Journal of Information Hiding and Multimedia Signal Processing
Ubiquitous International

Volume **5**, Number **3**, July **2014**

Hybrid GEMD Data Hiding

Wen-Chung Kuo and Sheng-Yi Chang

Department of Computer Science and Information Engineering
National Yunlin University of Science & Technology
123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C.
Corresponding author : simonkuo@yuntech.edu.tw

Received September, 2013; revised January, 2014

Abstract. *The rising popularity of Internet provides convenience while also raising the risk of disclosure from the viewpoint of information security. In particular, steganography is a common way to safeguard information because it can be used to embed the secret data into the meaningful message and the attacker cannot find it out. In order to increase embedding capacity and maintain others desirable steganographic attributes, a hybrid GEMD(generalized exploiting modification direction) data hiding method will be proposed in this paper. According to our proposed method, the embedding capacity can be achieved 4bpp (under accepted stego image, i.e., the stego image's quality also is better than 30dB). Moreover, there are three major contributions of this scheme. One, it does not change to the variable codes that eliminated additional communication, another is the number of projection vectors used no longer excessive and the other is no overflow/underflow problem exists. From our simulation results and performance discussion, we can prove that our method does not only to provide the improved embedding capacity but also to maintain good stego image quality.*
Keywords: Data hiding, generalized exploiting modification direction, embedding capacity, pixel segmentation.

1. **Introduction.** Information transmission over the Internet is pervasive. The open nature of the Internet allows the possibility for illicit manipulation of information during transmission. Though the information may be intended to be private, interception, modification or even generation of fake messages may occur. The protection of transmitted information is a fundamental issue. Current methods for safeguarding digital information are divided into cryptography and steganography. For cryptography, encryption algorithms are used to obfuscate the data before being sent. After the encrypted information reaches the destination, a key can be used to extract the secret. Without the key, extraction is designed to be untenable within a reasonable timeframe given current technology. However, the act of sending encrypted data invites scrutiny. In steganography, the message is embedded into a substrate media which provides the interesting attribute of security through stealth. Although they differ in execution, the goals of both cryptography and steganography are to achieve the confidentiality, integrity, availability of data.

In recent years, many researchers have proposed data hiding methods. These methods can be divided into two categories. One is reversible data hiding where extraction of the secret also provides the complete data of the original media. However, this approach cannot embed a large amount of secret data without attracting scrutiny. Conversely, the

other method is called irreversible data hiding where the substrate media is considered a disposable disguise. Since the original media is not reconstructed during secret extraction, methods such as least significant bit (LSB) replacement method [9] or the exploiting modification direction (EMD) method [13] increase secret data hiding capacity.

In 2006, a new data hiding scheme based on EMD was proposed by Zhang and Wang [13] to increase the embedding capacity of secret information. In EMD method, only one of n pixels is modified to embed a secret data value from 0 to $2n$. From a spatial point of view, the relationship of two pixels is limited to five states: upward, downward, left, right, or not moving at all. In other words, it can embedded a secret data value from 0 to 4. Although this method has good image quality, the embedding capacity was only 1.161 bits per pixel(bpp) at best. In 2007, Lee *et al.* (LWC) [5] improved the embedding capacity from 1.161 bpp to 1.5 bpp with a technique that uses two pixels at a time and gives both pixels a fixed evaluation value. However, there are disadvantages that exist in the LWC and Zhang-Wang schemes. For example, the secret data base must be changed from binary to $(2n+1)$-ary before using the Zhang-Wang scheme. Although binary secret data is used in LWC scheme, it only uses two pixels at a time. Recently, to increase hidden data conveniently and embedding capacity, a new data hiding scheme based on generalized exploiting modification direction method (GEMD) was proposed by Kuo and Wang [4]. Specifically, they modify the embedding secret data based from $(2n+1)$ to 2^{n+1} and hidden data capacity to $(n+1)$ binary bits embedded into n adjacent pixels. According to [4], the LWC-scheme is a special case of Kuo-Wang scheme when $n = 2$.

In 2008, Lee *et al.* (LCW) proposed a data hiding based on vector of coordinates and pixel segmentation strategy [6] to enhance the secret data capacity of EMD. However, the space of the vector modification area (VMA) was determined to be insufficient and the variable codes needed to be exchanged before additional communication between sender and receiver. In order to remove these disadvantages, we will propose a hybrid GEMD data hiding method in this paper. The major contributions of this approach are better utilization of space to improve data hiding capacity, changing the variable code into fixed that eliminated additional communication and the number of projection vectors used no longer excessive. According to the experimental results and performance discussion, we show that our method provides the improved embedding capacity while maintaining good stego image quality.

The organization of this paper is as follows: In section 2, we briefly review three data hiding schemes based on different extraction functions such as EMD [13], GEMD [4] and pixel segmentation strategy. The proposed method and experimental results are given in in Section 3 and 4, respectively. Finally, some conclusions are provided in Section 5.

2. **Related work.**

Many EMD-type data hiding schemes [3, 4, 5, 6, 10, 13] have been previously proposed. In this section, we will briefly review three of these schemes [4, 5, 13].

2.1. **The data hiding scheme based on EMD.**

In 2006, Zhang and Wang proposed a data hiding scheme based on EMD. The scheme incorporated the LSB concept which modified one pixel in the selected pixel group for the range between 1 and -1 resulting in fast and efficient secret embedding. Notations defined below are used to introduce the EMD-scheme.

I_C**:** The grayscale cover image.
I_S**:** The grayscale stego image.

$O_{EMD}(\cdot)$**:** Obtain all n-tuples $(p_1, p_2, \ldots, p_n)$ from partitioning the image I_C into non-overlapping n-pixel blocks by scanning each line of pixels from left to right in a top-down manner, as shown in Fig.1.

$O_{EMD-S}(\cdot)$**:** Obtain $2n+1$-ary data m from partitioning the secret data stream M for each block.

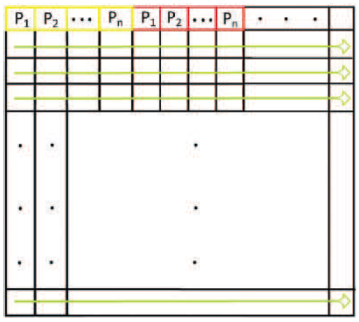

FIGURE 1. The embedding data sequence for EMD-scheme

Algorithm EMD (Embedding Algorithm for EMD Scheme):
Input: cover image I_C and binary secret data stream M
Output: stego image I_S

(EMD-1): Obtain all n-pixel blocks $(p_1, p_2, \ldots, p_n)$ from I_C by using $O_{EMD}(I_C)$ and secret data m from $O_{EMD-S}(M)$.

(EMD-2): For each block, calculate $t = f_a(p_1, p_2, \ldots, p_n)$ using Eq.(1).

$$f_a(p_1, p_2, ..., p_n) = \sum_{i=1}^{n} (p_i \times i) \bmod (2n+1). \tag{1}$$

(EMD-3): Calculate the difference $d = (m - t) \bmod (2n+1)$.

(EMD-4): If $(d = 0)$ then $(y_1, y_2, \ldots, y_n) = (p_1, p_2, \ldots, p_n)$,
else if $(n > d)$, then $(y_1, y_2, \ldots, y_d, \ldots, y_n) = (p_1, p_2, \ldots, p_d + 1, \ldots, p_n)$,
else $(y_1, y_2, \ldots, y_{(2n+1)-d}, \ldots, y_n) = (p_1, p_2, \ldots, p_{(2n+1)-d} - 1, \ldots, p_n)$.

(EMD-5): Modify the $(p_1, p_2, \ldots, p_n)$ in I_C by $(y_1, y_2, \ldots, y_n)$ to create I_S

2.2. **The general EMD method.** In order to improve the embedding capacity of the EMD method and allow secret data be transformed from $(2n+1)$ into (2^{n+1})-ary notation, a GEMD data hiding scheme was proposed by Kuo and Wang in 2012 [4]. This approach allows the secret data to use the binary stream directly. Note the improved method proposed by Lee *et al.* [5] where the parameter n is limited to 2 is a special case of Kuo-Wang scheme. In this method, $O_{GEMD}(\cdot)$ is a function which gives all n-tuples $(p_1, p_2, \ldots, p_n)$ obtained from partitioning the image I_C into the non-overlapping n-pixel blocks by scanning each line of pixels from left to right in a top-down manner, as shown in Fig.1, and $O_{GEMD-S}(\cdot)$ is a function which can obtain (2^{n+1})-ary data m from the secret data stream M for each block.

Algorithm GEMD (Embedding Algorithm for Kuo-Wang-scheme):
Input: cover image I_C and binary secret data stream M
Output: stego image I_S

(GEMD-1): Obtain all n-pixel blocks $(p_1, p_2, \ldots, p_n)$ from $O_{GEMD}(I_C)$ and the secret data m from $O_{GEMD-S}(M)$.

(GEMD-2): For each block, calculate $t = f_b(p_1, p_2, \ldots, p_n)$ where

$$f_b(p_1, p_2, ..., p_n) = \sum_{i=1}^{n} p_i \times (2^i - 1) \bmod 2^{n+1} \tag{2}$$

(GEMD-3): Calculate the difference $Dvalue = m - t$.
(GEMD-4): If $Dvalue \leq 2^n$, then $Dvalue' = Dvalue$ and go to (GEMD-5), else let $Dvalue' = 2^{n+1} - Dvalue$ and go to (GEMD-6).

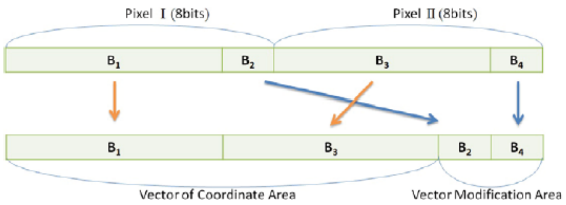

FIGURE 2. Pixel segmentation in a pair of pixels

(GEMD-5): If $Dvalue' = 2^n$, then $p'_n = p_n + 1$, $p'_1 = p_1 + 1$;
else transform $Dvalue'$ to $(V_n V_{n-1} \cdots V_1 V_0)_2$ and
For $i = n$ to 1 do
{If $(V_i = 0 \& V_{i-1} = 0)$ or $(V_i = 1 \& V_{i-1} = 1)$ then $p'_i = p_i$;
else if $(V_i = 0 \& V_{i-1} = 1)$ then $p'_i = p_i + 1$;
else if $(V_i = 1 \& V_{i-1} = 0)$ then $p'_i = p_i - 1$.}
Go to (GEMD-7).
(GEMD-6): Transform $Dvalue'$ to $(V_n V_{n-1} \cdots V_1 V_0)_2$
For $i = n$ to 1 do;
{If $(V_i = 0 \& V_{i-1} = 0)$ or $(V_i = 1 \& V_{i-1} = 1)$ then $p'_i = p_i$;
else if $(V_i = 0 \& V_{i-1} = 1)$ then $p'_i = p_i - 1$;
else if $(V_i = 1 \& V_{i-1} = 0)$ then $p'_i = p_i + 1$.}
(GEMD-7): End.

Although the binary data stream is used directly during the embedding process, it still can only embed an additional $\frac{1}{n}$ secret bit on average for each pixel for the Kuo-Wang scheme. In other words, at best, this scheme cannot embed more than two bits of secret data for each pixel.

2.3. **Pixel Segmentation Strategy and Vector of Coordinate Area Embedding Method.** In 2008, Lee *et al.* proposed a data hiding method using pixel segmentation strategy and VCA (vector of coordinate area) [6]. This method segments a pair of pixels into two parts named VCA and VMA as shown in Fig.2 where $B_2(B_4)$ represents the VMA block composed of the first(second) pixels and can be more than 1 bits. The VCA block is composed of the remaining $B_1(B_3)$ pixels and has $8 - |B_2|(8 - |B_4|)$ bits where $|B_2|(|B_4|)$ is the bit length of $B_2(|B_4|)$.

VCA represents a projection into a space and VMA represents a guide map. Using the guide map in the projected space, the secret can be extracted by following the map. The following notations are used to introduce the LCW-scheme.

$O_{LCW}(\cdot)$**:** : Obtain all 2-tuples (p_1, p_2) from partitioning the image I_C into non-overlapping 2-pixel blocks by scanning each line of pixels from left to right in a top-down manner as shown in Fig.3.

$O_{LCW-S}(\cdot)$**:** : Obtain (2n+1)-ary data m from partitioning the secret data stream M for each block.

Algorithm LCW (Embedding Algorithm for LCW Scheme):
Input: cover image I_C and binary secret data stream M
Output: stego image I_S

(LCW-1): Use $O_{LCW}(I_C)$ to obtain all 2-pixel blocks $(p1, p2)$ and secret data m from $O_{LCW-S}(M)$.
(LCW-2): Determine replacement P_{VMA} where P_{VMA} is the value of the VMA bits.

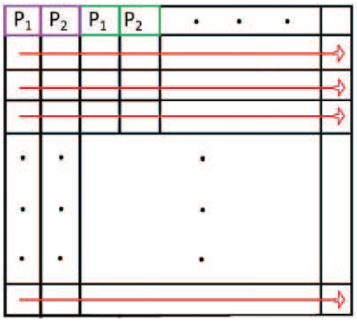

FIGURE 3. The embedding data sequence for pixel segmentation strategy

(LCW-3): Use a distribution function to randomly permute all cover pixels with a secret key.
(LCW-4): For each pair (p_1, p_2),
Step 1: Divide pixels into two areas, VCA and VMA shown as Fig.2.
Step 2: Use a random function to generate a vector of coordinates with VCA denoted as $(g_1, g_2, \ldots, g_n)$ where g_i can be any positive integer, for $i = 1, 2, \ldots, n$, where $n = 2^{P_{VMA}-1} - 1$.
Step 3: Calculate $t = f_c(g_1, g_2, \ldots, g_n)$ and the difference value $d = (m - t) \bmod (2_n + 1)$ where

$$f_c(g_1, g_2, ..., g_n) = \sum_{i=1}^{n} (g_i \times i) \bmod (2n + 1). \tag{3}$$

Step 4: Follow the EMD-algorithm to find the code and fill the code in VMA.

The advantages of the LCW-scheme include achieving a degree of security, maintaining good stego image quality and not having overflow issue. The embedding rate for one pixel is $1/2 \times (\log_2 (2n + 1))$ bits. For example, if $P_{VMA} = 2$, then $n = 3$ and the embedding rate is $1/2 \times (\log_2 7) = 1.4$ bpp. It has since been established that space defined for VMA was insufficient and additional communication between sender and receiver needed for the variable codes.

3. **The Proposed Data Hiding scheme.** Lee *et al.*'s data hiding scheme enhances the embedded secret data capacity using pixel segmentation strategy and EMD. However, it has insufficient space for utilization and initially requires both sender and receiver to have the same code table. In order to remove this disadvantage and embed the binary secret data directly, a novel data hiding scheme based on pixel segmentation strategy and GEMD will be proposed in this section. Specifically, this proposed method makes full use of space which increases embedding capacity without the requirement of previous

communication. At the same time, the overflow problem does not affect this scheme. The embedding algorithm for this scheme is referred to as KC-GEMD. The definition of $O_{KC-GEMD}(\cdot)$ and $O_{KC-GEMD-S}(\cdot)$ are as follows:

$O_{KC-GEMD}(\cdot)$**:** obtain all n-tuples $(p_1, p_2, p_3, \ldots, p_n)$ from partitioning the image I_C into the non-overlapping n-pixel blocks by scanning each line of pixels from left to right in a top-down manner, as shown in Fig.1.

$O_{KCGEMD-S}(\cdot)$**:** obtains 2^{n+1}-ary data m from partitioning the secret data stream M for each block.

Algorithm KC-GEMD (Embedding Algorithm for our proposed scheme):
Input: cover image I_C and binary secret data stream M
Output: stego image I_S

(KC-GEMD-1): Obtain all 2-pixel blocks (p_1, p_2) from I_C by using $O_{KC-GEMD}(I_C)$ and secret data m from $O_{KC-GEMD-S}(M)$.

(KC-GEMD-1): For each block (p_1, p_2),
Determine the value P_{VMA} for VMA and divide each pair of pixels into two areas, VCA and VMA shown as Fig.2.

Step 1: Calculate n for generating vector of coordinates, i.e., $n = (P_{VMA} - 1)$.
Step 2: Allocate the VCA evenly according to n, and generate n vectors denoted as $(g_1, g_2, \ldots, g_n)$.
Step 3: Calculate the reference value t by using Eq.(2).
Step 4: Calculate the difference value $Dvalue = m - t$.
Step 5: Fill Dvalue in VMA.
Step 6: Reconstruct stego pixels (p_1', p_2') by returning blocks back to original position, i.e., $(B_1||B_2, B_3||B_4)$.

Example 3.1. *Let the pixel pair* (174,105) $= (10101110, 01101001)_2$ *for a block,* $P_{VMA} = 4$ *and secret data is* $(1111)_2$. *Therefore, we can get the* $VCA = (101011)_2||(011010)_2$ *and* $VMA = (10)_2||(01)_2$ *from pixel-pair* (174, 105).

The stego pixel-pair is obtained by using the following steps:

Step 1: Compute $n = (4 - 1) = 3$.
Step 2: Generate 3 vectors $(g_1, g_2, g_3) = (1010, 1101, 1010)_2$.
Step 3: Compute $t = f_b(g_1, g_2, g_3) = (111)_2$.
Step 4: Calculate the difference value $D_{value} = (1111 - 111)_2 = (1000)_2$.
Step 5: Fill D_{value} in $VMA = (1000)_2$.
step 6: Get stego pixel-pair $(174, 104) = (10101110, 01101000)_2$ from modified the pixel-pair group.

Similarly, the KC-GEMD algorithm can be extended to use more than two pixels each time(for example 3.2) to embed the secret data.

Example 3.2. *Let three pixels* $(174, 104, 179) = (10101110, 01101000, 10110011)_2$ *for a block,* $P_{kn} = 6$ *and secret data is* $(110100)_2$. *Therefore,* $VCA = (101011)_2||(011010)_2||(101100)_2$ *and* $VMA = (10)_2||(00)_2||(11)_2$ *from pixels* (174, 104, 179).

The stego pixels are obtained by using the following steps:

Step 1: Compute $n = (6 - 1) = 5$.
Step 2: Generate 3 vectors $(g_1, g_2, g_3, g_4, g_5) = (1010, 1101, 101, 100)_2$.
Step 3: Compute $t = f_b(g_1, g_2, g_3, g_4, g_5) = (111110)_2$.
Step 4: Calculate the difference value $D_{value} = (110100 - 111110)_2 = (-1010)_2$.
Step 5: Fill D_{value} in the $VMA = (110110)_2$.
Step 6: Get stego pixel-pair (175, 105, 178) $= (10101111, 01101001, 10110010)_2$

Algorithm KCE-GEMD (Extract Secret Data Algorithm for our proposed scheme)
Input: stego image I_S.
Output: secret data M.

(KCE-GEMD-1): Obtain all 2-pixel blocks (p'_1, p'_2) from I_C by using the $O_{KC-GEMD}(I_S)$.
(KCE-GEMD-2): For each block,

TABLE 1. Pixel modified quality is dependent on D_{value}

When $D_{value} < 8$								
Binary$(b_4b_3b_2b_1)$	0000	0001	0010	0011	0100	0101	0110	0111
$g_3g_2g_1$	000	001	$01-1$	010	$1-10$	$1-11$	$10-1$	100
When $D_{value} \geq 8$								
Binary$(b_4b_3b_2b_1)$	1111	1110	1101	1100	1011	1010	1001	1000
2's complement	0001	0010	0011	0100	0101	0110	0111	
$g_3g_2g_1$	$00-1$	$0-11$	$0-10$	-110	$-11-1$	-101	-100	101

Step 1: Determine P_{VMA} for VMA and divide each pixel-pair into two areas, VCA and VMA as shown in Fig.2.
Step 2: Calculate value $n = P_{VMA} - 1$ for generating vector of coordinates.
Step 3: Allocate the VCA evenly according to n, and generate n vectors denoted as $(g_1, g_2, \ldots, g_n)$.
Step 4: Compute $(g'_1, g'_2, \ldots, g'_n)$ from $(g_1, g_2, ..., g_n)$ using VMA following the GEMD method.
Step 5: Calculate the secret $d_{10} = f_b(g'_1, g'_2, \ldots, g'_n)$
Step 6: Convert decimal number d_{10} to binary m_2.
(KCE-GEMD-3): Concatenate m_2 from each block to recover the original secret data M .

Example 3.3. *If the stego pixel-pair is* $(10101110, 01101000)_2$ *for a block,* $P_{VMA} = 4$, *and* $n = (4-1) = 3$ *then we can recover the secret data is* $(1111)_2$.

Step 1: Allocate the VCA evenly according to $n = 3$, and generate 3 vectors denoted as $(g_1, g_2, g_3) = ((1010)_2, (1101)_2, (1010)_2) = (10, 13, 10)$.
Step 2: Compute (11, 13, 11) from (10, 13, 10) with $(1000)_2$ following GEMD method shown as Table 1.
Step 3: Calculate the secret $d_10 = f_b(11, 13, 11) = f_b(11, 13, 11) = (15)_10$.
Step 4: Convert decimal number $(15)_10$ to binary $(1111)_2$.

4. **Experimental results and Performance discussion.** In this section, we use our proposed scheme for simulations and show their results. The experiment hardware environment is a personal computer with an Intel Core Duo 2 E4600 2.4 (GHz) CPU having 2G RAM. The operating system is Windows XP Professional and the experiment software is MATLAB. The grayscale image is 512×512 pixels and the stego image quality is represented by peak signal to noise ratio (PSNR). Greater PSNR means the smaller difference between the cover image and stego image, i.e., The higher the PSNR, the more similar the stego image is to the cover image. If PSNR is lower than 30dB, the stego image variation from the cover image can be visually distinguished. The PSNR and the mean square error (MSE) is calculated as follows:

Method Image ($\|B_2\|$, $\|B_4\|$)	LCW-scheme [6]	Our scheme
Lenna(1,2)	 46.6 dB	 46.4 dB
embedded	367,971 bits	393,216 bits
Lenna(2,2)	 44.3 dB	 44.2 dB
embedded	512,085 bits	524,288 bits
Lenna(2,3)		 40.03 dB
embedded		655,360 bits

FIGURE 4. Comparison between LCW-scheme and the proposed scheme table

Method Image ($\|B_2\|$, $\|B_4\|$)	Lena	Pepper	Baboon
(1,2)	 46.383 dB	 46.388 dB	 46.376 dB
embedded	393,216 bits	393,216 bits	393,216 bits
(2,2)	 44.164 dB	 44.164 dB	 44.157 dB
embedded	524,288 bits	524,288 bits	524,288 bits
(2,3)	 40.026 dB	 40.016 dB	 40.020 dB
embedded	655,360 bits	655,360 bits	655,360 bits

FIGURE 5. Simulation results using the proposed method for three picture samples

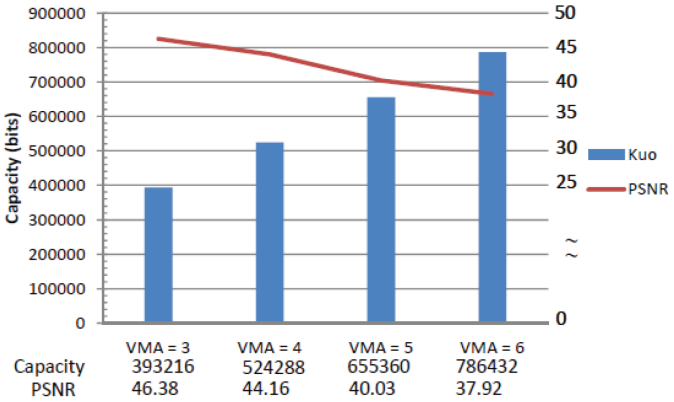

FIGURE 6. Embedding data capacity for different VMA value (Lena)

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE}, \tag{4}$$

$$MSE = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} \left(I\left(x, y\right) - I'\left(x, y\right) \right)^2, \tag{5}$$

where M and N represent the length and width of the image, respectively. The stego images were produced in a raster-scan order.

Fig.4 shows disguised image quality decline with more embedded bits where $|B_2|$ represents the VMA bits for first pixel and $|B_4|$ for the second pixel. The method proposed by Lee *et al.* cannot generate enough vectors from VCA after P_{VMA} is five and cannot produce the stego image. Conversely, our proposed method does not require as many vectors to reach comparable embedding capacity to LCW-scheme. The respective cost to image quality for applying our method verses LCW is only 0.1dB or 0.2dB which contributes to imperceptible difference.

Fig.6 shows the embedding data capacity is inversely proportional to the PSNR value and determines the embedding capacity to employ in order to maintain stego image's quality. Then, the performance comparison figure is shown as Fig.7.

The following results arise from the Fig.7:

1. For the LSB method, the highest stego image quality is near 53dB and decreases when the embedding capacity increases. Specifically, the stego image variation from the cover image can be visually distinguished when the embedding capacity is larger than 4bpp.
2. For the EMD method, the stego image quality is always above 52dB and the largest embedding capacity is 1.16bpp. The stego image quality increases and the embedding capacity will be decreases when the embedding pixels numbers increases.
3. For the GEMD method, the stego image quality is above 50dB and the largest embedding capacity is 1.5bpp. Similarly, its embedding capacity is inversely proportional to the embedding pixels numbers but it also maintains at least 1bpp when the embedding group pixels increase.
4. For Lee *et al.*'s method, although the stego image quality is worse than EMD scheme, it has better embedding capacity performance. With acceptable visibility, the embedding capacity achieves 1.95bpp. In the condition where it only uses 16 bits (2

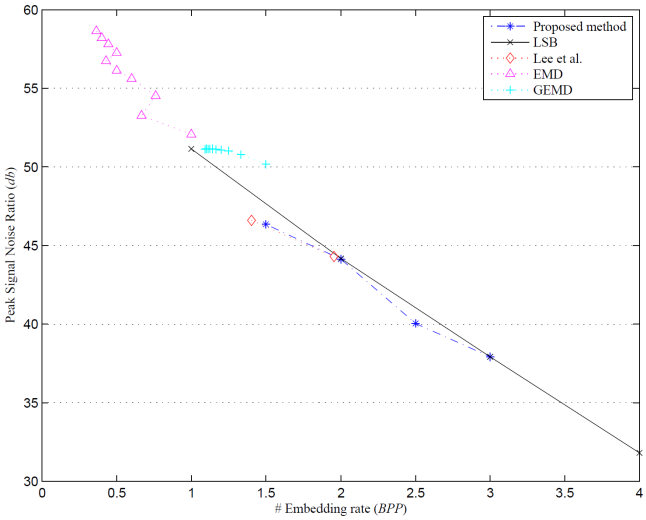

FIGURE 7. Performance comparison between several data hiding schemes [4, 5, 9, 13] on the host image Lena

pixels), not enough vectors can be generated. So, LCW-scheme was unable to achieve larger embedding capacity.

5. For Our proposed scheme, it can generate enough vectors under the same conditions as the LCW-scheme. Because the extraction function is different from the LCW-scheme, our proposed scheme, needs less vectors to achieve similar embedding capacity. So, our proposed scheme can not only to achieve embedding capacity 3bpp but also to keep at least 37dB for stego image quality.

The following Table 2 compares the message capacity and features of the LCW-scheme and our proposed scheme. The LCW-scheme requires generation of many vectors exhausting the VCA space for particular pixel segmentation combinations even though the VCA space is larger than the VMA space. On the hand, the proposed scheme is applicable in all cases where the VCA space is larger than the VMA space since it uses less vectors. Thus, our method simultaneously enlarges hidden capacity and also does not require the variable codes to be exchanged before communication.

5. Conclusions.

We proposed an improvement of the GEMD embedding method using pixel segmentation. The major goal of this method is to improve the characteristics of the LCW-scheme. According to experimental results, our method not only to enlarge the capacity but also to avoid the overflow problem. Although the result of RS analysis is not satisfactory, secret message still not be obtained. Nevertheless we take this issue as the future work.

Acknowledgement. The authors extend appreciation to the anonymous reviewers for providing invaluable comments during the revision process.This work was supported by NSC grant number 102-2218-E-150-001.

REFERENCES

[1] M. Arabzadeh, and M. R. Rahimi, Reversible data hiding scheme based on maximum histogram gap of image blocks, *KSII Trans. on Internet and Information Systems*, Vol. 6, No. 8, pp. 1964-1981, 2012.

[2] J. Fridrich, M. Goljan, and R. Du, Reliable detection of LSB based image steganography, *Proc. of the ACM Workshop on Multimedia and Security*, pp. 27-30, 2001.

TABLE 2. Characteristics comparison table for LCW and the proposed scheme

Method	LCW-scheme [6]	Our scheme
Pixel processing	EMD and segmentation	GEMD and segmentation
Vector of coordinate area	$n = 2^{P_{VMA}-1} - 1$	$n = P_{VMA} - 1$
Embedding function	$\sum_{i=1}^{n}(p_i \times i) \bmod (2n+1)$	$\sum_{i=1}^{n} p_i \times (2^i - 1) \bmod 2^{n+1}$
Code table fabrication	Required the two sides	Unused
Message capacity (Bits Per Pixel)	$\frac{1}{2}\log_2(2^{P_{VMA}-1} - 1)$	$\frac{1}{2}\log_2(2^{P_{VMA}}) \sim \frac{P_{VMA}}{2}$

[3] W. C. Kuo, L. C. Wuu, C. N. Shyi, and S. H. Kuo, A data hiding scheme with high embedding capacity based on general improving exploiting modification direction method, *Proc. of The 9th International Conference on Hybrid Intelligent Systems*, pp. 69-73, 2009.
[4] W. C. Kuo, and C. C. Wang, Data hiding based on generalized exploiting modification direction method, *Imaging Science*, vol. 61, no. 6, pp. 484-490, 2013.
[5] C. F. Lee, Y. R. Wang, and C. C. Chang, A steganographic method with high embedding capacity by improving exploiting modification direction, *Proc. of the 3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 497-500, 2007.
[6] C. F. Lee, C. C. Chang, and K. H. Wang, An improvement of EMD embedding method for large payloads by pixel segmentation strategy, *Image and Vision Computing*, vol. 26, no. 12, pp. 1670-1676, 2008.
[7] C. W. Lee, and W. H. Tsai, A lossless data hiding method by histogram shifting based on an adaptive block division scheme, *P.S.P. Wang (Ed.), Pattern Recognition and Machine Vision-in Honor and Memory of the Late Professor King-Sun Fu*, pp. 1-14, 2011.
[8] X. Liao, Q. Y. Wen and J. Zhang, A steganographic method for digital images with four-pixel differencing and modified LSB substitution, *Journal of Visual Communication and Image Representation*, vol. 22, no. 1, pp. 1-8, 2011.
[9] J. Mielikainen, LSB matching revisited, *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285-287, 2006.
[10] J. Wang, Y. Sun, H. Xu, K. Chen, H. J. Kim, and S. H. Joo, An improved section-wise exploiting modification direction method, *Signal Processing*, vol. 90, no. 11, pp. 2954-2964, 2010.
[11] X. T. Wang, C. C. Chang, C. C. Lin, and M. C. Li, A novel multi-group exploiting modification direction method based on switch map, *Signal Processing*, vol. 92, no. 6, pp. 1525-1535, 2012.
[12] A. Westfeld, and A. Pfitzmann, Attacks on steganographic systems, *Proc. of the 3th International Workshop on Information Hiding*, LNCS 1768, Springer, pp. 61-76, 1999.
[13] X. Zhang, and S. Wang, Efficient steganographic embedding by exploiting modification direction, *IEEE Communications Letters*, vol. 10, no. 11, pp. 1-3, 2006.