Journal of Information Hiding and Multimedia Signal Processing ©2015 ISSN 2073-4212
Ubiquitous International Volume **6**, Number **4**, July **2015**

A Generalization of Fully Exploiting Modification Directions Data Hiding Scheme

Wen-Chung Kuo
Department of Computer Science and Information Engineering
National Yunlin University of Science & Technology
simonkuo@yuntech.edu.tw
The corresponding author

Ming-Chih Kao
Industrial Technology Research Institute, R.O.C.

Chin-Chen Chang
Department of Information Engineering and Computer Science,
Feng Chia University, 40724, R.O.C.
Department of Computer Science and Information Engineering,
Asia University, Taichung, Taiwan, 41354, R.O.C.
Email:alan3c@gmail.com
Corresponding author

Received May, 2014; revised March, 2015

Abstract. *Many FEMD - type (fully exploiting modification directions) data hiding schemes have been proposed to improve stego image data hiding capacity from* 1 *bpp to* 4.5 *bpp. In this paper, we will summarize these FEMD-type approaches and then propose a general fully exploiting modification directions data hiding scheme which can directly be used to represent some existing FEMD-type data hiding schemes and provide new features while solving the overflow problem that affects other methods. The simulation results and performance analysis show that the proposed scheme not only builds an adaptive FEMD-type data hiding scheme but also retains all FEMD-type data hiding scheme characteristics such as good embedding capacity and stego image quality without using extra memory resources and having reasonable encoding time.*

Keywords: Data hiding, General Fully Exploiting Modification Direction(GFEMD), Embedding capacity, Stego image

1. **Introduction.** Following the rapid growth of network and smart phone technology, people can share digital multimedia such as photos or video easily. However, there are many threats such as illegal duplication, forgery and spoofing when this data is transmitted through public channels. Therefore, how to protect the digital data security has become very important. An interesting way to solve this problem is to hide personal data behind a meaningful image such that an unintended observer will not be aware of the existence of the hidden secret message.

Until now, many EMD-type(Exploiting Modification Direction) data hiding schemes[2, 3, 4, 6, 8, 9] have been proposed since Zhang and Wang [11] firstly proposed a data hiding scheme based on EMD method in 2006. From experimental results of these literatures, we can find that EMD-type schemes keep good embedding capacity and stego-image quality. However, the maximum embedding capacity is 1.5 bpp(bits per pixel). Recently, Kieu and Chang[1] provided a new embedding function called fully exploiting modification direction

(FEMD) and then a robust data hiding scheme based on the FEMD method is proposed to improve the original data hiding capacity from 1 bpp to 4.5 bpp. According to the Kieu-Chang scheme, a new extraction function is proposed to setup the search matrix and then use this reference table method to embed the secret data. Lately, two new data hiding schemes without the extra memory based on formula FEMD or square FEMD were proposed [5, 7]. According to our analysis, there are three major goals for FEMD-type schemes: (1) providing good embedding capacity; (2) acceptable stego image quality without using extra memory resources; (3) and to perform within a reasonable computing time. In order to achieve the above characteristics, a general extraction function to summarize the FEMD-type schemes is proposed in this paper. The simulation results and performance analysis show the proposed scheme not only maintains all FEMD-type data hiding characteristics but also provides good embedding capacity and acceptable stego image quality without using extra memory resources. Additionally, it performs within a reasonable computing time and allows adaptive selection of the modulo and its power.

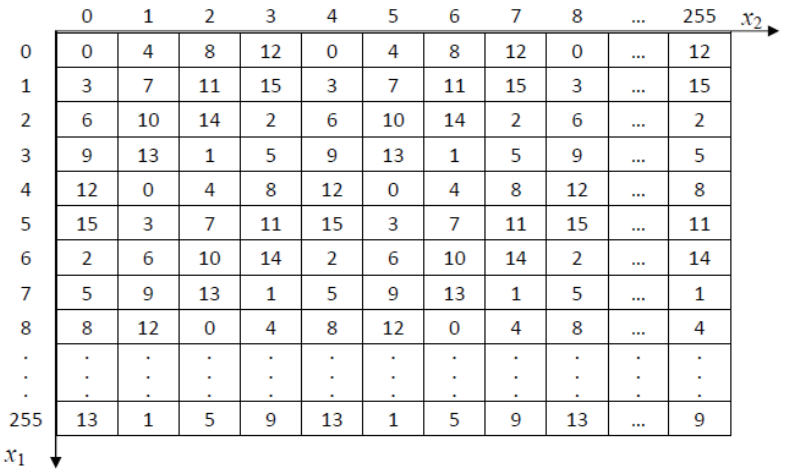
The result of this paper is organized as follows: Section 2 will introduce the FEMD-type data hiding schemes. Then, the general fully exploiting modification direction function and embedding procedure are proposed in Section 3 and experimental results and performance discussions are shown in Section 4. Finally, concluding remarks will be given in Section 5.

2. **Review Current FEMD-type Data Hiding Schemes.**

2.1. **The Full EMD Scheme.** In 2011, Kieu and Chang[1] proposed another extraction function $F(x_i, x_{i+1})$ shown as Eq.(1)

$$F(x_i, x_{i+1}) \quad = \quad (s-1) \times x_i + s \times x_{i+1} \pmod{s^2}. \tag{1}$$

They used a 256×256 s-matrix to represent the result of $F(x_i, x_{i+1})$, where the value of the x_ith row and the x_{i+1}th column in s-matrix is $F(x_i, x_{i+1})$ for $x_i = 0, 1, \ldots, 255$ and $x_{i+1} = 0, 1, \ldots, 255$. In other words, the symbol $s[x_i][x_{i+1}]$ is used to represent the matrix, i.e. $s[x_i][x_{i+1}] = F(x_i, x_{i+1})$. The result of the extraction function $F(x_i, x_{i+1})$ with $s = 4$ is shown in Fig.1.



	0	1	2	3	4	5	6	7	8	...	255	x_2
0	0	4	8	12	0	4	8	12	0	...	12	
1	3	7	11	15	3	7	11	15	3	...	15	
2	6	10	14	2	6	10	14	2	6	...	2	
3	9	13	1	5	9	13	1	5	9	...	5	
4	12	0	4	8	12	0	4	8	12	...	8	
5	15	3	7	11	15	3	7	11	15	...	11	
6	2	6	10	14	2	6	10	14	2	...	14	
7	5	9	13	1	5	9	13	1	5	...	1	
8	8	12	0	4	8	12	0	4	8	...	4	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
255	13	1	5	9	13	1	5	9	13	...	9	
x_1												

FIGURE 1. The embedding matrix for $3 \times x_1 + 4 \times x_2 \pmod{16}$

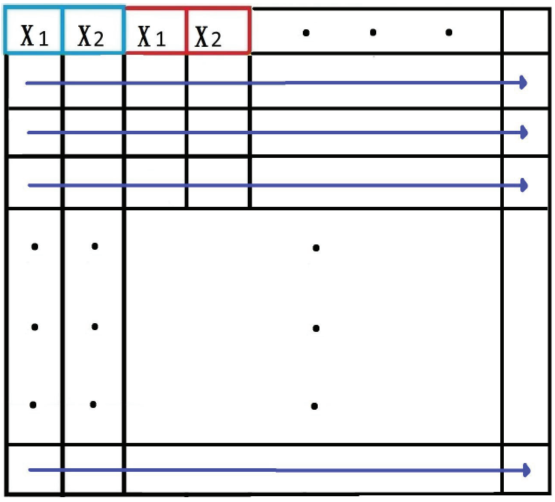


Figure 2. The embedding data sequence for FEMD or FFEMD

Subsequently, Kieu and Chang used s-matrix to embed the secret data. In other words, the k-bit secret data can be embedded into pair (x_i, x_{i+1}) of the cover image using s-matrix with the search range r, where $k = \lfloor \log_2 s^2 \rfloor$ and $r = \lfloor s/2 \rfloor$. $O_{FEMD}(\cdot)$ obtains all 2-tupes (x_1, x_2) from partitioning the image I_C into non-overlapping 2-pixel blocks by scanning from the left side to right side and from top to down, as shown in Fig.2. $O_{s-FEMD}(\cdot)$ obtains s^2 -ary data m from partitioning the secret data stream M for each block.

Algorithm FEMD-scheme (Embedding Algorithm for Kieu-Chang Scheme)
Input: cover image I_C and the binary secret data M
Output: stego image I_S

(KC-1): Generate the 256 × 256 s-matrix by using Eq.(1).
(KC-2): Obtain all 2-pixel blocks (x_1, x_2) from I_C by using $O_{FEMD}(I_C)$.
(KC-3): Obtain s_{temp} from $O_{s-FEMD}(M)$ for each block.
(KC-4): If $s[x_1][x_2] = s_{temp}$, then $(y_1, y_2) = (x_1, x_2)$;
Otherwise, use the s-matrix and minimum distortion strategy to get the local optimal solution $s[p][q] = s_{temp}$ and $(y_1, y_2) = (p, q)$.

The authorized receiver can recover secret data from the stego image I_s since they know the modulus parameter s. For details on the extraction procedure, please refer to [1].

2.2. **A formula FEMD method.** In order to reduce memory use, a formula FEMD method is proposed by Kuo and Kao. In [7], they proposed the KG-Theorem to find the pixel pair x_1 and x_2 such that $d = F_f(x_1, x_2) = (s-1) \times x_1 + s \times x_2 \bmod s^2$ when d and modulus s are given. Here, we introduce the KG-Theorem as the following:

Theorem 1(KG-Theorem) If d and modulus s are given, then (x_i, x_{i+1}) is derived directly, i.e., $x_i = (s-1) \times d \bmod s$ and $x_{i+1} = [\frac{(d-(s-1)\times x_i)}{s}] \bmod s$ such that $d = (s-1) \times x_i + s \times x_{i+1} \bmod s^2$.

Example 1. If $d = F_f(x_1, x_2) = 7$ and $s = 3$, then the pair $(x_1, x_2) = (2, 1)$ from the following steps.

Step 1: From $s = 3$, we can get the extraction function $F(x_1, x_2) = 2 \times x_1 + 3 \times x_2 \bmod 9$.
Step 2: Compute $x_1 = 2 \times 7 \bmod 3 = 2$.
Step 3: Calculate $x_2 = [\frac{(7-2\times 2)}{3}] \bmod 3 = 1$.

Algorithm FFEMD-scheme (Embedding Algorithm for Kuo-Kao Scheme)
Input: cover image I_C and binary secret data M
Output: stego image I_S

```
(I-1): Obtain all 2-pixel blocks $(x_1, x_2)$ from $I_C$ and $O_{FEMD}(I_c)$.
(I-2): Obtain $m_1$ from $O_{s-FEMD}(M)$ for each block.
(I-3): Compute $d = F(x_1, x_2) = (s-1) \times x_1 + s \times x_2 \bmod s^2$.
(I-4): If $d = m_1$ then $(y_1, y_2) = (x_1, x_2)$;
    Otherwise, get $\{(y_{1,1}, y_{1,2}), (y_{2,1}, y_{2,2}), (y_{3,1}, y_{3,2}), (y_{4,1}, y_{4,2})\}$ by using the KG-Theorem.
(I-5): Compute all distortions as follows. $D = \{(|x_1-x|+|x_2-y|)|(x, y) \in \{(y_{1,1}, y_{1,2}),$
    $(y_{2,1}, y_{2,2}), (y_{3,1}, y_{3,2}), (y_{4,1}, y_{4,2})\}\}$.
(I-6): Select a $(x, y)$ with minimum distortion in D, and let stego pixel pair $(y_1, y_2) =$
    $(x, y)$.
```

Example 2. If the cover pixels pair is $(x_1, x_2) = (162, 162)$ and the secret data $m_1 = 0$ when $s = 4$, then the stegoimage pixels pair $(y_1, y_2) = (164, 161)$ by the following steps:

Step 1: $d = F(162, 162) = 3 \times 162 + 4 \times 162 \bmod 16 = 14 \neq m_1$.

Step 2: By using the KG-Theorem, get $\{(160, 160), (160, 164), (164, 161), (164, 165)\}$.

Step 3: Compute all distortions between $\{(160, 160), (160, 164), (164, 161), (164, 165)\}$ and the original pixel pair $(162, 162)$.

Step 4: Select $(164, 161)$ with minimum distortion as the stego pixel pair.

According to their simulation results and performance analysis, they claim that their scheme maintains good embedding capacity and stegoimage quality because their scheme does not use extra memory resources and executes within reasonable computing time. Therefore, the Kuo-Kao data hiding scheme can fit the usage scenarios of limited resource mobile devices.

2.3. **A Square FEMD method.** Recently, Kuo[5] proposed a new extraction function as Theorem 2 in order to enhance the secret data capacity.

Theorem 2(SFEMD-Theorem) If $F_s(x_1, x_2)$ and modulus s are given, then we can find out (x_1, x_2) directly, $x_1 = (s^2 - 1) \times F_s(x_1, x_2) \bmod s^2$ and $x_2 = (\frac{(F_s(x_1,x_2)-(s^2-1)\times x_1)}{s^2}) \bmod s^2$, such that $F_s(x_1, x_2) = [x_1 \times (s^2 - 1) + x_2 \times s^2] \bmod s^4$.

Example 3. If $F_s(x_1, x_2) = 11$ and $s = 3$, then the pair $(x_1, x_2) = (7, 4)$ by the following steps.

Step 1: From $s = 3$, we can get $F_s(x_1, x_2) = 8 \times x_1 + 9 \times x_2 \bmod 81$.

Step 2: Compute $x_1 = 8 \times 11 \bmod 81 = 7$.

Step 3: Calculate $x_2 = (\frac{(11-8\times7)}{9}) \bmod 9 = 4$.

Therefore $F_s(x_1, x_2) = 8 \times x_1 + 9 \times x_2 \bmod 81 = 8 \times 7 + 9 \times 4 \bmod 81 = 11$.

The data hiding scheme based on the SFEMD-theorem was proposed previously and the embedding algorithm of this scheme is similar to [7]. For more information, please refer to [5].

3. **General Formula FEMD-Scheme.** Basically, the FFEMD-scheme and SFEMD-scheme belong to the FEMD-type data hiding schemes. According to the proposed theorem in [1, 5, 7] and the embedding algorithm, they are similar in function. Therefore, we try to summarize and then propose the general formula FEMD theorem and embedding algorithm in this section.

Theorem 3(GFEMD-Theorem) If $F_g(x_1, x_2)$ and modulus s are given, then we can find out (x_1, x_2) directly, $x_1 = (s^k - 1) \times F_g(x_1, x_2) \bmod s^k$ and $x_2 = (\frac{(F_g(x_1,x_2)-(s^k-1)\times x_1)}{s^k}) \bmod s^k$, such that $F_g(x_1, x_2) = [x_1 \times (s^k - 1) + x_2 \times s^k] \bmod s^{2k}$, for $k = 1, 2, \cdots, n$ and $\{x_1, x_2\} \in \{-s^{k-1} + 1, s^{k-1} - 1\}$.

Proof. We use generalized mathematical induction [10], i.e., for all $k = 1, 2, \ldots, n$

$$\begin{cases} x_1 = (s^k - 1) \times F_g(x_1, x_2) \bmod s^k, \\ x_2 = (\frac{F_g(x_1,x_2)-(s^k-1)\times x_1}{s^k}) \bmod s^k \end{cases} \tag{2}$$

is the solution of $F_g(x_1, x_2)$.
From Theorem 3, we have general solutions as the following:

Step 1: When $k = 1$, Eq.(2) is the solution of $F_g(x_1, x_2)$ by Theorem 1.
Step 2: When $k = 2$, Eq.(2) is also the solution by Theorem 2, too.
Step 3: Let $k = 1, 2, \ldots$ and suppose Eq.(2) is the solution of $F_g(x_1, x_2)$ for $k = n$. When $k = n + 1$, then we will discuss the following two cases:

1. If k is odd, then let $k = 2t + 1 = n + 1$, i.e., $2t = n$. So, let $\alpha = s^t$ and use the characteristic of Theorem 2. We have the following equation:

$$\begin{aligned} F_g(x_1, x_2) &= [(s^n - 1) \times x_1 + s^n \times x_2] \bmod s^{2n} \\ &= [(s^{2t} - 1) \times x_1 + s^{2t} \times x_2] \bmod s^{2(2t)} \\ &= [x_1 \times (\alpha^2 - 1) + x_2 \times \alpha^2] \bmod \alpha^4. \end{aligned}$$

2. If k is even, then let $k = 2t = n + 1$, i.e. $n = 2t - 1$. So, let $\beta = s^{2t-1}$ and use the characteristic of Theorem 1. We can find the following equation:

$$\begin{aligned} F_g(x_1, x_2) &= [(s^n - 1) \times x_1 + s^n \times x_2] \bmod s^{2n} \\ &= [(s^{2t-1} - 1) \times x_1 + s^{2t-1} \times x_2] \bmod s^{2(2t-1)} \\ &= [x_1 \times (\beta - 1) + x_2 \times \beta] \bmod \beta^2. \end{aligned}$$

Thus, Eq.(2) is the solution of $F_g(x_1, x_2)$ when $k = n+1$. By the principle of induction, Theorem 3 is proven.

Example 4. If $F_g(x_1, x_2) = 10$, $k = 3$ and $s = 2$, i.e., $F_g(x_1, x_2) = 7 \times x_1 + 8 \times x_2 \bmod 64$, then the pair $(x_1, x_2) = (6, 4)$ by the following steps.

Step 1: From $s = 2$ and $k = 3$, we can get the extraction function $F_g(x_1, x_2) = 7 \times x_1 + 8 \times x_2 \bmod 64$.
Step 2: Compute $x_1 = 7 \times 10 \bmod 8 = 6$.
Step 3: Calculate $x_2 = (\frac{(10-7\times6)}{8}) \bmod 8 = 4$.

Therefore, $F_g(x_1, x_2) = 7 \times x_1 + 8 \times x_2 \bmod 64 = 7 \times 6 + 8 \times 4 \bmod 64 = 10$. Before we introduce the embedding algorithm, there are some notations are defined as following:

$O_{GFEMD}(\cdot)$**:** Obtain all 2-tupes (x_1, x_2) from partitioning the image I_C into non-overlapping 2-pixel blocks by scanning from the left to right and top down, as shown in Fig.2.
$O_{s-GFEMD}(\cdot)$**:** Obtain s^2-ary data m from partitioning the secret data stream M for each block.

Embedding Algorithm:
Input: cover image I_C and the secret data stream M
Output: stego image I_S

(GF-1): Obtain all 2-pixel blocks (x_1, x_2) from I_C and $O_{GFEMD}(I_C)$.
(GF-2): Obtain m_1 from $O_{s-GFEMD}(M)$ for each block.
(GF-3): Calculate $d = F_g(x_1, x_2) = x_1 \times (s^k - 1) + x_2 \times s^k \bmod s^{2k}$.
(GF-4): If $d = m_1$ then $(x_1, x_2) = (y_1, y_2)$, otherwise, compute four neighbor pixel pairs $\{(y_{1,1}, y_{1,2}), (y_{2,1}, y_{2,2}), (y_{3,1}, y_{3,2}), (y_{4,1}, y_{4,2})\}$ by using Theorem 3.
(GF-5): Compute distortions from $D = \{(|x_1 - x| + |x_2 - y|) | (x, y) \in \{(y_{1,1}, y_{1,2}), (y_{2,1}, y_{2,2}), (y_{3,1}, y_{3,2}), (y_{4,1}, y_{4,2})\}\}$.

(GF-6): Select a (x, y) with minimum distortion in D, and let stego pixel pair $(y_1, y_2) = (x, y)$.

Example 5. If the pixels pair $(x_1, x_2) = (163, 166)$ and the secret data $m_3 = 13$ with $s = 2$ and $k = 4$, then the stego image's pixel pair is $(y_1, y_2) = (163, 168)$ by the following steps:

Step 1: Calculate $d = (15 \times 163 + 16 \times 166) \bmod 256 = 237 \neq 13$.
Step 2: Compute four neighbor pixels pair by using Theorem 3 and give $\{(163, 168), (163, 152), (179, 169), (179, 153)\}$.
Step 3: Compute all distortions from $\{(163, 168), (163, 152), (179, 169), (179, 153)\}$.
Step 4: Select $(163, 168)$ with minimum distortion as the stego pixel pair.

3.1. **Extraction Procedure.** The designated receiver can recover the secret data when after obtaining stego image I_S. The extraction algorithm is detailed as following:

Extraction Algorithm:
Input: stego image I_S
Output: binary secret data M

(EI-1): Obtain all 2-pixel blocks (x_1, x_2) from I_S and $O_{GFEMD}(I_S)$.
(EI-2): Compute $m_i = F_g(x_1, x_2) = (s^k - 1) \times x_1 + s^k \times x_2 \bmod s^{2k}$, where i represents each 2-pixel blocks from (EI-1).
(EI-3): Concatenate all m_i then convert them to binary streams to form secret data M.

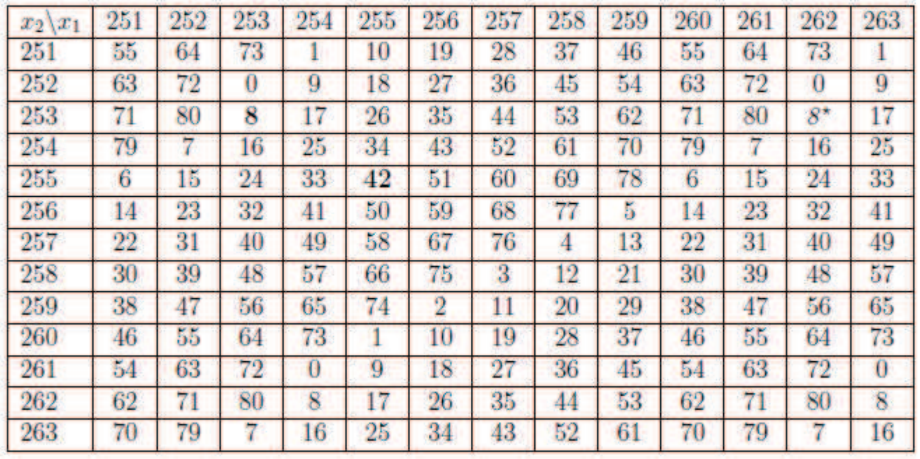
Example 6. If the stego image's pixels pair is $(y_1, y_2) = (163, 168)$, then the secret data $m_1 = 13$ by using the extraction function $m_1 = F_g(163, 168) = 15 \times 163 + 16 \times 168 \bmod 256 = 13$ when $s = 2$ and $k = 4$.

3.2. **Overflow problem solution.** In general, overflow will happen when x_i or x_{i+1} is $[0, r]$ or $[255 - r, 255]$ during the embedding procedure where $r = \lfloor \frac{s}{2} \rfloor$. For simplicity, $x_i(x_{i+1})$ can be shifted to $r(255 - r)$ when $x_i(x_{i+1})$ is in $[0, r]([255 - r, 255])$ to solve the overflow problem in the KC scheme, though this solution will reduce stego image quality[8]. In this subsection, an improved method similar to [7] will be proposed in our embedding procedure. In our embedding algorithm, we compute the difference between a cover pixel pair (x_i, x_{i+1}) and a stego image pixel pair (y_i, y_{i+1}) by adding or decreasing one s^k first. Therefore, our embedding algorithm produces 4 stego image pixel pairs distributed on both sides of the cover pixel pair (x_i, x_{i+1}). Thus, we can modify the embedding algorithm slightly to overcome the overflow problem.

Example 7. If the cover pixels pair is $(x_1, x_2) = (255, 255)$, the secret data $m_2 = 8$, $s = 3$ and $k = 2$, i.e., $F_g(x_1, x_2) = 8 \times x_1 + 9 \times x_2 \bmod 81$, then the stego image pixels pair $(y_1, y_2) = (253, 253)$ by the following steps:

Step 1: Compute $d = F_g(255, 255) = 8 \times 255 + 9 \times 255 \bmod 81 = 42 \neq m_2$.
Step 2: Compute

1. $t = 8 \times 8 \bmod 9 = 1$.
2. $t_{1,1} = 1 - (255 \bmod 9) = -2$.
3. $y_{1,1} = y_{2,1} = 255 - 2 = 253$.
4. $t_{1,2} = [(8 - 8 \times 253)/9] \bmod 9 = 1$.
5. $t_{1,2} = 1 - (255 \bmod 9) = -2$ and $t_{2,2} = -2 + 9 = 7$.
6. $y_{1,2} = 255 - 2 = 253$.
7. Since $y_{2,2} = 255 + 7 = 262$ (overflow), let $y_{2,2} = \infty$.
8. $t_{2,1} = -2 + 9 = 7$.
9. $y_{3,1} = y_{4,1} = 255 + 7 = 262$.
10. Let $(y_{3,1}, y_{3,2}) = (\infty, 0)$ and $(y_{4,1}, y_{4,2}) = (\infty, 0)$.



$x_2\backslash x_1$	251	252	253	254	255	256	257	258	259	260	261	262	263
251	55	64	73	1	10	19	28	37	46	55	64	73	1
252	63	72	0	9	18	27	36	45	54	63	72	0	9
253	71	80	**8**	17	26	35	44	53	62	71	80	$8^{\star}$	17
254	79	7	16	25	34	43	52	61	70	79	7	16	25
255	6	15	24	33	**42**	51	60	69	78	6	15	24	33
256	14	23	32	41	50	59	68	77	5	14	23	32	41
257	22	31	40	49	58	67	76	4	13	22	31	40	49
258	30	39	48	57	66	75	3	12	21	30	39	48	57
259	38	47	56	65	74	2	11	20	29	38	47	56	65
260	46	55	64	73	1	10	19	28	37	46	55	64	73
261	54	63	72	0	9	18	27	36	45	54	63	72	0
262	62	71	80	8	17	26	35	44	53	62	71	80	8
263	70	79	7	16	25	34	43	52	61	70	79	7	16

FIGURE 3. The result of $F_g(x_1, x_2) = 8 \times x_1 + 9 \times x_2 \bmod 81$ from 251 to 263

Step 3: Compute all distortions $\{(253, 253), (253, \infty), (\infty, 0), (\infty, 0)\}$.

Step 4: Select $(253, 253)$ as the stego pixel pair.

From Fig.3, we can find the pixels pair (x_1, x_2) will be changed from $(255, 255)$ to $(253, 253)$ not $(253, 262)$ (with $\star$) after the secret data $m_2 = 8$ is embedded.

Therefore, we can recover the secret data $m_2 = 8$ from the following equation, i.e., $m_2 = F(253, 253) = 8 \times 253 + 9 \times 253 \bmod 81 = 8$.

4. Simulation Performance and Discussion.

In order to confirm the experimental results of the proposed scheme, we used the Peak Signal-to-Noise Ratio (PSNR) to evaluate the image quality of the stego images. The PSNR value is most commonly used as a measure of image quality. It is most easily defined via the Mean-Square-Error (MSE) value. The bigger the PSNR value, the better the image quality. The formulas of PSNR and MSE calculation are defined below:

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE}, \tag{3}$$

$$MSE = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} [P(x, y) - P^*(x, y)]^2}{M \times N}, \tag{4}$$

where $M \times N$ represent the image size, $P(x, y)$ and $P^*(x, y)$ stand for the original pixel value and the stego pixel value in position (x, y).

The proposed scheme was tested on ten 512×512 gray images (Lena, Baboon, F16, Barbara, Boat, Goldhill, Elaine, Tiffany, Pepper and Bridge) as shown in Fig.4. Then, stego images produced when $s = 2$ and $k = 1$ or $k = 5$ are shown in Fig.5. There is no perceivable difference in appearance between cover images and stego images when $s^2 = 4$. However, there are significant visual differences between cover images and stego images when $s = 2$ and $k = 5$. In Table 1, we show simulation results for various modulo s and k.

In order to show the quality of the stego image, the secret data is embedded by using various modulo $s = 2, 3, 4$ and $k = 1, 2, \cdots, 5$ shown as Fig.6. The human vision system can not differentiate variation from original when the stego image PSNR is better than 30dB. Empirically, we employed the Lena image for Eq.(2) to determine acceptable parameters of k for various s. Specifically, for $s = 2$, values of $k = 1, 2, 3, 4$ results in undiscernible stego image differences; for $s = 3$, acceptable k values are $1, 2, 3$; for $s = 4$, $k = 1, 2$. In this instance, a serious overflow problem occurs when $s = 4$ and $k = 5$ (the

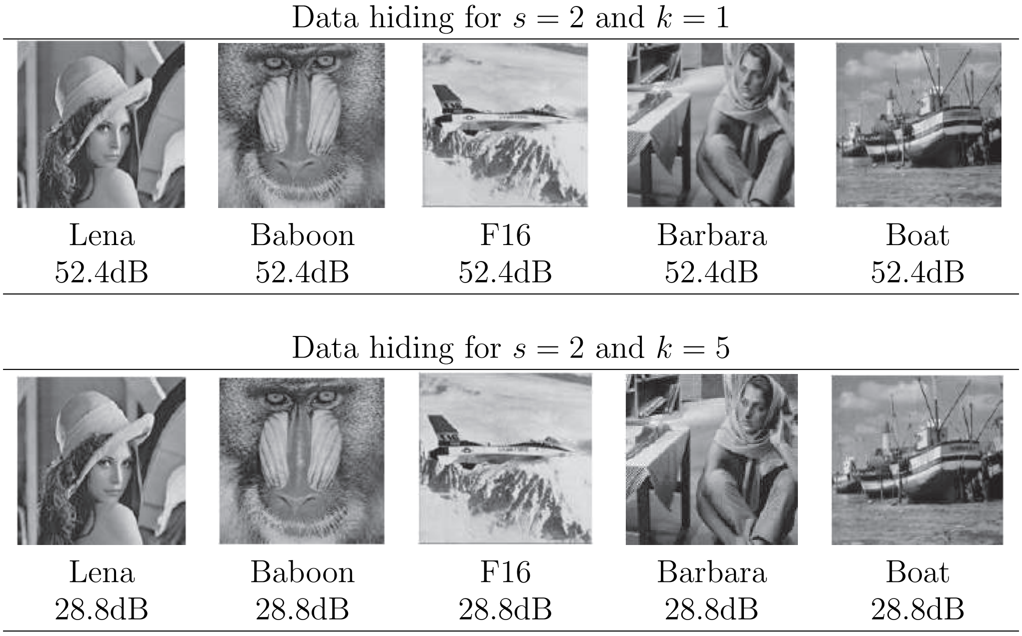
TABLE 1. Simulation results with various modulo and k

Modulus			$s = 2$				$s = 3$			$s = 4$	
k	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 1$	$k = 2$	$k = 3$	$k = 1$	$k = 2$	$k = 3$
s^k	2	4	8	16	32	3	9	27	4	16	64
Lena	52.4	46.8	40.8	34.8	28.8	49.9	39.9	30.4	46.8	34.8	22.8
Baboon	52.4	46.7	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.8
F16	52.4	46.7	40.8	34.8	28.8	49.9	39.9	30.4	46.8	34.8	22.8
Barbara	52.4	46.8	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.8
Boat	52.4	46.8	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.7
Goldhill	52.4	46.8	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.7
Elaine	52.4	46.8	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.7
Tiffany	52.4	46.7	40.8	34.7	28.8	49.9	39.8	30.2	46.8	34.8	22.6
Pepper	52.4	46.7	40.8	34.8	28.8	49.9	39.9	30.3	46.8	34.8	22.6
Bridge	52.4	46.7	40.8	34.8	28.8	49.9	39.8	30.1	46.8	34.8	22.6

(a)Lena (b)Baboon (c)F16 (d)Barbara (e)Boat

(f)Goldhill (g)Elaine (h)Tiffany (i)Pepper (j)Bridge

FIGURE 4. Ten 512 × 512 gray test images



Data hiding for $s = 2$ and $k = 1$

Lena 52.4dB | Baboon 52.4dB | F16 52.4dB | Barbara 52.4dB | Boat 52.4dB

Data hiding for $s = 2$ and $k = 5$

Lena 28.8dB | Baboon 28.8dB | F16 28.8dB | Barbara 28.8dB | Boat 28.8dB

FIGURE 5. Five 512 × 512 gray stego images with $s = 2$ and $k = 1$ or $k = 5$

encircled point in Fig.6) resulting in failure to generate a stego image since the combination of these parameters is not viable. This condition will always occur when the value of s^k exceeds the allowed pixel value.

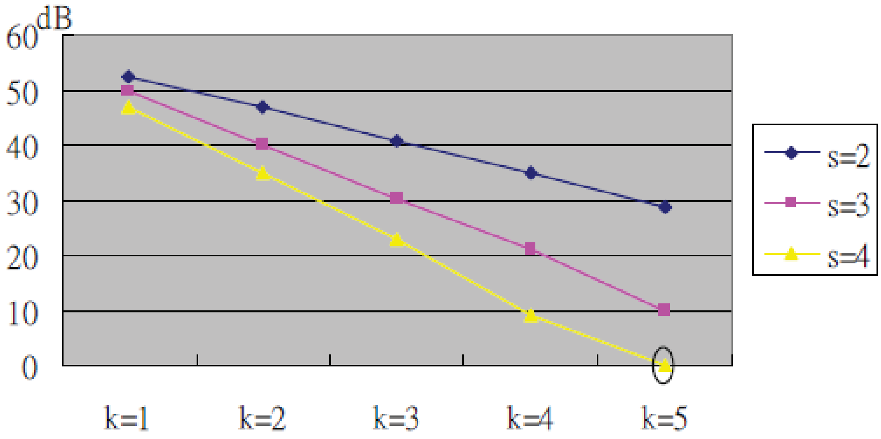


Figure 6. Simulation result table with various modulo and k for Lena

4.1. **Discussions.** Our proposed scheme is also based on FEMD model, though there are many advantages in our scheme compared with the KC scheme. First, in our model all embedding procedures use the GFEMD-Theorem. Secondly, the exact solution to overcome the overflow problem is provided in our scheme. Thirdly, our method does not require any memory space whereas KC scheme requires approximately $k \times 524$ Kbits $(256 \times 256 \times 8 = 524,288)$ to store the embedding matrix.

In Table 2, we summarize some characteristics such as embedding method, the overflow problem solution, adaptive advantage, embedding capacity and the extra storage space between KC-scheme, KK-scheme, Kuo-scheme and our proposed scheme.

In this work, stego images are produced with a selected k value for each stego image. The embedding capacity of our proposed method and the PSNR of the resulting stego image varies based on the selected k as seen in Table 2. As an example, for $s = 2$, the embedding capacity which varies between 1 and 5 bpp corresponds to the PSNR ranges from 52.4 to 28.8 dB according to the selected k value.

However, one interesting feature of the proposed scheme is the ability to use different values of k to embed data in a *single* image. Using different k within a stego image will raise the difficulty of stegoanalysis. Complexity is increased because the value of k is irregular and cannot be predicted for every pixel pair. In other words, variable k skews the normal distribution of embedded bits. Applying this feature requires that the sender and the receiver have some method to share the same k values akin to the concept of one time pad (OTP). Future work may see the possibility of a method to automatically select k values without the need for external secret key sharing but increasing convenience also dilutes the security afforded by the OTP-like mechanism.

5. **Conclusions.** In this paper, a data hiding scheme based on general fully exploiting modification directions method is proposed where secret data can be embedded by using various s and selectable k. It can directly substitute for any of the existing FEMD-type data hiding schemes and it adds a new feature that increases the complexity of guessing hidden data while maintaining the other positive benefits from FEMD-type schemes such as good embedding capacity and acceptable image quality. In addition, the overflow problem, which plagues other embedding approaches, is addressed and resolved in this paper.

Table 2. The comparisons between FEMD-type data hiding scheme and our scheme

Items	KC scheme[1]	Kuo-Kao[7]	Kuo[5]	Our scheme
Embedding method	Matrix and Search	Mathematic	Mathematic	Mathematic
To overcome overflow problem	No	Yes	Yes	Yes
Selectable k per image	No	No	No	Yes
Selectable k within image*	No	No	No	Yes
Embedding capacity when $s = 2$	1 bpp	1 bpp	2 bpp	$1 \sim 5$ bpp
PSNR when $s = 2$	52.4 dB	52.4 dB	46.8 dB	$52.4 \sim 28.8$ dB
Required storage space	$\sim$ 524.3Kb per k	0	0	0

* Future Work Direction

According to the simulation results and performance analysis, our scheme maintains good embedding capacity and stego image quality. It may prove fitting for the usage scenarios of limited resource mobile devices because this scheme does not use extra memory resources and executes within reasonable computing time. The new feature of having selectable k within a stego image looks to increase the complexity and therefore may add to the difficulty of obtaining the secret via stegoanalysis, though this aspect is to be explored and verified in future work.

Acknowledgement: This work was supported in part by the Ministry of Science and Technology of the Republic of China under Contract No. MOST 103-2218-E-492 -006.

REFERENCES

[1] T. D. Kieu, and C. C. Chang, "A steganographic scheme by fully exploiting modification directions," *Expert Systems with Applications*, Vol.38, No.8, pp.10648-10657, 2011.

[2] S. Kim, D. K. Shin, D. G. Shin and X. P. Zhang, "Improved steganographic embedding exploiting modification direction in multimedia communications," Springer, *CCIS 186*, pp.130-138, 2011.

[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," *The Ninth International Conference on Hybrid Intelligent Systems* (HIS2009), pp.69-73, Aug. 2009.

[4] W. C. Kuo, L. C. Wuu, and S. H. Kuo, "The high embedding steganographic method based on general multi-EMD," *The 2012 International Conference on Information Security and Intelligent Control* (ISIC'12), pp.288-291, Aug. 2012.

[5] W. C. Kuo "A data hiding scheme based on square formula fully exploiting modification directions," *Journal of Information Hiding and Multimedia Signal Processing*, Vol.4, No.3, pp.127-136, July 2013.

[6] W. C. Kuo and C. C. Wang, "Data hiding based on generalized exploiting modification direction method," *Imaging Science Journal*, Vol.61, No.6, pp.484-490, 2013.

[7] W. C. Kuo and M. C. Kao, "A steganographic scheme based on formula fully exploiting modification directions,"*IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E96-A, No.11, pp.2235-2243, 2013.

[8] C. F. Lee, Y. R. Wang, C. C. Chang, "A. steganographic method with high embedding capacity by improving exploiting modification direction," *Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing* (IIHMSP07), pp.497-500, 2007.

[9] 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.

[10] K. H. Rosen, Elementary Number Theory and Its Applications, Addison-Wesley, 1993.

[11] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, Vol.10, No.11, pp.1-3, 2006.