IJICTR
International Journal of Information & Communication Technology Research

Volume 13- Number 4 – 2021 (43 -52)

Optimum Group Pixel Matching Strategies for Image Steganography

Alireza Shahanaghi*[1] Mohammad Ali Akhaee[1] Saeed Sarreshtedari[1] Ramin Toosi[1]
[1] School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran
emails: {a.shahanaghi@ut.ac.ir, akhaee@ut.ac.ir, s.sarreshtedari@ut.ac.ir, r.toosi@ut.ac.ir}

Received: 7 August 2021 - Accepted: 3 October 2021

Abstract— **LSB matching techniques are widely applied in the field of image steganography. In such algorithms, pixel values of each group must be changed in a way that a predefined function of the pixel group matches the secret digit. The notational system of the secret digits can be every desired number, as well as the size of the pixel groups. In order to preserve the quality of the stego image, it is desired to limit the changes in the pixel groups as much as possible. Therefore, optimum strategies must be found to match the function of the pixel group to the secret digit with the least possible imposed distortion in terms of mean square error. Having been recently found for pixel pairs, such strategies are found for the larger pixel groups by the proposed method in this paper. Among all the strategies providing the similar minimum MSE value, the one is chosen that helps to preserve the histogram of the original image. Optimum strategies found for all notational systems and pixel group sizes makes the algorithm flexible for various application with different payloads, while it improves the similar techniques in terms of both MSE reduction and histogram preservation, as is confirmed by the experimental results.**

Keywords: component; adaptive pixel group matching (APGM); exploiting modification direction (EMD); least significant beats (LSB) steganography.

Article type: Research Article

Publisher: ICT Research Institute

I. INTRODUCTION

Steganography is the art of communicating a secret message by concealing it into a seemingly normal signal without raising any suspect [1], [2]. Due to ease of access and manipulation, digital images have attracted more interests among all potential carriers. In steganography terminology, a digital image is known as cover image and stego image before and after data embedding respectively. A main category of image steganography tries to embed the secret message by manipulating the least significant bit(s) (LSB) of the cover image.

The simplest algorithm in this category is LSB replacement in which the secret bit replaces the LSB of the pixel value. LSB replacement results in the pair of values (PoV) defect in the histogram of the stego image where the histogram appears in a pairwise format [3] LSB matching (LSBM) steganography is proposed to overcome the PoV issue by randomly increasing or decreasing the pixel value by one if necessary. Later on, steganalysis techniques were developed by another defect in the histogram of the characteristic function (HCF) caused by LSB matching [4], [5]. Recent steganalysis techniques work on a large set of features extracted from the image rather than focusing on specific features or steganographic algorithms [6]-[9].

* Corresponding Author

[DOI: 10.52547/itrc.13.4.43]

The basis of all steganalysis techniques is to detect the distortion imposed to the original image using features that efficiently capture it [10], [11]. Therefore, steganographic algorithms aim at imposing less distortion to the original image according to different criteria, without sacrificing the embedding capacity [12]. For instance, in LSBM every pixel changes with the probability of one half in capacity of one bit per pixel. LSB matching revisited (LSBMR) technique decreases the probability of change per pixel (PCPP) to 0.375 keeping the same embedding capacity [13]. For this sake, LSBMR functions on the pixel pairs rather than individuals. The LSB of the first pixel carries the first secret bit, while a function of both pixels carries the second bit. In this way, pixels remain more intact and the total embedding distortion decreases. This probability is further decreased to one-third, with a scheme working on a group of three pixels [14]. In this scheme, three secret bits are carried by three similar functions of three possible pixel pairs. Li et. al. show that the probability of change per pixel for LSB matching criteria (embedding a capacity of one bit per pixel while pixels change by one or remain unchanged) finds its lowest boundary of about 0.22 in hypothetical case of working on the groups of asymptotically infinite pixels [15].

There have been other efforts as well to reduce the distortion according to other criteria [16]. For example, the histogram of pixel values can be used as a steganalysis feature [4]. In LSB substitution compatible steganography (LSCS), the increasing/decreasing randomness of LSBM is exploited to keep the histogram of the original image as intact as possible [17]. The one-third probability embedding is also modified to keep the histogram of the original image as intact as possible, in addition to keeping the previous feature of lowering the PCPP to one-third [18]. The capacity of some other algorithms varies over the entire image to conceal more bits in edges or textured regions of the image [19]-[21]. Since the human visual system (HVS) is less sensitive to the modifications in such area, these algorithms reduce the visual distortion according to HVS, while keeping the same embedding capacity.

There exist arbitrary cases of modification both in LSBM and LSBMR. While the above-mentioned schemes exploit this randomness to keep the histogram intact, exploiting modification direction (EMD) scheme employs all modification cases for data embedding [22]. Consider a pair of pixels, where there are a total number of five cases for changing one of them by one or keeping them all unchanged. Therefore, a digit in 5-ary notation (0 to 4) can be embedded in this pair, resulting in a capacity of $\log_2(5)/2$ bit per pixel (bpp). It can be easily shown that the capacity of EMD decreases by increasing the size of pixel group. Several algorithms have been proposed to improve EMD in terms of its capacity or total distortion. For instance, diamond encoding (DE) method increases the capacity of pairwise EMD by allowing the total change of pixel pair up to a certain value [23]. Leng et. al. improved the EMD security employing mapping matrix [24]. Exploiting HashedWeightage Array improves EMD using dynamic weightage array in terms of payload and security (EEMDHW) [25]. By reformulation the original EMD embedding algorithm, Ke et. al., introduce parallel EMD, which improves the embedding efficiency and security [26]. In advanced EMD (AEMD) edge masking characteristics of human visual system is exploited to increase the embedding capacity [27]. Adaptive pixel per matching (APPM) method [28] shows that the changes offered by DE are not optimum in terms of mean square error (MSE) distortion imposed to the original image. Moreover, in DE embedding, the secret information can be embedded only using certain notational systems in form of $B = 2k^2 + 2k + 1$for all positive integer values of k. Therefore, APPM offers optimum modification strategies for embedding digits for all *B*-ary notational systems in pixel pairs.

Although APPM offers the embedding strategies for all B-ary notational system, its functionality is limited only to pixel pairs. In this paper, we propose an adaptive pixel triple matching (APTM) method that offers the best modification strategies on pixel triples for every desired notational system. We also show that the original APPM algorithm can be extended to enjoy additional capabilities of histogram preserving. We call this new version of APPM as histogram preserving APPM (HP-APPM), and extend the similar idea to the proposed APTM algorithm to develop the histogram preserving APTM (HP-APTM). Finally, we present the general concept of adaptive pixel group matching (APGM) and similarly its histogram preserving version (HP-APGM). As a result, this paper provides all optimal strategies for embedding information in all **B**-ary notational systems functioning on every desired size of pixel groups. The results show that extending the pixel group size helps to reduce the MSE distortion. On the other hand, more equivalent optimum embedding strategies are available for larger pixel groups that their combination helps to preserve the histogram as much as possible. Moreover, all *B*-ary notational systems are available for all pixel groups that means the extreme payload flexibility of the proposed method.

II. Adaptive Pixel Pair Matching

Functioning on the pixel pairs, adaptive pixel pair matching (APPM) is introduced to embed the payloads more than one bit per pixel efficiently in terms of mean square error (MSE) between the cover and stego image. The original pixel pair (x, y) is considered as a coordinate. Digit s_B in B-ary notational system is embedded in this pixel pair by modifying it into a (x', y') coordinate within a predefined neighborhood $\Phi(x, y)$ such that $f(x', y') = s_B$. In this design, f stands for the predefined extraction function in the form of $f(x, y) = (x + c_B \times y)\ mod\ B$ and c_B is a digit in the range of 0 to B-1.

Every neighborhood $\Phi(x, y)$ consists of B coordinates $(x_i, y_i), i = 0, \dots, B-1$, where

$$\forall s_B \in \{0,1,\dots,B-1\}, \exists (x_i, y_i) \in \Phi(x, y) | f(x_i, y_i) = s_B, \quad i = 0, 1, \dots, B-1 \tag{1}$$

assuming $f(x_i, y_i) = (x_i + c_B \times y_i)\ mod\ B = s_B$ for a certain i, we will have $f(x_i - mB, y_i - kB) = f(x_i, y_i) = s_B$; $m, k \in \mathbb{Z}$ that means the neighborhood Φ(x, y) is not unique. Here we look for a neighborhood

[DOI: 10.52547/itrc.13.4.43]

that minimizes distortion imposed to the original image in terms of MSE, defined as:

$$\text{MSE}_{\Phi(x,y)} = \frac{1}{2B}\sum_{i=0}^{B-1}\left((x_i - x)^2 + (y_i - y)^2\right)$$
$$\text{for all } (x_i, y_i) \in \Phi(x, y) \qquad (2)$$

For each c_B from 0 to $B-1$ the neighborhood with minimum distortion can be found. The optimum c_B among all is the one that results in the minimum $MSE_{\Phi(x,y)}$. Finding the optimum c_B, the corresponding extraction function and neighborhood set are found consequently, as the solution of the optimization problem below:

$$\text{Minimize}: \sum_{i=0}^{B-1}(x_i - x)^2 + (y_i - y)^2$$
$$\text{Subject to}: f(x_i, y_i) \in \{0, 1, \ldots, B-1\}$$
$$f(x_i, y_i) \neq f(x_j, y_j),\ if\ i \neq j$$
$$\forall\ 0 \leq i, j \leq B-1 \qquad (3)$$

Each pixel pair (x, y) and its neighborhood Φ(x, y) forms a certain shape in the Cartesian coordinates. As a general rule, optimization results in picking the most concentrated neighborhoods. For instance, consider two possible neighborhoods in Fig. 1 for B = 9 and corresponding values of c_{B_1} and c_{B_2}. The optimization process results in picking the c_B of the square shape for the extraction function, as its corresponding neighborhood is more concentrated than the other. It can be simply shown that the optimum neighborhood in Fig. 1 ensures less MSE than the other. In another example, APPM finds c = 5 and a 5 × 5 square as the optimum neighborhood for B = 25 as shown in Fig. 2. It can be inferred from this figure that APPM gives less MSE distortion compared to the diamond encoding (DE) method [23] with k = 3 that applies the less concentrated shape represented by dashed lines in Fig. 2.

Without losing the generality of problem, $\Phi_B(x, y)$ can be formed around the origin. In this way, (x, y) is transferred to the (0,0) and its neighbors are shifted around it consequently.

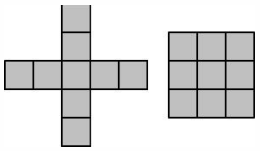

Fig. 1. Two different Φ(x, y) for B = 9 and two different c_9.

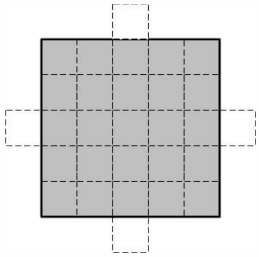

Fig. 2. DE (25) for k = 3 (dashed) and $\Phi_{25}(x, y)$ (solid).

x_i	y_i	$f(x_i, y_i) = (x_i + 3y_i) mod 9$
-1	-1	**5**
-1	1	**2**
0	-1	**6**
0	0	**0**
0	1	**3**
1	-1	**7**
-1	0	**8**
1	0	**1**
1	1	**4**

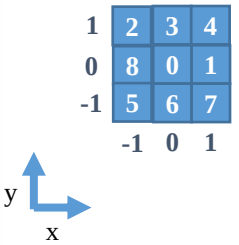

Fig. 3. Neighborhood table and $\Phi_B(x, y)$ for B = 9 in APPM.

Hereafter, $\Phi_B(0,0)$ means the neighborhood around the origin. Therefore, (3) can be restated as:

$$\text{Minimize}: \sum_{i=0}^{B-1}(x_i - 0)^2 + (y_i - 0)^2$$
$$\text{Subject to}: f(x_i, y_i) \in \{0, 1, \ldots, B-1\}$$
$$f(x_i, y_i) \neq f(x_j, y_j),\ if\ i \neq j$$
$$\forall\ 0 \leq i, j \leq B-1 \qquad (4)$$

Considering the M × N cover image and the bit-stream S, the embedding algorithm is explained in the following steps [28]:

1) Find the minimum B in $\lfloor M \times N/2 \rfloor \geq S_B$. S_B represents the S in B-ary notational system and |.| denotes cardinality.
2) Solve (4) to find the optimum $\Phi_B(x, y)$ and C_B.
3) For all values i from 0 to B − 1, record $(\hat{x}_i, \hat{y}_i)$ in the neighborhood defined by $\Phi_B(0,0)$ such that $f(\hat{x}_i, \hat{y}_i) = i$
4) Take a pixel pair (x, y) of the cover image to embed a message digit S_B. Find the modulus distance $d = (s_B - f(x, y))$ mod B between s_B and f(x, y), then replace (x, y) with $(x + \hat{x}_d, y + \hat{y}_d)$
5) Repeat step 4 to embed all the message digits.

Etraction process will be implemented simply by extracting B-ary digits $s_B = f(x', y')$ from all pixel pairs.

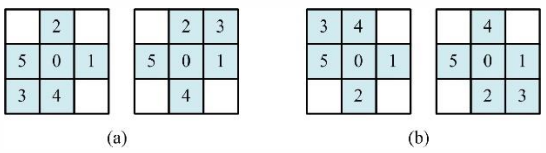

Fig. 4. Two possible $\Phi_6(x, y)$ for c = 2 (a), and c = 4 (b).

[DOI: 10.52547/itrc.13.4.43]

x_i	y_i	$f(x_i,y_i)$ $=(x_i+3y_i) mod 9$
0	0	**0**
-1	0	**5**
0	-1	**4**
-1 1	-1 1	**3**
0	1	**2**
1	0	**1**

x_i	y_i	$f(x_i,y_i)$ $=(x_i+3y_i) mod 9$
0	0	**0**
-1	0	**5**
0	-1	**2**
-1 1	-1 1	**3**
0	1	**4**
1	0	**1**

Fig. 5. Corresponding neighborhood tables for cases in Fig. 4

Since APPM works on the pixel pairs, the neighborhoods can be represented as two dimensional (2D) Cartesian shapes. A hypothetic similar algorithm working on the pixel triples results in $\Phi_B(x,y,z)$ which can be presented by 3D shapes. Extending the algorithm to the pixel groups larger than three, it would not be possible to visualize the neighborhoods. Therefore, we introduce a different approach to the APPM algorithm.

In this approach, the optimum neighborhood found around the origin is represented by a neighborhood table consisting of three columns and B rows rather than a shape. The first and second columns denote xi and yi values, and their corresponding $f(x_i,y_i)$ values are recorded in the last column. Both neighborhood shape and table for B = 9 and $c_9 = 3$ are sketched in Fig. 3. The table representation will be efficient for algorithms working on the pixel groups of size three or more.

III. Histogram Preserving Adaptive Pixel Pair Matching

The optimum c_B and $\Phi(x, y)$ solution is not unique. As an instance, both shapes in Fig. 4(a) are the solutions for optimization problem with B = 6 and c6 = 2. Similarly, B = 6 and c6 = 4 gives the solutions sketched in Fig. 4(b). All of these four shapes yield the similar MSE distortion. In other words, for each $f(x,y)$, there is a degree of freedom for choosing the optimum shape. This issue can be well observed in neighborhood table representations in Fig. 5. According to Fig. 5, whenever $d = (s_B - f(x,y); c_6 = 2)\ mod\ 6 = 3$ happens during the message embedding, both pixels can be alternatively increased or decreased by one upon our choice. Table I summarizes the minimum c_Bs and the number of their corresponding alternative $\Phi_B(x,y,z)$ for $2 \le B \le 64$.

This degree of freedom can be exploited for the sake of image histogram preservation. We define $C_{256\times1}$ change vector that records the number of alterations for 256 different pixel values during the embedding procedure. C(u) represents the number of changes in pixels with value u for $u = 0, \dots, 255$ due to embedding. C is set to 0 at the beginning. Consider the pixel pair (x,y) and distance $d = s_B - f(x,y)$ mod B. Assume that (x_d, y_d) is found using the neighborhood table. Two cases are possible for embedding:

1) $d = 0$, then (x,y) and consequently **C** remains intact.
2) $d \neq 0$, then $(x',y') = (x,y) + (x_d + y_d)$ and:
   - if $x' \neq x$, $\mathbf{C}(x) = \mathbf{C}(x) - 1$, $\mathbf{C}(x') = \mathbf{C}(x') + 1$.
   - if $y' \neq y$, $\mathbf{C}(y) = \mathbf{C}(y) - 1$, $\mathbf{C}(y') = \mathbf{C}(y') + 1$.

Above rules show how the number of pixels with intensity value u changes in C during embedding. Entries in C corresponding to the original and new pixel values are decreased and increased respectively. The histogram of the original image is preserved if for each certain pixel value i, the number of pixels with value i remain the same before and after embedding. This situation is equivalent to the C = 0 after embedding. Therefore, embedding with the lowest possible value of $D = \sum_{u=0}^{255} |C(u)|$ equals to the most preserved possible image histogram. For this sake, whenever there exists more than one choice for (x_d, y_d), the one is chosen that contributes more in decreasing the value of D. The detail of minimizing D is similar to [18]. Modifications on the original image are divided into two groups: mandatory changes where only one choice of (x_d, y_d) is available, and optional changes with more than one (x_d, y_d). Mandatory changes are first accomplished, and C is modified properly. When more than one (x_d, y_d) is available, the optimum is one that compensates the changes imposed by the mandatory phase.

The histogram preserving APPM (HP-APPM) algorithm for M × N image and input bit-stream S can be explained as below:

1) Find the minimum B in $\lfloor M \times N/2 \rfloor \ge |S_B|$. S_B represents the S in B-ary notational system and $|.|$ denotes cardinality.
2) Solve (4) to find the optimum $\Phi_B(x,y)$ and c_B.
3) For all values *i* from 0 to B-1, recoed $(\hat{x}_i, \hat{y}_i)$ from the neighborhood defined by $\Phi_B(0,0)$ in the neighborhood table such that $f(\hat{x}_i, \hat{y}_i) = i$. There might be more than one $(\hat{x}_i, \hat{y}_i)$ for a certain value of $i = f(\hat{x}_i, \hat{y}_i)$.
4) Take a pixel pair (x,y) of the cover image to embed a message digit s_B. Find the modules distance $d = s_B - f(x,y)$ mod B between s_B and $f(x,y)$.
5) Find (x_d, y_d) pair in neoghborhood table for which $f(x_d, y_d) = d$. If there is more than one (x_d, y_d) pair, take one which decreases $D = \sum_{u=0}^{255} |C(u)|$ as much as possible. Replace (x,y) with $(x',y') = (x,y) + (x_d, y_d)$ and update C correspondingly.
6) Repeat steps 4 and 5 until all the message digits are embedded.

TABLE I. Some c_B Values And The Number Of Their Corresponding $\Phi(X, Y)$ For $2 \leqslant B \leqslant 64$ In APPM.Table I

B	c_B	$\lvert\Phi_B\rvert$	B	c_B	$\lvert\Phi_B\rvert$	B	c_B	$\lvert\Phi_B\rvert$	B	c_B	$\lvert\Phi_B\rvert$	B	c_B	$\lvert\Phi_B\rvert$	B	c_B	$\lvert\Phi_B\rvert$
2	1	4	13	5	1	24	5	2	35	10	1	46	7	2	57	24	1
3	1	4	14	4	2	25	5	1	36	15	2	47	7	1	58	22	2
4	2	2	15	4	4	26	10	2	37	6	1	48	7	2	59	9	1
5	2	1	16	6	2	27	5	1	38	16	2	49	14	1	60	8	2
6	2	2	17	4	1	28	5	2	39	7	1	50	14	2	61	8	1
7	2	1	18	4	2	29	5	1	40	7	8	51	9	1	62	8	2
8	3	2	19	4	1	30	12	2	41	6	1	52	22	2	63	14	1
9	3	1	20	8	2	31	12	1	42	12	2	53	8	1	64	14	8
10	3	4	21	4	1	32	7	8	43	12	1	54	12	2			
11	3	1	22	5	2	33	6	1	44	8	2	55	21	16			
12	4	2	23	5	1	34	6	2	45	7	1	56	16	2			

[DOI: 10.52547/itrc.13.4.43]

When an overflow or underflow happened, i.e., $x' < 0, y' < 0, x' > 255$ or $y' > 255$ is replaced by (x'', y'') which is the solution of this optimization problem:

$$\text{Minimize} : (x - x'')^2 + (y - y'')^2$$
$$\text{Subject to} : f(x'', y'') = s_B,\ 0 \le x'', y'' \le 255 \quad (5)$$

In addition to the neighborhood tables for a certain c_B, there exists a degree of freedom in choosing c_B for a certain B. As a consequence, HP-APPM is implemented for the c_B with more degree of freedom. In cases such as Figs. 4(a) and 4(b) where both c_B result in the same degree of freedom, the lower value is chosen. Similar to the APPM, extraction function sB = $f(x'', y'')$ helps the receiver to restore the embedded bit-stream.

Both APPM and HP-APPM embedding methods are applied on Lena 512×512 gray-scale image for B = 10. The absolute differences between the histograms of the cover and stego images for both methods are demonstrated in Fig. 6. From this figure, it can be inferred that HP-APPM has managed to efficiently compensate the modification imposed to the cover image due to APPM data embedding.

IV. Adaptive Pixel Triple Matching

Exploiting the degree of freedom in equivalent 2D shapes, HP-APPM helps to preserve the original image histogram. However, this degree of freedom is limited for some B values. For instance, neighborhood set for B = 9 is a 3 × 3 square that offers no alternative option. Extending the size of pixel group to three increases the degree of freedom. This is the motivation behind the design of our proposed adaptive pixel triple matching (APTM) method.

The original image is divided into the groups of three pixels denoted by (x, y, z) in APTM method. The general extraction function follows the below form:

$$f(x, y, z) = (x + c_{B_1} \times y + c_{B_2} \times z) \bmod B \quad (6)$$

For the sake of embedding, the pixel triple (x, y, z) is modified to (x_0, y_0, z_0) where $f(x_0, y_0, z_0) = s_B$. c_{B_1} and c_{B_2} find their values within the range of 0 and B − 1. There is a neighborhood set ΦB(x, y, z) and a coefficient pair $C_B = (c_{B_1}, c_{B_2})$ for each value of B. MSE is defined as below:

TABLE II. Some c_B Values And Their Corresponding Φ(X, Y) For 2 ⩽ B ⩽ 64 In Aptm.Table Ii

B	c_{B_1}	c_{B_2}	B	c_{B_1}	c_{B_2}	B	c_{B_1}	c_{B_2}	B	c_{B_1}	c_{B_2}	B	c_{B_1}	c_{B_2}	B	c_{B_1}	c_{B_2}
3	1	1	14	2	5	25	3	8	36	6	9	47	4	18	58	4	16
4	1	2	15	2	5	26	3	9	37	3	14	48	4	18	59	5	17
5	1	2	16	2	6	27	3	9	38	6	9	49	4	14	60	22	26
6	2	3	17	2	6	28	3	9	39	12	18	50	8	12	61	4	17
7	2	3	18	2	6	29	3	9	40	4	14	51	4	15	62	5	18
8	2	3	19	2	6	30	3	11	41	5	13	52	18	22	63	5	25
9	2	3	20	2	7	31	3	11	42	4	16	53	4	21	64	7	18
10	2	3	21	3	8	32	4	10	43	3	17	54	4	20			
11	2	3	22	3	8	33	6	15	44	14	20	55	5	21			
12	2	4	23	3	8	34	3	13	45	4	17	56	5	18			
13	2	4	24	3	8	35	11	16	46	8	12	57	5	22			

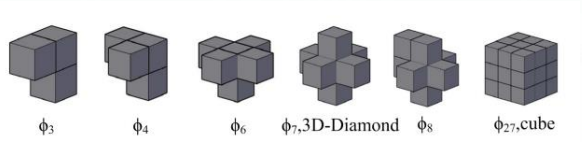

Fig. 7. Some of the APTM Optimum Embedding Shapes.

$$\text{MSE}_{\Phi(x,y,z)} = \frac{1}{3B}\sum_{i=0}^{B-1}\left((x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2\right)$$
$$\text{for all } (x_i, y_i, z_i) \in \Phi(x, y, z) \quad (7)$$

$\Phi_B(x, y, z)$ and $C_B = (c_{B_1}, c_{B_2})$ are found by solving the optimization problem below:

$$\text{Minimize} : \sum_{i=0}^{B-1}(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2$$
$$\text{Subject to} : f(x_i, y_i, z_i) \in \{0, 1, \ldots, B-1\}$$
$$f(x_i, y_i, z_i) \neq f(x_j, y_j, z_j),\ if\ i \neq j$$
$$\forall\ 0 \le i, j \le B - 1 \quad (8)$$

Similar to APPM, neighborhoods are found around the origin. Some optimum (c_{B_1}, c_{B_2}) pairs for 3 ≤ B ≤ 64 are shown in Table II. Some of the optimum 3D neighborhood shapes are sketched in Fig. 7. Having the optimum neighborhood and c_B found, the embedding and extraction processes will be implemented similar to the Section II.

The theoretical MSE error of DE, APPM and APTM under various payloads are compared in Table III. Note that DE is applicable only for some specific notational systems corresponding to DE parameter k ranging from 1 to 10. Equivalent APPM and APTM algorithms for these specific payloads are then implemented and compared. Aside from the case k = 1, it is inferred from Table III that APTM exhibits a better performance comparing to the DE and APTM in terms of MSE.

TABLE III. Mse Comparison Of De, Appm And Aptm Methods For Practical De Payloads

bpp	DE		APPM		APTM		MSE Improvement	
	k	MSE	c_B	MSE	(c_{B_1}, c_{B_2})	MSE	to DE	to APPM
1.161	1	0.4	2	0.4	(2, 3)	0.424	-0.024	-0.024
1.850	2	1.077	5	1.077	(4, 18)	1.049	0.028	0.028
2.322	3	2.080	5	2.000	(10, 50)	1.957	0.123	0.043
2.679	4	3.415	6	3.341	(7, 46)	3.237	0.178	0.104
2.965	5	5.082	8	4.902	(24, 156)	4.797	0.285	0.105
3.205	6	7.082	10	6.847	(184,295)	6.687	0.395	0.160
3.410	7	9.416	31	9.071	(30,290)	8.883	0.533	0.188
3.590	8	12.083	22	11.890	(13,729)	11.386	0.697	0.504
3.750	9	15.083	39	14.519	(15,745)	14.219	0.864	0.300
3.894	10	18.416	26	17.787	(185,1010)	17.364	1.052	0.423

[DOI: 10.52547/itrc.13.4.43]

TABLE IV. MSE COMPARISON OF APPM AND APTM METHODS FOR PRACTICAL APPM PAYLOADS

bpp	APPM			APTM		
	B	c_B	MSE	B	(c_{B_1}, c_{B_2})	MSE
1	4	2	0.375	8	(2, 3)	0.333
1.292	6	2	0.5	15	(2, 5)	0.488
1.5	8	3	0.625	23	(3, 8)	0.608
1.729	11	3	0.909	36	(6, 9)	0.888
2	16	6	1.343	64	(7, 18)	1.265
2.229	22	5	1.795	103	(5, 40)	1.734
2.5	32	7	2.625	181	(10, 35)	2.530
2.745	45	7	3.644	302	(74, 89)	3.539
3	64	14	5.203	512	(22, 230)	5.027
3.250	91	27	7.318	867	(50,259)	7.114
3.500	128	12	10.308	1448	(51,139)	10.065
3.750	181	39	14.519	2435	(15,745)	14.219
4	256	60	20.57	4096	(390, 667)	20.114

APPM and APTM algorithms for these specific payloads are then implemented and compared. Aside from the case k = 1, it is inferred from Table III that APTM exhibits a better performance comparing to the DE and APTM in terms of MSE.

Although APPM is not as limited as DE in practical bpp payloads, yet it can input only specific bpp values with considerable gap among them. Table IV compares the MSE performance of APPM and APTM for some practical APPM payloads and their equivalent APTM systems. However, it can be shown in Section V that APTM is much more flexible in terms of input payload comparing to APPM.

In order to experimentally investigate this superiority, APPM and APTM are implemented on six famous images. 512×512 gray-scale images Lena, Jet, Boat, Elaine, Couple and Baboon are used for data embedding under the payloads of 426000, 757000 and 1048000 bits. Message bits are randomly generated and embedded via both methods. The MSE between the cover and stego images of this test are reported in Table V. It can be perceived from this Table that MSE improvements up to 0.5 in the APPM method are achievable by implementing the APTM embedding technique. Like the 2D case, the optimization problem in (8) does not have a unique solution. For example, there are 24 coefficient pairs for B = 9. Moreover, for a certain coefficient pair such as (c_{B_1}, c_{B_2}) = (2, 3), there will be four neighborhood set shown in Fig. 8. Therefore, there is more degree of freedom comparing to the APPM optimization for B = 9.

x_i	y_i	z_i	$f(x_i,y_i,z_i) = (x_i + 2y_i + 3z_i) \bmod 9$
0	0	0	**0**
-1	0	0	**8**
0	-1	0	**7**
0	0	-1	**6**
-1	0	-1	**5**
0	1	1	
0	-1	-1	**4**
1	0	1	
0	0	1	**3**
0	1	0	**2**
1	0	0	**1**

Fig. 8. Neighborhood table for B = 9 in APTM.

Again, this degree of freedom will be exploited for the sake of histogram preserving. The change vector C is defined and modified similar to Section III. When there is more than one option to change a pixel triple, the one is chosen that helps in decreasing the value of D. Our histogram preserving algorithm which works on pixel triples is called histogram preserving APTM (HP-APTM). Extending the concepts from 2D to 3D, the embedding and extraction processes are very similar to those in Section III.

Histogram preserving improvement of HP-APTM to APTM is theoretically evident. APPM, HP-APPM and HP-APTM embedding methods are applied on Lena 512×512 gray-scale image. HP-APTM is implemented for B = 42, while the other two are implemented for B = 12 that yields the equivalent bpp payload. The absolute differences between the histograms of the cover and stego images for all methods are demonstrated in Fig. 9. From this figure it can be perceived that both histogram preserving techniques outperforms APPM. However, histogram preserving can be performed more successfully in HP-APTM comparing to HP-APPM, based on Fig. 9.

The histogram improvement achieved by HP-APTM and HP-APPM to APPM can be well presented by KullbackLeibler (KL) divergence, as a metric for histogram similarity.

Consider G = {0, 1, . . . 255} as the set of possible pixel intensity values. The KL divergence of the cover and stego images is calculated as:

$$D(c\|s) = \sum_{i \in G} c(i) \log_2 \left(\frac{c(i)}{s(i)} \right) \qquad (9)$$

Where c(i) and s(i) represent the frequency of the pixel value i in the cover and stego images respectively. The less value of the KL divergence means the more similar histograms for the cover and stego images and thus; the more secure algorithm. The KL divergence is compared for APPM, HP-APPM and HP-APTM data embedded 512×512 gray-scale images, and the payload of 470000 bits (1.79 bpp) in Table VI. The results confirm that HP-APTM has been the most successful in histogram preservation in terms of KL divergence. The Lena cover image and stego image due to APPM, HP-APPM and HP-APTM data embedding. B = 42 in HP-APTM and B = 12 for the other two. Images generated in this experiment are shown in Fig. 10, in which no visible trace of data embedding is perceivable.

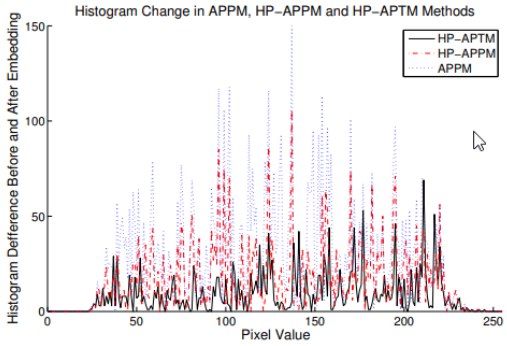

Fig. 9. Differences in the cover and stego image histograms for Lena 512×512 grey scale

[DOI: 10.52547/itrc.13.4.43]

V. Adaptive Pixel Group Matching

Based on the concepts used to extend APPM to APTM, adaptive pixel group matching is proposed that works on groups of k pixels. This algorithm is called adaptive pixel group matching (APGM). More degree of freedom and thus, better preserved histogram is expected with this approach. Moreover, neighborhood table is still applicable for k ≥ 4 where the neighborhood shapes are not feasible anymore.

The original image is divided into the groups of k pixels in APGM. $P = [p_1, p_2, \dots, p_k]^T$ represents a group of k pixels. The extraction function follows the general form of $f(p_1, p_2, \dots, p_k) = p_1 + c_{B_1} p_2 + \cdots + c_{B_{k-1}} p_k$ mod B. The pixel group P is modified to P0 that satisfies $f(P') = s_B$. P' denotes the pixel group after data embedding. $c_{B_1}, \dots, c_{B_{k-1}}$ find their values within the range of 0 to B − 1. In this case, MSE for a certain B, neighborhood set $\Phi_B(P)$, and coefficient vector $C_B = (c_{B_1}, c_{B_2}, \dots, c_{B_{k-1}})$ is defined as:

$$MSE_{\Phi(\mathbf{P})} = \frac{1}{kB} \times \|\mathbf{P} - \mathbf{P}_n\|^2, \ \mathbf{P}_n \in \Phi(\mathbf{P}) \qquad (10)$$

TABLE VI. KL Divergence Comparison Of APPM, HP-APPM And HP-APTM For The Payload Of 470000 Bits

Image	APPM $B = 12, c = 4$	HP-APPM $B = 12, c = 4$	HP-APTM $B = 42, (c_1, c_2) = (5, 13)$
Baboon	0.0009	0.0005	0.0002
Boat	0.0102	0.0090	0.0064
Couple	0.3357	0.3141	0.2597
Elaine	0.0039	0.0024	0.0006
Jet	0.0024	0.0016	0.0011
Lena	0.0009	0.0004	0.0002

Fig. 10. Lena 512×512 gray-scale cover image (a) and its corresponding stego images generated by APPM, HP-APPM and HP-APTM at the rate of 1.79 bpp (b)-(d).

Where ‖.‖ denotes the Euclidean norm. Therefore, ΦB(P) and C_B will be found by solving the optimization problem below:

$$\begin{aligned} &\text{Minimize} : \|\mathbf{P}\|^2 \\ &\text{Subject to} : f(\mathbf{P}) \in \{0, 1, \dots, B-1\} \\ &f(\mathbf{P}_i) \neq f(\mathbf{P}_j) \\ &\forall\, 0 \leq i, j \leq B-1 \end{aligned} \qquad (11)$$

B(P) is defined around the origin. (11) does not necessarily yield a unique solution. Therefore, there might be a set of optimum C_B vectors, with a set of optimum neighborhoods for each one of them. As a result, a degree of freedom is offered by APGM algorithm that can be exploited to propose histogram preserving APGM (HP-APGM). Among all possible modifications for a pixel group in HP-APGM, the one is picked that helps to decrease $D = \sum_{u=0}^{255} |C(u)|$. Therefore, the APGM embedding method can be summarized as below:

1) Find the minimum B in $\lfloor M \times N/2 \rfloor \geq |S_B|$. S_B represents the S in B-ary notational system and $|.|$ denotes cardinality.
2) Solve (11) to find the optimum $\Phi_B(P)$ and c_B.
3) For all values i from 0 to B-1, recoed $\hat{P}_i$ from the neighborhood defined by $\Phi_B(0)$ in the neighborhood table such that $f(\hat{P}_i) = i$. There might be more than one $\hat{P}_i$ for a certain value of $i = f(\hat{P}_i)$.
4) Take a pixel group $P = [p_1, \dots, p_k]^T$ of the cover image to embed a message digit s_B. Find the modules distance $d = s_B - f(P)$ mod B between s_B and $f(P)$.
5) Find P_d pair in neoghborhood table for which $f(P_d) = d$. If there is more than one P_d pair, take one which decreases $D = \sum_{u=0}^{255} |C(u)|$ as much as possible. Replace P with $P' = P + P_d$ and update C correspondingly.
6) Repeat steps 4 and 5 until all the message digits are embedded.

In case of overflow or underflow during the embedding process, Pˆ 0 is found such that f(Pˆ 0) = sB and MSE is minimized. In other words, the below optimization is solved:

$$\begin{aligned} &\text{Minimize} : \|\mathbf{P} - \hat{\mathbf{P}}'\|^2 \\ &\text{Subject to} : f(\hat{\mathbf{P}}') = s_B, \ \hat{\mathbf{P}}' = [\hat{p}'_i], 0 \leq \hat{p}'_i \leq 255 \end{aligned} \qquad (12)$$

Similar to the HP-APTM, the C_B that offers more degree of freedom is picked among the optimized C_B vectors found for a certain B. When there are more than such optimum vectors with the most degree of freedom, the vector with the lower norm is chosen. The extraction process is simply implemented by applying the extraction function $f(p')$ to the pixel groups of the stego image.

[DOI: 10.52547/itrc.13.4.43]

TABLE V
MSE COMPARISON OF APPM AND APTM METHODS FOR VARIOUS PAYLOADS AND IMAGES

Image	Payload=426000 bits, 1.625bpp		Payload=757000 bits, 2.887bpp		Payload=1048000 bits, 4bpp	
	APPM	APTM	APPM	APTM	APPM	APTM
	$B=10,c=3$	$B=29,c_1=3,c_2=9$	$B=55,c=21$	$B=405,c_1=31,c_2=54$	$B=256,c=60$	$B=4096,c_1=390,c_2=667$
Baboon	0.8505	0.7341	4.5182	4.2973	20.5750	20.0843
Boat	0.8502	0.7339	4.5053	4.3023	20.5796	20.0868
Couple	0.8474	0.7341	4.5064	4.3108	20.6149	20.1373
Elaine	0.8488	0.7357	4.5058	4.3006	20.5792	20.1426
Jet	0.8505	0.7344	4.5082	4.2967	20.5654	20.1595
Lena	0.8477	0.7357	4.5031	4.3000	20.5904	20.0937

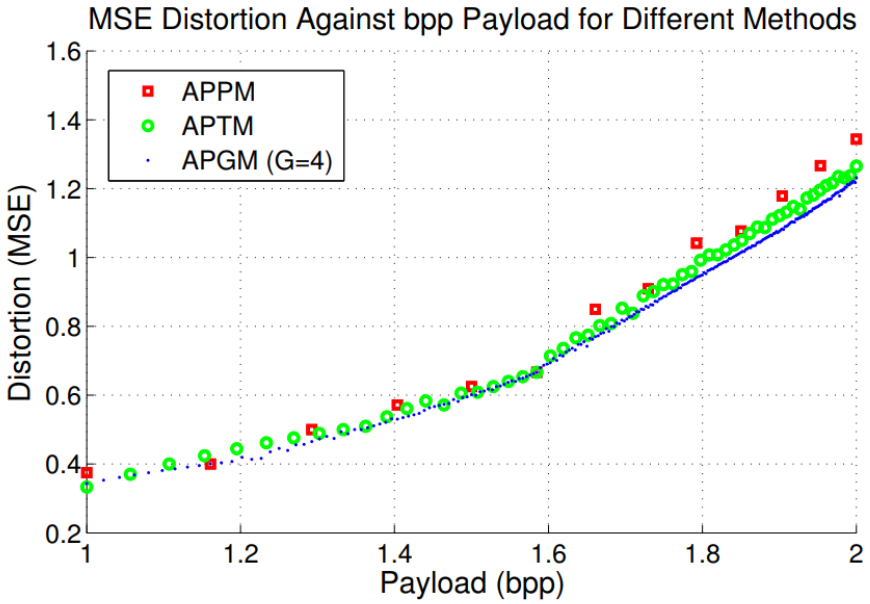

Fig. 11. MSE comparison of different methods. Adaptive pixel group matching is implemented for groups of size four pixels.

The theoretical results for an MSE comparison among different methods are given in Fig. 11. From this figure it can be observed that the embedding distortion decreases by extending the pixel groups. APTM improves the MSE performance of APPM by about 0.1 for higher payloads, while its own performance is boosted by APGM working on groups of four pixels. Another important conclusion about the flexibility of the algorithms can be drawn from this figure. While the APPM is applicable only for limited number of bpp payloads shown by squares, much more flexibility for input payload is achievable by APTM presented in circles. However, the payload flexibility is significantly improved for APGM technique where almost all bpp payloads are available, that makes the dots in Fig. 11 to look like a continuous line.

VI. SECURITY ANALYSIS

A security analysis of the proposed method is performed using SPAM [8] and SRM [29] stegnlyzers. The results are compared for APPM, HP-APPM and HP-APTM techniques. Six thousand 512×512 gray-scale images from BOWS2 image database [30] are used as the training set. All algorithms are implemented for 0.1, 0.2 and 0.4 bpp payloads. For each pair of applied method and payload, three different B values are used for implementation. Since the SPAM is originally designed to analyze the ±1 LSB techniques, B values above 9 in APPM and 27 in APTM that result in more than ±1 modifications are avoided. The second order SPAM feature set that is based on second order markov features is applied in this analysis [8]. The other steganalyzer applied in our experiments is the recently proposed and efficient SRM stegnalyzer that is based on a rich model including numerous diverse sub models [29].

The training set consists of 6000 cover images and 6000 stego images divided uniformly among nine different embedding profiles. Thereafter, in order to investigate the performance of the trained steganalyzer, 1000 cover and 1000 stego images other than the training images are used for each embedding profile. The ensemble classifier [9] is used to separate the cover and stego images according to the SPAM and SRM features. The results of the SPAM security analysis are summarized in Table VII. Pe = 1 2 (PF P +PF N) represents the error performance of the steganalyzer, where PF P and PF N stand for the probability of false positive and false negative respectively. From this Table, it can be observed that SPAM detection is deteriorated about two percent for both proposed HP-APPM and HP-APTM methods compared to the rival APPM technique. This observation confirms the security superiority of the proposed method to APPM.

This superior performance is confirmed through SRM steganalysis too. Similar results for the SRM are reported in Table VIII. Note that the performance of the APPM and the proposed method are almost the same at 0.4 bpp payload due to the sophisticated and efficient design of the rich model. Normally, data

[DOI: 10.52547/itrc.13.4.43]

embedding at such high r ates is avoided due to the high probability of the detection. From Table VIII it can be seen that the differences between the performance of the APPM an d the proposed method are meaningful for the lower embedding rates.

TABLE VII. SPAM ERROR RATE FOR DIFFERENT METHODS AND PAYLOADS

Payload (bpp)	APPM		HP-APPM		HP-APTM	
	B	P_e	B	P_e	B	P_e
0.1	4	0.2255	4	0.2820	11	0.3690
	6	0.2195	6	0.2815	13	0.3970
	8	0.2105	8	0.3635	15	0.3910
0.2	4	0.1450	4	0.1755	11	0.2275
	6	0.1385	6	0.1720	13	0.2215
	8	0.1410	8	0.1665	15	0.1985
0.4	4	0.1240	4	0.1480	11	0.1650
	6	0.1155	6	0.1420	13	0.1605
	8	0.1225	8	0.1460	15	0.1675

TABLE VIII. SRM ERROR RATE FOR DIFFERENT METHODS AND PAYLOADS

Payload (bpp)	APPM		HP-APPM		HP-APTM	
	B	P_e	B	P_e	B	P_e
0.1	4	0.1865	4	0.2235	11	0.2570
	6	0.1705	6	0.2075	13	0.2430
	8	0.1590	8	0.1860	15	0.2455
0.2	4	0.0845	4	0.1125	11	0.1315
	6	0.0820	6	0.1110	13	0.1095
	8	0.0725	8	0.1020	15	0.1110
0.4	4	0.0365	4	0.0515	11	0.0550
	6	0.0335	6	0.0460	13	0.0475
	8	0.0350	8	0.0395	15	0.0425

VII. CONCLUSION

In this paper, adaptive pixel group matching method was proposed for data embedding. In this method, a group of pixels is optimally modified such that its extraction function matches the embedding message. Therefore, the best embedding strategies are found for different values of embedding rates and pixel group sizes. Whenever there exists more than one optimal solution, a combination of them is applied for the sake of histogram preservation. The theoretical and experimental results confirm that extending the size of pixel groups helps to achieve less value for the imposed distortion. On the other hand, more equivalent optimum embedding strategies are achieved as a result of this extension which proper selection among them allows us to keep the histogram of the original image as intact as possible.

REFERENCES

[1] M. A. Akhaee and F. Marvasti, "A survey on digital data hiding schemes: Principals, algorithms, and applications." ISeCure, vol. 5, no. 1, 2013.

[2] R. Toosi, M. Sadeghi, and M. A. Akhaee, "Robust image watermarking using sample area quantization," Multimedia Tools and Applications, vol. 78, no. 24, pp. 34 963–34 980, 2019.

[3] A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," in Proc. 3rd Int. Workshop on Information Hiding, vol. 1768, 1999, pp. 61–76.

[4] J. Harmsen and W. A. Pearlman, "Steganalysis of additive-noise modelable information hiding," in Society of Photo-Optical Instrumentation Engineers (SPIE) Conf., vol. 5020, Jun. 2003, pp. 131–142.

[5] A. D. Ker, "Steganalysis of LSB matching in grayscale images," IEEE Signal Process. Lett., vol. 12, no. 6, pp. 441 – 444, Jun. 2005.

[6] Y. Q. Shi, C. Chen, and W. Chen, "A markov process based approach to effective attacking JPEG steganography," in Information Hiding. Springer, 2007, pp. 249–264.

[7] J. Kodovsky and J. Fridrich, "Steganalysis in high dimensions: Fusing ` classifiers built on random subspaces," in IS&T/SPIE Electronic Imaging. International Society for Optics and Photonics, 2011, pp. 78 800L– 78 800L.

[8] Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," Information Forensics and Security, IEEE Transactions on, vol. 5, no. 2, pp. 215–224, 2010.

[9] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," Information Forensics and Security, IEEE Transactions on, vol. 7, no. 2, pp. 432–444, 2012.

[10] R. Toosi, S. Salehkalaibar, and M. A. Akhaee, "Improved ensemble growing method for steganalysis of digital media," Multimedia Tools and Applications, vol. 78, no. 8, pp. 9877–9893, 2019.

[11] M. Hamghalam, S. Mirzakuchaki, and M. A. Akhaee, "Robust image watermarking using dihedral angle based on maximum-likelihood detector," IET Image Processing, vol. 7, no. 5, pp. 451–463, 2013.

[12] M. Sadeghi, R. Toosi, and M. A. Akhaee, "Blind gain invariant image watermarking using random projection approach," Signal Processing, vol. 163, pp. 213–224, 2019.

[13] J. Mielikainen, "LSB matching revisited," Signal Processing Letters, IEEE, vol. 13, no. 5, pp. 285–287, 2006.

[14] . Sarreshtedari, M. Ghotbi, and S. Ghaemmaghami, "One-third probability embedding: Less detectable LSB steganography," in Multimedia and Expo, ICME 2009. IEEE International Conference on, 2009, pp. 1002–1005.

[15] X. Li, B. Yang, D. Cheng, and T. Zeng, "A generalization of LSB matching," Signal Processing Letters, IEEE, vol. 16, no. 2, pp. 69 – 72, feb. 2009.

[16] M. A. Akhaee, S. M. E. Sahraeian, and F. Marvasti, "Contourlet-based image watermarking using optimum detector in a noisy environment," IEEE Transactions on Image Processing, vol. 19, no. 4, pp. 967–980, 2009.

[17] H. M. Sun, K. H. Wang, C. C. Liang, and Y. S. Kao, "A LSB substitution compatible steganography," in TENCON 2007 - IEEE Region 10 Conf., Nov. 2007, pp. 1 –3.

[18] S. Sarreshtedari and M. Akhaee, "One-third probability embedding: a new ± one histogram compensating image least significant bit steganography scheme," Image Processing, IET, vol. 8, no. 2, pp. 78–89, February 2014.

[19] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," Pattern Recognition Letters, vol. 24, no. 9–10, pp. 1613 – 1626, 2003.

[20] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited," Info. Forensics and Security, IEEE Trans. on, vol. 5, no. 2, pp. 201 –214, Jun. 2010.

[21] H. M. Sun, C. Y. Weng, C. F. Lee, and C. H. Yang, "Anti-forensics with steganographic data embedding in digital images," Selected Areas in Communications, IEEE Journal on, vol. 29, no. 7, pp. 1392 –1403, Aug. 2011.

[22] . Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," Communications Lett., IEEE, vol. 10, no. 11, pp. 781 –783, Nov. 2006.

[23] R. M. Chao, H. C. Wu, C. C. Lee, and Y. P. Chu, "A novel image data hiding scheme with diamond encoding," EURASIP J. Inf. Security, vol. 2009, 2009.

[24] H.-S. Leng, J.-F. Lee, and H.-W. Tseng, "A high payload emd-based steganographic method using two extraction functions," Digital Signal Processing, vol. 113, p. 103026, 2021.

[DOI: 10.52547/itrc.13.4.43]

[25] S. Saha, A. Chakraborty, A. Chatterjee, S. Dhargupta, S. K. Ghosal, and R. Sarkar, "Extended exploiting modification direction based steganography using hashed-weightage array," Multimedia Tools and Applications, vol. 79, no. 29, pp. 20 973–20 993, 2020.

[26] Q. Ke, Q. Liao, and R. Pan, "An improved emd parallel steganography algorithm," in Journal of Physics: Conference Series, vol. 1621, no. 1. IOP Publishing, 2020, p. 012006.

[27] R. Atta, M. Ghanbari, and I. Elnahry, "Advanced image steganography based on exploiting modification direction and neutrosophic set," Multimedia Tools and Applications, pp. 1–19, 2021.

[28] W. Hong and T. S. Chen, "A novel data embedding method using adaptive pixel pair matching," Info. Forensics and Security, IEEE Trans. on, vol. 7, no. 1, pp. 176 –184, Feb. 2012.

[29] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," Information Forensics and Security, IEEE Transactions on, vol. 7, no. 3, pp. 868–882, June 2012.

[30] The Dataset from the 2nd Bows Contest. (2012, Mar. 26) [Online]. Available: http://bows2.ec-lille.fr/.

Alireza Shahanaghi After receiving the B.Sc. and M.Sc. degrees in Electrical and Computer Engineering from the University of Tehran, Iran in 2013 and 2016, Alireza started to work as a research assistant in the Array Processing Lab at the University of Tehran. In his new position, Alireza was the coordinator of the MIMO research team, focusing on the development of Wireless Communication Systems using the Zynq-7000 FPGA Board. In 2017, he joined Virginia Tech to pursue his Ph.D. degree in electrical and Computer Engineering. Now he is a member of the Center for Embedded Systems for Critical Applications (CESCA) and Wireless at Virginia Tech research groups, engaged in the research on the Modeling and Simulation of Wireless Links in Marine Environments using Signal-Processing, Machine-Learning, Stochastic Analysis, and Optimization Techniques.

Mohammad Ali Akhaee received the B.Sc. degree in Electronics and Communications Engineering from the Amirkabir University of Technology, Tehran, Iran, and the M.Sc. and Ph.D. degrees from the Sharif University of Technology, Tehran, Iran, in 2005 and 2009, respectively. He is currently an Assistant Professor with the College of Engineering and the Director of the Secure Communication Laboratory, University of Tehran, Tehran, Iran. He has authored or coauthored more than 60 papers, and holds one Iranian patent. His research interests include the area of Signal Processing, in Particular Multimedia Security, Data Hiding, and Machine Learning. Prof. Akhaee was the Technical Program Chair of EUSIPCO '11 and the Executive Chair of ISCISC '14 and Financial Chair of RTEST'18. He received the Governmental Endeavour Research Fellowship from Australia in 2010 and the Governmental award from Ministry of Information and Communication Technology (ITC) from Iran in 2017.

Saeed Sarreshtedari was born in Iran in 1985. He received the B.Sc. degree in electrical engineering, in 2007, and the M.Sc. degree in Communications Security from the Sharif University of Technology, Tehran, Iran, in 2009. He is currently pursuing the Ph.D. degree in Communications Systems with the University of Tehran, Iran. His research interests include Multimedia Security, Information Hiding, Watermarking, Image, Speech and Video processing, Radar Systems, and Array Processing.

Ramin Toosi received the B.Sc. degree in Electrical Engineering (Communication Devision) from Shahid Beheshti University, Tehran, Iran in 2014. He received the M.Sc. degree in Electrical Engineering (System Division) from University of Tehran in 2016. He is currently pursuing the Ph.D. degree in Communications Systems at the Secure Communication Lab, University of Tehran. His fields of interest include Signal Processing, Multimedia Security, Pattern Recognition, and Information Theory.

[DOI: 10.52547/itrc.13.4.43]