cover image, and $W$ and $H$ are the width and height of the cover image, respectively.

4.2. Evaluation of the quality of the image

PSNR is used to distinguish the quality of stego images; the larger PSNR represents the lower distortion of the image, and the smaller PSNR represents the greater distortion of the image. PSNR is defined as follows:
\[
\text{Stego image block}
\]

\[
\begin{array}{cccc}
13 & 13 & 8 & \\
3 & 0 & 5 & \\
4 & 10 & 9 & \\
7 & 9 & 5 & \\
8 & 11 & 6 & \\
9 & 10 & 6 & \\
7 & 7 & 6 & \\
\end{array}
\]

\[
\begin{array}{cccc}
000111 & 01 & 000111 & 01 & 000100 & \\
000000 & 11 & 000000 & 00 & 000001 & 01 & \\
000011 & 00 & 000100 & 10 & 000010 & 01 & \\
000101 & 00 & 001010 & 10 & 000010 & 01 & \\
001001 & 11 & 001101 & 11 & 000010 & 01 & \\
001011 & 00 & 001011 & 11 & 000010 & 01 & \\
001101 & 11 & 001111 & 11 & 000010 & 01 & \\
\end{array}
\]

\[
C'=(11010100111110100110010111)00100001101000111111)
\]

Extract data from 1 LSB plane by applying five times (3, 1) Hamming coding, and extract data from 2 LSB plane by using two times (7, 4) Hamming coding.

Secret message \( S = (10110000010011) \)

Fig. 9. Example of message extraction process.

**Table 1**

<table>
<thead>
<tr>
<th>Method</th>
<th>Payload 0.1 bpp</th>
<th>0.2 bpp</th>
<th>0.25 bpp</th>
<th>0.3 bpp</th>
<th>0.4 bpp</th>
</tr>
</thead>
<tbody>
<tr>
<td>APVD [7]</td>
<td>61.94</td>
<td>59.41</td>
<td>58.99</td>
<td>57.24</td>
<td>55.95</td>
</tr>
<tr>
<td>EALSB—MR [11]</td>
<td>60.55</td>
<td>57.85</td>
<td>56.86</td>
<td>55.73</td>
<td>54.39</td>
</tr>
<tr>
<td>MPBDH [12]</td>
<td>62.65</td>
<td>59.64</td>
<td>58.68</td>
<td>57.87</td>
<td>56.65</td>
</tr>
<tr>
<td>Proposed</td>
<td>63.97</td>
<td>60.73</td>
<td>59.70</td>
<td>58.88</td>
<td>57.59</td>
</tr>
</tbody>
</table>

\[
PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right)
\]

where MSE (Mean Square Error) is defined as follows:

\[
MSE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} (C_{ij} - S_{ij})^2
\]

where, \( C_{ij} \) and \( S_{ij} \) are the gray values of the pixel at the \((i, j)\) position of the cover image and stego image, respectively. \( W \) and \( H \) are the width and the height of the cover image, respectively, which is the same size as the stego image.

We conducted several experiments to evaluate the performance of our proposed scheme. First, we used different block sizes \( m_1 \times m_2 \in \{4 \times 4, 7 \times 6, 7 \times 3\}\) for the grayscale image “Baboon,” and fixed \( th = 1 \) or \( th = 3 \), respectively, where \( 1 \leq x \leq th \). We compared the proposed scheme with the method proposed by Al-Dmour and Al-Ani [13], as shown in Fig. 10. In method [13], the payload only considers the edge pixels, and the edged block with size \( 3 \times 3 \). The lower-payload method in [13] can only embed 3 bits; however, the higher-payload method can embed 9 bits at most. The lower-payload method in [13] is comparable to our method with \( th = 1 \), and only the 1 LSB planes of the cover image was changed; the higher-payload method is comparable to our method with \( th = 3 \), and the 3 LSB plane of the cover image can be changed.

Fig. 10(a) shows that the maximal embedding rate of the proposed scheme is more than that of the lower-payload method in [13], and \( m_1 \times m_2 = 7 \times 3 \) is the best choice of block size for our proposed scheme when \( th = 1 \). Fig. 10(b) shows that, with the same embedding rate, the performance of our approach obviously exceeds that of the higher-payload method in [13], and \( m_1 \times m_2 = 7 \times 6 \) is the best choice of block size for our proposed scheme when \( th = 3 \).
Different from some schemes aiming for optimal embedding capacity at the expense of distortion of the image quality, the proposed scheme can achieve high image quality at a given embedding capacity requirement. For further description, the schemes in [20,14,16] can achieve high embedding capacity up to 2 bpp, 2.67 bpp, and 3.32 bpp, with PSNRs of 29 dB, 33 dB, and 32 dB, respectively. However, stego images with PSNR of around 30 dB are not acceptable for some practical applications such as medical images and forensic evidence. On the contrary, the proposed scheme can maintain a high image quality at an acceptable embedding rate. For the “Lena” test image, the PSNR of the proposed scheme is much higher than that of [14] with an embedding capacity of 0.74 bpp (that is, 54.95 dB of the proposed scheme vs. 42.3 dB of [14]). For the “Baboon” test image, the maximum embedding capacity of the proposed scheme is 1.1747 bpp, with a PSNR of 52.62 dB, which is much higher than that of [16], which has a PSNR of 41.47 dB and embedding capacity of 1.06 bpp. In addition, to verify the performance of the proposed scheme, it is also compared with the high image quality-orientated schemes. Table 1 shows the quality of the

![Table 2](image)

The maximum embedding rate of 8 standard grayscale images, where $m_1 \times m_2 = 7 \times 3$.

<table>
<thead>
<tr>
<th>Payload (bpp)</th>
<th>Test Image</th>
<th>Lena</th>
<th>Lake</th>
<th>Baboon</th>
<th>Airplane</th>
<th>Barbara</th>
<th>House</th>
<th>Peppers</th>
<th>Boat</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta = 1$</td>
<td></td>
<td>0.4457</td>
<td>0.4482</td>
<td>0.4858</td>
<td>0.4461</td>
<td>0.4614</td>
<td>0.4437</td>
<td>0.4418</td>
<td>0.4595</td>
</tr>
<tr>
<td>$\theta = 2$</td>
<td></td>
<td>0.5605</td>
<td>0.5807</td>
<td>0.7576</td>
<td>0.5073</td>
<td>0.6217</td>
<td>0.5558</td>
<td>0.5419</td>
<td>0.6212</td>
</tr>
<tr>
<td>$\theta = 3$</td>
<td></td>
<td>0.7425</td>
<td>0.7985</td>
<td>1.1747</td>
<td>0.7705</td>
<td>0.8663</td>
<td>0.7475</td>
<td>0.7075</td>
<td>0.8774</td>
</tr>
</tbody>
</table>

![Fig. 11](image)

(a1) Lena (b1) Lake (c1) Baboon (d1) Airplane

(a2) PSNR=54.9516 (b2) PSNR=54.6775 (c2) PSNR=52.6235 (d2) PSNR=54.9195

(c1) Barbara (f1) House (g1) Peppers (h1) Boat

(c2) PSNR=54.0115 (f2) PSNR=55.1778 (g2) PSNR=55.2946 (h2) PSNR=54.1297

Fig. 11. (a1)–(h1) are cover images, (a2)–(h2) are the corresponding stego images at the max payload where $\theta = 3$. $m_1 \times m_2 = 7 \times 3$. 

Table 2
stego images produced by different steganographic methods with \( th = 1 \), \( m_1 \times m_1 = 7 \times 3 \), and the payload ranging from 0.10 to 0.40 bpp. Obviously, the image quality of our approach exceeds that of other state-of-the-art methods.

In the following section, we discuss the effect of threshold \( th \) on embedding rate and image quality with block size \( m_1 \times m_1 = 7 \times 3 \). We vary the value of \( th \) from 1 to 3, and test our method with the eight standard grayscale images previously mentioned.

Table 2 shows the maximum embedding rate of each image; the maximum embedding rate increases with \( th \), and the larger the embedding rate, the more complex the cover image. For example, the maximum embedding rate of Baboon (1.1747 bpp) was significantly greater than that of Peppers (0.7070 bpp). The image size is 512 \( \times \) 512, and Baboon can embed an additional 122,473 bits of information compared to Peppers. Thus, the embedding capacity is connected with the pictorial content; in a cover image, the predominance of sharp regions will yield a higher embedding rate, and the more the smoother regions, the lower the embedding rate.

Fig. 11 shows the eight cover images and their related stego images with embedded information, with \( th \) varied from 1 to 3. As shown, the quality of the stego images generated by our proposed scheme is completely acceptable from the perspective of the human visual capability.

To discuss the effect of threshold \( th \) on the relationship between maximum embedding rate and the image quality, Fig. 12 shows the performance of our proposed scheme with different values of \( th \) for “Baboon” image where the block size \( m_1 \times m_1 = 7 \times 3 \). From Fig. 12, the smaller the \( th \), the larger the PSNR, which will yield a better image quality. On the contrary, the larger the \( th \), the higher the maximum embedding rate. Obviously, the proposed novel image steganography scheme, which combines the methods of edge detection and hybrid Hamming codes, makes the embedding capacity adjustable via rectifying the threshold. Therefore, using it, varying quantities of secret message can be embedded into a cover image according to the sharpness degree of the image areas, so that an adaptive and high-fidelity embedding is achieved.

5. Conclusions

To meet the requirement of high image quality and a relatively lower embedding capacity for some practical applications, such as medical images and forensic evidence, we propose an adaptive steganographic scheme to adapt to various needs for embedding capacity and image quality. We work on the basis that the human eye is more sensitive to the smooth regions than the sharp regions in an image, and we combine edge detection and matrix embedding using a hybrid of (7, 4) Hamming code and (3, 1) Hamming code to make a trade-off between the stego image quality and embedding capacity. Users can adjust the amount of information embedded into each block in a cover image depending on its sharpness degree. Furthermore, the proposed scheme outperforms other related steganographic schemes in achieving a higher image quality with a relatively low embedding capacity.

Acknowledgments

This research was partially supported by the Ministry of Science and Technology of the Republic of China under the Grants MOST 105-2221-E-324-014 and MOST 106-2221-E-324-006-MY2.

Conflict of interest

None.

References

degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. Since February 2005, he has been a Chair Professor of Feng Chia University. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.

Xiao-Zhu Xie received the MS degree in computer engineering from Xiamen University, China. She is currently pursuing the Ph.D. degree in Computer Science and Engineering from Feng Chia University, Taiwan. Her research interests include reversible data hiding and image processing.

Ke Mao was born in Anhui Province, China, in 1992. She received the B.S. degree in Computer Science and Technology in 2013 from Anqing Normal University, Anhui, China. She is currently a Master student in Computer Application Technology at Anhui University. Her current research interests include information security and information hiding.

Run-Hua Shi received the Ph.D. degree from University of Science and Technology of China in 2011. He is currently a Professor with Anhui University, and a visiting fellow at the School of Computing and Information Technology, University of Wollongong. His current research interest includes classical and quantum cryptography, in particular, privacy-preserving multi-party computation.