Enhancing the Capability of Data Hiding Method Based on Reduced Difference Expansion

Pascal Maniriho *Member, IAENG*, Tohari Ahmad, *Member, IAENG*

***Abstract*— The last few decades have been marked by a rapid growth and significant enhancement of data hiding techniques. Data hiding is performed by utilizing some carriers to conceal private information which is further extracted later to verify the genuineness. Digital steganography has been recognized among the recent and most popular data hiding techniques. It is a technique of concealing the existence of exchanging information between individuals or communication parties which is performed by embedding confidential information into multimedia carriers such as audio, text, image and video. It originates from the concept that if the communication is visible, the suspicion or attack is obvious. Hence, the aim is to always disguise the presence of the hidden secret data. Recently, digital image steganography has been applied in various applications. However, achieving good quality of the stego image and a reasonable embedding capacity is still a severe challenge. Thereby, in this paper we suggest a digital image steganographic technique which is developed based on pixel block, reduced difference expansion (RDE) and constant base point which is intended to enhance the quality of the stego image while achieving a good embedding capacity. According to the experimental results, both quality and capacity which are respectively evaluated by measuring the peak signal-to-noise ratio (PSNR) and number of hidden secret bits are well preserved. That is, our method outdoes the previous ones.**

***Index Terms*— Data hiding, data protection, information security, reduced difference expansion, secret data**

I. Introduction

IN the field of Information and Communication Technology (ICT), the security of information has become a matter of utmost concern due to the evolution of the internet. The illegitimate users can easily intercept and alter sensitive information while being transmitted to the intended recipient via the internet. Thus, this problem has brought the need for securing sensitive information which can be easily available for intruders intending to violate user rights. Recently, many data hiding research dealing with protecting information being shared between individuals via the internet channel have been already carried out. Data hiding aims at protecting privacy, intellectual property rights and content authentication by concealing the sensitive information (also called secret message) into multimedia objects. That is, it does play a significant role in multimedia security [1]. One of the best data hiding techniques to enforce the information security is steganography.

The concept of steganography differs from cryptography in way that steganography is all about keeping the existence of information itself unknown whereas the purpose of cryptography is keeping the contents of information secret, that is, it does provide the security with respect to the contents of the message by transforming it into another unreadable form.

In steganography based on digital image, the information to be kept confidential is called secret message and the cover in which the secret message is embedded is called the cover image. Thus, steganography serves to conceal the presence of the secret message. The image obtained after embedding the secret message is called stego image. Furthermore, the secret message is concealed in the pixels of the original cover image without much distorting it. In reversible data hiding schemes, the cover media can be completely retrieved from the stego media after extracting the original message [2] [3]. Spatial domain and transform domain are two main approaches that are used in digital image steganography. In spatial domain, there is no transformation done before hiding the secret message in the cover image. That is, the secret message is directly hidden in the pixel values of the image. Several approaches that conceal secret data by employing spatial domain approach have been implemented [4] [5] [6] [7]. Different from the spatial domain, however, in the transform domain approach before embedding the secret message the image is first transformed from spatial to frequency domain by utilizing some of the transform schemes such as discrete wavelet transform (DWT), discrete cosine transform (DCT), double density dual tree (DD DT), Hadamard transform, curvelet transform, etc. After the transformation, the secret

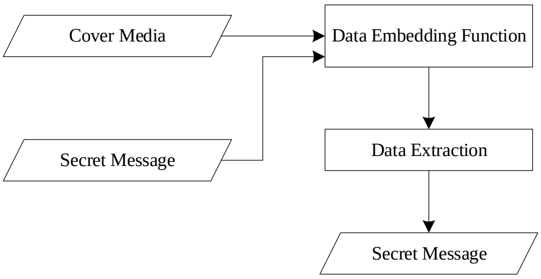


Fig. 1 Architecture of steganography in spatial domain

Manuscript received July 19, 2017; revised November 23, 2017. Part of this works was supported by the Ministry of Research, Technology and Higher Education Research Project No 010/SP2H/LT/DRPM/IV/2017.

Pascal Maniriho is with Department of Informatics, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia (email: pascal15@mhs.if.its.ac.id).

Tohari Ahmad is with Department of Informatics, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia; (corresponding author, phone: +62315939214; email: tohari@if.its.ac.id).

(Advance online publication: 10 February 2018)

message is then embedded in the transform coefficients. That is, the information is concealed in the regions of the image that are less exposed to image processing operation such as compression or cropping and this demonstrates its advantage over the spatial domain [8] [9]. However, the capacity of the secret message can be a challenge.

The main goal of any digital image stenographic approach is to simultaneously enhance security, the visual quality of the stego image, and embedding capacity [10]. The embedding capacity was increased by hiding data into smooth blocks having different sizes [11]. Data was concealed in digital image by utilizing mean and additive modulus [12]. The robustness and security of data were achieved by the model developed in the frequency domain [13]. Tanwar and Bisla [14] implemented an approach that conceals secret data in an audio cover. The audio was first processed thereafter the data were hidden in least significant bit (LSB). In our previous method, medical data were hidden in audio files using the data hiding scheme [15]. To conceal data into the carrier image, histogram modification was performed [16]. Fig. 1 illustrates the architecture of steganography in the spatial domain approach.

In the existing methods, however, the number of secret bits which can be embedded in the cover and its respective visual quality of the stego image are still the problems. In this paper, we present a data hiding technique to deal with those problems by using digital grayscale images. The proposed method is based on the spatial domain approach and it is able to hide 3 bits in one pixel block of the image in accordance with the predefined criteria which control the embedding process. The image is first divided into non-overlapping blocks of size 2 × 2. That is, four pixels are defined in each block. Moreover, with this proposed method the embedding capacity and the visual quality can be controlled so as to achieve high payload capacity and good PSNR.

The rest of this paper is organized as follows: Background and related work in the literature are described in section 2, the proposed method is discussed in section 3, the experimental results and discussion are given in section 4. Finally, the paper is wrapped up with conclusion and the suggested future work.

II. Background and Related Works

According to the previous research carried out on digital image steganography, high embedding capacity and good visual quality of the stego image can be achieved by using reduced difference expansion [2]. An enhanced multi-layer data hiding method based on IRDE was implemented in [17]. Their scheme was built by combining two approaches proposed in [2] [18]. The IRDE was applied in all layers to control the embedding capacity and the quality as well. Besides, their results show that the quality and capacity were improved. Both visual quality of the stego image and the payload capacity were enhanced by utilizing modulo function and four-pixel differencing [19]. Moreover, the difference between pixels was calculated by first defining four neighboring pixels in each block, after that the data were hidden based on the obtained difference.

The data hiding approach developed based on pixel value differencing and FFEMD was further suggested [20]. Data were concealed into two layers of RGB colored images [21]. In 2015, Swain and Lenka [22] proposed a scheme that utilizes correlation calculated between neighboring pixels to conceal data into the digital image. The existing methods based on pixel value ordering were enhanced by a model proposed in [23] which utilizes a dynamic blocking technique to adaptively partition the image into blocks of various sizes. Two important areas (“flat and rough areas”) were considered within the image. To achieve high payload capacity, small blocks were generated from flat areas and large blocks were constructed from the rough areas in order to prevent PSNR from being decreased. Their experimental results show that making the block size dynamic has advantage over blocks with fixed size since it can simultaneously improve the embedding capacity while achieving a low degradation of the stego image.

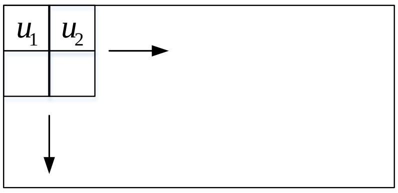


Fig. 2. Neighboring pixels in a grayscale image

To improve the quality and embedding capacity, the scheme in [24] was built based on reduced difference expansion and quad. Secret data were embedded in the difference and sum computed by considering adjacent pixel pairs in non-overlapping blocks [25]. Besides, both secret message and the cover image were fed to the algorithm as inputs. Since their algorithm makes use of two parameters to conceal data, some options have been defined before embedding data. For example, if the data cannot be embedded in the difference, they are then embedded into the sum. However, if it is possible to embed data in the sum or difference, the option that achieves more embedding bits is chosen. In addition, if both methods are able to achieve high embedding capacity, the method that makes less changes in the value of the pixel pair is utilized for embedding.

The LSB and 8nPVD were utilized to implement the method presented in [26]. This method divides a grayscale image into blocks of size 3 × 3 which are non- overlapped. Different from other schemes, blocks were constructed in row major order. To obtain the number of secret bits that can be hidden in the LSB of the difference values, the gray level was further split up into different ranges thereafter the secret bits to be concealed were computed using the generated gray level range table. Additionally, the research done in [27] provides a mechanism that utilizes difference expansion (DE) to embed bits of the secret message with low complexity. This method is based on computing the average and difference between two neighboring pixels. Considering a grayscale digital image depicted in Fig. 2, let u_1 and u_2 be two neighboring pixels, while m and v represent their average and difference correspondingly as shown in (1).

$$m = \left\lfloor \frac{u_1+u_2}{2} \right\rfloor \quad \text{and} \quad v = u_1 - u_2 \qquad (1)$$

To perform the embedding, the difference v is extended before being used. If the secret message b belongs to {0,1}, the difference extension is done by utilizing (2). Furthermore, v' is used to denote the difference between pixel’s pair having secret bit added to it. Thereafter, the new pixels u'_1 and u'_2 are generated using (3).

$$v' = 2 \times v + b \qquad (2)$$

$$u'_1 = m + \left\lfloor \frac{v'+1}{2} \right\rfloor \quad \text{and} \quad u'_2 = m - \left\lfloor \frac{v\prime}{2} \right\rfloor \qquad (3)$$

To avoid loss of the original data, both pixels u'_1 and u'_2 must not be underflow or overflow. The underflow means that the value of the pixel is less than 0 whereas in the case of overflow, the pixel's value exceeds 255. To prevent this problem from happening, two criteria in (4) have to be fulfilled.

$$\begin{cases} |v'| \le 2 \times (255 - m) & \text{if } 128 \le m \le 255 \\ |v'| \le 2 \times m + 1 & \text{if } 128 \le m \le 127 \end{cases} \qquad (4)$$

Notice that in the above DE method, only one bit of the secret message can be embedded into one pixel's pair. A new scheme that has the capability to improve this method was further proposed in [18]. Their experimental results show that it was improved in terms payload capacity, i.e., the secret bits that can be embedded in the original digital image were increased. Different from [27], 3 bits of the secret message can be embedded in one quad. Their scheme works as follows.

The image was first divided into blocks called quad of size 2 × 2. That is, four pixels were defined in each quad. Blocks were formed from both direction, left to right and top to bottom. Furthermore, as it is shown in (5) pixels in each quad were converted into a vector and an integer transformation was further defined. Each vector v has the form of $v = (v_o, v_1, v_2, v_3)$ which is obtained by calculating the difference between pixels in each block having pixels arranged in a vector $p = (u_o, u_1, u_2, u_3)$.

$$\begin{cases} v_o = \left\lfloor \frac{u_o + u_1 + u_2 + u_3}{4} \right\rfloor \\ v_1 = u_1 - u_o \\ v_2 = u_2 - u_1 \\ v_3 = u_3 - u_2 \end{cases} \qquad (5)$$

In order to embed the secret b, the difference values obtained in (5) have to be modified. Two different stages in (6) and (7) were considered.

1. Expanding the difference

$$\begin{cases} v'_1 = 2 \times v_1 + b_1 \\ v'_2 = 2 \times v_2 + b_2 \\ v'_3 = 2 \times v_3 + b_3 \end{cases} \qquad (6)$$

2. LSB modification

$$\begin{cases} v'_1 = 2 \times \left\lfloor \frac{v_1}{2} \right\rfloor + b_1 \\ v'_2 = 2 \times \left\lfloor \frac{v_2}{2} \right\rfloor + b_2 \\ v'_3 = 2 \times \left\lfloor \frac{v_3}{2} \right\rfloor + b_2 \end{cases} \qquad (7)$$

If the expression in (6) causes underflow or overflow, (7) is used otherwise the block is marked as non-changeable (no data are embedded to it). Notice that the bits of the secret message to be embedded are denoted by b_1, b_2, b_3 with all belonging to the set $b_n = \{0,1\}$. After embedding data, the new pixels are reconstructed by transforming the vector $v' = (v'_o, v'_1, v'_2, v'_3)$ into $p'= (u'_o, u'_1, u'_2, u'_3)$ using (8). The original pixel block p is replaced by the new one (p') containing the bits of the secret data in the stego image.

$$\begin{cases} u'_o = v'_0 - \left\lfloor \frac{u_o + u_1 + u_2 + u_3}{4} \right\rfloor \\ u'_1 = v'_1 + u_o \\ u'_2 = v'_2 + u'_1 \\ u'_3 = v'_3 + u'_2 \end{cases} \qquad (8)$$

Two criteria have to be fulfilled in order to ensure that the secret data are well embedded. First, the block p' has to meet conditions in (5-8). Second, the resulted pixels containing data must not be underflow or overflow. If the data are embedded using (6), the block p is said to be expandable whereas if it is carried out using (7), it is said to be changeable.

The work in [28] presented the data hiding scheme where the embedding process is performed by making use of smoothness level. The variance was utilized to determine which block to be embedded before the others. Furthermore, different from other research, they preferred to use the median as the base point. However, although the obtained experimental results show that their proposed method can yield a high payload capacity while maintaining the quality of the stego image, this method does not perform well for some images which results in distorting the quality of the stego image. Besides, their experiment shows that the embedding capacity is less than the one from the previous methods for certain images.

In the research carried out [29], an improved quad of quad and RDE algorithm was built. Similar to other methods, pixel blocks were divided into "expandable, changeable and non-changeable" respectively. The data embedding was done by first computing the difference v_n between pixel's pairs in each quad of quad and reducing v_n using the RDE scheme in (9) to get v''_n. Thereafter, the secret data were embedded to the reduced difference v''_n and finally the new pixel was calculated. Notice that their method suggested the reduction function presented in (9) where v''_n denotes the reduced difference expansion. The details about their method can be found in [29].

$$v''_n = \begin{cases} v_n - (2^{\lfloor \log_2 v_n \rfloor} + \lfloor \log_2 v_n \rfloor) \; if \; v_n > 1 \\ v_n + (2^{\lfloor \log_2 v_n \rfloor} + \lfloor \log_2 v_n \rfloor) \; if \; v_n > 1 \end{cases} \qquad (9)$$

III. The Proposed Method

This proposed method aims at improving the data hiding techniques by utilizing pixel block, constant base point for each block and the reduced difference expansion computed between pixels' pairs within the whole image. It is worth to note that this proposed method is designed to improve the capability of the existing techniques, especially the one presented in [29]. Therefore, our main contributions are detailed as follows. First, we enhance the expression for computing the reduced difference expansion (RDE) in order to get possible small values to be used while concealing the secret data. The main objective behind this enhancement is to allow data to be concealed while preserving the quality of the cover media. Second, the new constant base point for each pixel block is chosen differently for increasing the quality.

Third, we vary the size of pixel block which achieves a high embedding capacity while distorting the cover media from size of 4 × 4 to 2 × 2 .

A. Steps for Embedding

Given an image P of size F by P $(F \times P)$, the embedding steps are summarized as follows.

Step 1: It is first split up into m blocks or structures of size 2 × 2. That is, with this proposed method each block has 4 pixels. From Fig. 3, a block of pixel is represented by u_o, u_1, u_2, and u_3. To remove bewilderment, the terms block and structure are going to be used interchangeably.

Step 2: Before computing the difference between pixel's pairs, all pixels in each block are first stored as vector. If u_o, u_1, u_2, and u_3 are pixels in the first block, the vector is defined as $u_{vec} = (u_o, u_1, u_2, u_3)$. Similar to what was proposed in [29], pixel blocks are first categorized into three groups namely expandable, changeable and non-changeable. To avoid the problem of overflow and underflow, the secret data are only embedded in the first and second groups. Besides, as in the previous methods, for expandable blocks data are concealed using (6), while (7) is used for changeable blocks. Owing to the fact that non-changeable blocks can lead to the problem of underflow or overflow, they are disregarded during the embedding process.

Step 3: In contrast to the previous method, use the last pixel of each block as the base point. For the pixel block mentioned in Fig. 3, the base point would be u_3.

Step 4: Loop through all defined blocks and compute the difference between pixels' pairs using (10). It is also important to mention that (10) totally differs from the expression presented in [29]. Besides, as in [24], only three differences (v_0, v_1 and v_2) are computed.

$$\begin{cases} v_0 = u_0 - u_3 \\ v_1 = u_1 - u_3 \\ v_2 = u_2 - u_3 \\ v_3 = 0 \end{cases} \qquad (10)$$

However, in contrast to their method, v_3 is not being used to conceal the secret data for our method. That is, v_3 is not utilized ($v_3 = 0$) since the fourth pixel in each block is taken as the base point. Schematically the process in (10) can be viewed in Fig. 4.

Step 5: Hide data by first reducing the difference values v_0, v_1 and v_2 according to the defined rule. That is, compute the reduced difference expansion (RDE) for any difference (v_0, v_1 and v_2) which is greater than one or less than minus one (v_0, v_1 and v_2) >1 or (v_0, v_1 and v_2) <-1. Similar to [24] [29], values between 1 and -1 or (-1≤ v_n ≤1) are not reduced as they may result in distorting the secret message and the cover image.

Note that as mentioned before, the RDE expression in (9) which was implemented in [29] is enhanced in order to reduce the difference between pixel's pairs to the possible smallest value which is perfect for data to be embedded suitably. This enhancement is made by doubling the second logarithmic term. By doing this, as it is represented in (11), it could be easily seen that by utilizing the new proposed expression small difference values can be obtained. RDE is computed using both parts of (11), those are: (i) if $v_n > 1$, the first part is utilized; and (ii) if v_n <-1, the second part of the expression is applied.

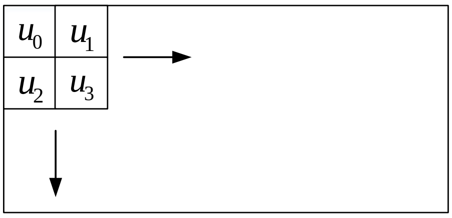


Fig. 3. Pixel block for the proposed method

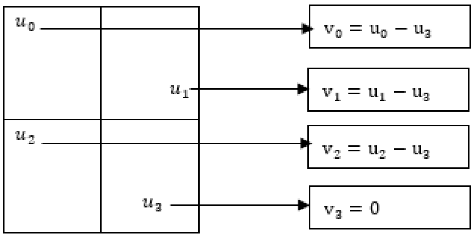


Fig. 4. Calculating the difference between pixel's pairs

$$v''_n = \begin{cases} v_n - (2^{\lfloor \log_2 v_n \rfloor} + 2(\lfloor \log_2 v_n \rfloor)) & if\ v_n > 1 \\ v_n + (2^{\lfloor \log_2 v_n \rfloor} + 2(\lfloor \log_2 v_n \rfloor)) & if\ v_n < -1 \end{cases} \qquad (11)$$

Here, v_n for each block starts from 0 to 3 $(0 \leq v_n \leq 3)$, $\forall\, n \in \mathbb{R}^+$except that $v_n = 3$ (v_3) is not utilized to conceal data for each block. The difference between the proposed RDE scheme (11) and the one in [29] as it is shown in (9) can be demonstrated as follows. considering a pixel block $u = (u_o, u_1, u_2, u_3)$ having pixel values $u_0 = 90,\ u_1 = 65,\ u_2 = 100\ and\ u_3 = 40$. By utilizing u_3 as the base point, the difference is computed using (10), thereafter we get the vector v having difference values $v_0, v_1, and\ v_2$.

$$\rightarrow v = (v_0, v_1, v_2)$$

$$\begin{cases} v_0 = u_0 - u_3 = 90 - 40 = 50 \\ v_1 = u_1 - u_3 = 65 - 40 = 25 \\ v_2 = u_2 - u_3 = 100 - 40 = 60 \\ v_3 = 0 \end{cases}$$

Since all difference values are still greater than one $(v_0, v_1\ and\ v_2 > 1)$, they have to be reduced before embedding data. Notice that all of these 3 difference values have to fulfill the same condition. Now let us evaluate how these two reduction schemes differ by first using (i) the existing RDE in (9); and (ii) the proposed RDE in (11) whose results can be summarized in Table I. Besides, steps for computing the difference are summarized in Fig. 5.

(i)Existing RDE → $v''_n = v_n - (2^{\lfloor \log_2 v_n \rfloor} + \lfloor \log_2 v_n \rfloor)$

$$\rightarrow v_0 = 50$$
$$v''_0 = 50 - (2^{\lfloor \log_2 50 \rfloor} + \lfloor \log_2 50 \rfloor)$$
$$v''_0 = 50 - (32 + 5)$$
$$v''_0 = 13$$

$$\rightarrow v_1 = 25$$
$$v''_1 = 25 - (2^{\lfloor \log_2 25 \rfloor} + \lfloor \log_2 25 \rfloor)$$
$$v''_1 = 25 - (16 + 4)$$
$$v''_1 = 5$$

$\rightarrow v_2 = 60$
$v''_2 = 60 - (2^{\lfloor \log_2 60 \rfloor} + \lfloor \log_2 60 \rfloor)$
$v''_2 = 60 - (32 + 5)$
$v''_2 = 23$

(ii)Proposed RDE $\rightarrow v''_n$
$= v_n + (2^{\lfloor \log_2 v_n \rfloor} + 2(\lfloor \log_2 v_n \rfloor))$
$\rightarrow v_0 = 50$
$v''_0 = 50 - (2^{\lfloor \log_2 50 \rfloor} + 2(\lfloor \log_2 50 \rfloor))$
$v''_0 = 50 - (32 + 10)$
$v''_0 = 8$

$\rightarrow v_1 = 25$
$v''_1 = 25 - (2^{\lfloor \log_2 25 \rfloor} + 2(\lfloor \log_2 25 \rfloor))$
$v''_1 = 25 - (16 + 8)$
$v''_1 = 1$

$\rightarrow v_2 = 60$
$v''_2 = 60 - (2^{\lfloor \log_2 60 \rfloor} + 2(\lfloor \log_2 60 \rfloor))$
$v''_2 = 60 - (32+10)$
$v''_2 = 18$

Notice that all values between -1 and 1 are not reduced but they are still being used for embedding data. From the reduced differences (v_0, v_1 and v_2) obtained in (i) and (ii), we find that by using the proposed RDE scheme in (11), small difference values are generated compared to the ones obtained using (9). These small difference values that are generated after the reduction process are used for embedding data by utilizing (6) or (7). To compute the new pixel in the stego image, in contrast to [18] [24] [29], we provide (12). Furthermore, to prevent the cover image from being worsened, the secret data are not embedded in the last pixel of each block (u_3) since it is taken as the base point.

$$\begin{cases} u'_0 = v'_0 + u_3 \\ u'_1 = v'_1 + u_3 \\ u'_2 = v'_2 + u_3 \\ u'_3 = u_3 \end{cases} \qquad (12)$$

To prevent underflow and overflow, each new pixel u'_n in each block must fulfill the condition $0 \leq u'_n = v'_n + u_n \leq 255$ otherwise the whole block is marked as non-changeable. Note that u_n denotes the last pixel in each block and v'_n denotes the difference having secret bit after using (6) or (7). As in [24] [29], the location map LM is utilized in our proposed method. The main purpose of the location map is to keep track of the embedding information for each block which makes the extraction straightforward if it is well defined and recorded. To make the process clear, the bit 1 in the location map indicates that the expansion in (6) was utilized while 0 shows that LSB in (7) was used to embed data. For example, from (11) two blocks are defined. That is, expandable RDE if the first or second condition are met and non-expandable RDE if (11) is not fulfilled. Moreover, -1 is used to represent those pixel blocks which are unchanged.

Each pixel block's information in the location map is stored in the form of vector. That is, the location map vector $LM = (LM1, LM2, LM3, LM4, LM5)$ is defined and allocated as follows. Assign bits 1, 0 and -1 for expandable, changeable and non-changeable pixel blocks correspondingly. As well as that, for expandable $LM1 = 1$ is defined and $LM2 = 1$ is assigned for expandable RDE. Additionally, for those blocks falling in the category of expandable RDE, it is also important to assign the location map to keep track of information about each pixel reduction. That is, if $v''_n \pm (2^{(\lfloor \log_2 v''_n \rfloor)-1}) + 2(\lfloor \log_2 v''_n \rfloor) = v_n$, then $LM3, LM4,$ $LM5$ are set to 0 and if $v''n \pm (2^{\lfloor \log_2 | v''n | \rfloor} + 2(\lfloor \log_2 | v''_n | \rfloor)) \neq v_n$, then the location maps $LM3, LM4, LM5$ take the value of 1. To distinguish expandable block categories, $LM1 = 1$ and $LM2 = 0$ are further assigned to the blocks which are non RDE expandable. This non RDE expandable block means that only those values which are between -1 and 1 are directly utilized without having to be reduced. Furthermore, $LM1 = 0$ is for changeable blocks. If the differences (v_0, v_1, v_2) are odd, then the location maps $LM3$, $LM4$ and $LM5$ are set to 1; and if the differences are even, then these location maps are

TABLE I
COMPARING THE REDUCED DIFFERENCE

Original difference (v_n)	Reduced difference	Al_Huti [29]	Proposed method
$v_0 = 50$	v''_0	13	8
$v_1 = 25$	v''_1	5	1
$v_2 = 60$	v''_2	23	18
	Average →	13.66	9

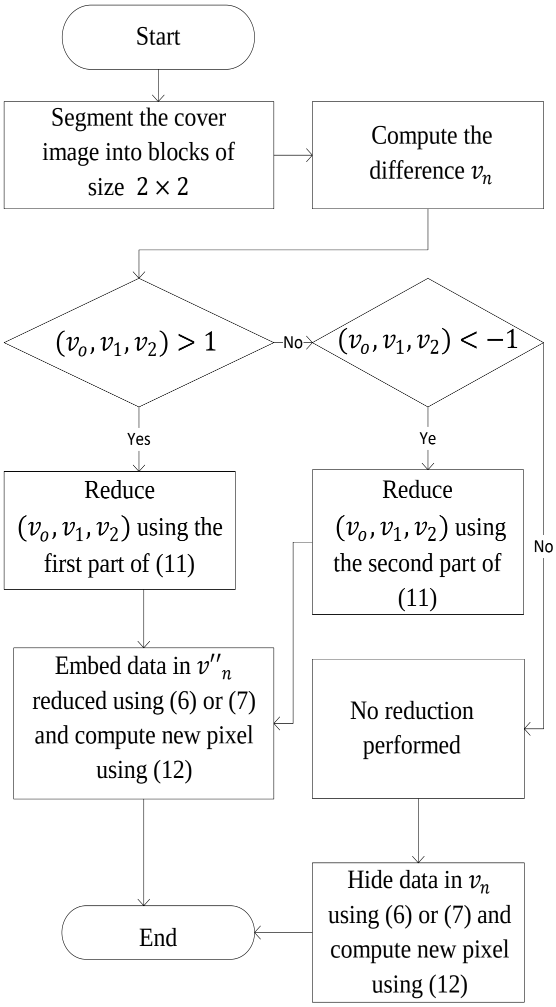

Start
Segment the cover image into blocks of size 2×2
Compute the difference v_n
$(v_o, v_1, v_2) > 1$
No
$(v_o, v_1, v_2) < -1$
Yes
Ye
No
Reduce (v_o, v_1, v_2) using the first part of (11)
Reduce (v_o, v_1, v_2) using the second part of (11)
Embed data in v''_n reduced using (6) or (7) and compute new pixel using (12)
No reduction performed
Hide data in v_n using (6) or (7) and compute new pixel using (12)
End

Fig. 5. Computing and reducing the difference

set to 0. Finally, the stego image and the location map are kept apart. Notice that at this stage the concealment process is done, the stego image and the location map can be sent to the intended recipient through the public network.

A. *Steps for Extraction*

The extraction process is performed in order to obtain the hidden secret message and it is carried out as follows.

Step1: The extraction phase begins by first dividing the stego image into blocks, each having four pixels, thereafter the difference between pixels' pairs is computed using (10), i.e., $v''_n = (v'_0,\ v'_1,\ v'_2)$ is computed in each block after that the location map is utilized to get the secret message and the value of the original pixels. Perform the extraction of expandable RDE if only the location maps $LM1 = 1$ and $LM2 = 1$. Process non-RDE expandable if $LM1 = 1$ and $LM2 = 0$. Moreover, if $LM1 = 0$, the changeable blocks can be accessed. To be able to process non-changeable blocks, the defined location map $LM1$= -1 is utilized.

Step2: Recovering the original difference and the secret bits for RDE expandable. First, to get the secret bits the LSB is extracted from v''_n, after that v''_n has to be right shifted in order to get the original difference thereafter the original difference v_n is recovered as follows.

First, if v''_n >1 and $LM3, LM4, LM5 = 0$, use (13) to get the original difference v_n.

$$v_n = v''_n + (2^{(\lfloor \log_2 v''n \rfloor)-1} + 2(\lfloor \log_2 v''_n \rfloor) - 1) \quad (13)$$

Second, if v''_n >1 and $LM3, LM4, LM5 = 1$, then (14) is utilized to get v_n.

$$v_n = v''_n - (2^{(\lfloor \log_2 v''n \rfloor)-1} + 2(\lfloor \log_2 |v''_n| \rfloor) - 1) \ (14)$$

If $v''n$ <-1 and $LM3, LM4, LM5 = 1$, use (15) to obtain v_n and then compute the new pixel using (16), where $v_n = (v_o, v_1, v_2, v_3\ )$.

$$v_n = v''_n - (2^{\lfloor \log_2 |v''n| \rfloor} + 2(\lfloor \log_2 |v''_n| \rfloor)) \qquad (15)$$

$$\begin{cases} u_0 = v_0 + u_3 \\ u_1 = v_1 + u_3 \\ u_2 = v_2 + u_3 \\ u_3 = u_3 \end{cases} \qquad (16)$$

If $v''_n < -1$ and $LM3, LM4, LM5 = 0$, utilize (17) to get v_n and calculate the new pixel using (16).

$$v_n = v''_n + (2^{\lfloor \log_2 |v''n| \rfloor} + 2(\lfloor \log_2 |v''_n| \rfloor)) \qquad (17)$$

To process non-RDE expandable blocks, the secret message is obtained by taking LSB of v''_n and the expression defined in (18) is used to get v_n.

$$v_n = \left\lfloor \frac{v''_n}{2} \right\rfloor \qquad (18)$$

The secret bits are extracted from changeable blocks by taking the LSB of v''_n using modulus function (mod 2 of v''_n) Thereafter, the original difference v_n is computed as follows.

a. If the location map $LM3, LM4, LM5 = 0$ and the difference v''_n is odd, then the recovery is carried out using (19)

$$v_n = 2 \times \left\lfloor \frac{v''_n}{2} \right\rfloor - 1 \qquad (19)$$

b. If the location map $LM3, LM4,\ LM5 = 1$ and the difference v''_n is even, then use (20) to recover v_n.

$$v_n = 2 \times \left\lfloor \frac{v''n}{2} \right\rfloor + 1 \qquad (20)$$

Note that the expression in (19) and (20) are similar to the ones presented in [29]. Furthermore, in (a) if the location map $LM3, LM4, LM5 = 0$, the difference cannot be even and this is similar to (b), if $LM3, LM4,\ LM5 = 1$, the difference cannot be odd and this is because these location maps were defined during the embedding process to keep track of information about any operation done in changeable blocks.

Generally, the differences between the method in [29] and the proposed one are provided in Table II.

TABLE II
COMPARISON BETWEEN THE METHOD OF AL_HUTI ET AL [29] AND THE PROPOSED METHOD

Stage	Method of Al_Huti et al [29]	Proposed method
Computing difference between pixel pairs	$\begin{cases} v_0 = 0 \\ v_1 = u_1 - u_0 \\ v_2 = u_2 - u_1 \\ v_3 = u_3 - u_2 \end{cases}$	$\begin{cases} v_0 = u_0 - u_3 \\ v_1 = u_1 - u_3 \\ v_2 = u_2 - u_3 \\ v_3 = 0 \end{cases}$
Reduction function for RDE	$v''_n = \begin{cases} v_n - (2^{\lfloor \log_2 v_n \rfloor} + \lfloor \log_2 v_n \rfloor)\ if\ \ v_n > 1 \\ v_n +\ (2^{\lfloor \log_2 v_n \rfloor} + \lfloor \log_2 v_n \rfloor)\ if\ v_n <\ -1 \end{cases}$	$v''_n = \begin{cases} v_n - (2^{\lfloor \log_2 v_n \rfloor} + 2(\lfloor \log_2 v_n \rfloor)) \quad if\ v_n > 1 \\ v_n + (2^{\lfloor \log_2 \lvert v_n \rvert \rfloor} + 2(\lfloor \log_2 \lvert v_n \rvert \rfloor))\ \ if\ v_n < -1 \end{cases}$
Pixel block	4×4	2×2
Base point pixel	u_0	u_4 (constant for each block)
Computing new pixel	$\begin{cases} u'_0 = u_0 \\ u'_1 = v'_1 + u'_0 \\ u'_2 = v'_2 + u'_1 \\ u'_3 = u_3 +\ u'_2 \end{cases}$	$\begin{cases} u'_0 = v'_0 + u_3 \\ u'_1 = +v'_1 + u_3 \\ u'_2 = +v'_2 + u_3 \\ u_3 = u_3 \end{cases}$

IV. Experimental Results

After extracting the secret data, it is important to measure some similarities between the extracted data and the original ones. If they match, we say that they are identical and trustworthy. That is, the process was well performed. Besides, the main goal of the experiment is to measure and evaluate the distortion level of stego image with respect to the number of secret bits that are concealed which is carried out by measuring the visual quality of the stego image. Additionally, since MATLAB does provide many features such as manipulating matrix, function and data plotting, implementing algorithms, creating user interface etc., it is chosen to be used for implementing the proposed approach. This algorithm is developed and tested in a laptop computer having the following specifications: 64-bit Windows 8.1 Professional, Intel (R) Core (TM) i3-4005U CPU@ 1.70 GHz Processor with no Pen or Touch input display.

Furthermore, it is also important to notice that to evaluate how well the proposed approach does perform compared to the previous ones, we have also implemented the scheme presented in [29] and drawn conclusion about the performance of the proposed approach by comparing the obtained peak signal-to-noise ratio achieved after concealing data using both methods. Moreover, 10 well-known standard cover images of size 512 × 512 obtained from [30] [31] are utilized to evaluate the performance of the proposed method on the given size of the secret data as provided in Table III and IV. Note that all images are freely available to be used. A binary bit stream of the secret data whose size depends upon the needed payload capacity to be embedded in the image is randomly generated using a function available in MATLAB. During our experiment, five different sizes of the secret message (with $size_1 = 16569$ bits, $size_2 = 37629$ bits , $size_3 = 90000$ bits, $size_4 = 147762$ bits and the last one ($size_5$) having 196508 bits) are first randomly generated, thereafter they get stored into five different text files to make sure that the same secret message is utilized for all images throughout the experiment.

Additionally, the PSNR is computed to analyze and evaluate how the stego image degrades with respect to the original cover image. If the PSNR value is high, the quality of the stego image is better. That is, the cover image is not drastically distorted. Besides, the value of the PSNR is computed using (21), where MSE is obtained using (22).

$$\mathrm{PSNR} = 10\log_{10}\frac{(\mathrm{MAX})^2}{\mathrm{MSE}} \qquad (21)$$

$$\mathrm{MSE} = \left(\frac{1}{\mathrm{WH}}\right)\sum_{i=1}^{H}\sum_{j=1}^{W}(P_{ij} - M_{ij})^2 \qquad (22)$$

In (21), MAX is used to denote the maximum pixel value while W and H represent the width and height of the image respectively. In (22), P_{ij} represents the pixel's value in the original image and M_{ij} corresponds to the stego image pixel's value which is located at (i,j) position. The MSE refers to the mean squared error. It gives information about how the stego image P' differs from the original image P. From the experimental results shown in Table III and Table IV, it is found that with the proposed method, a good PSNR is achieved. That is, by considering the quality, the proposed method outdoes the one implemented in [29]. Furthermore, since the proposed method has improved the previous reduced difference expansion (RDE-scheme) in [29], it is crucial to visualize it in Fig. 6 by plotting the results from Table I. From Fig. 6 by utilizing the proposed RDE scheme, small difference values are obtained, which results in a good payload capacity as well as a good visual quality of the stego image. Additionally, it is also important to note that the pixel block and the suggested constant base point for each block have also greatly influenced the quality of stego image.

To observe the changes made in the cover images after embedding the secret data, Fig. 7 depicts Hand original medical image, Pepper original image and their respective stego images obtained after concealing 16569 bits of the

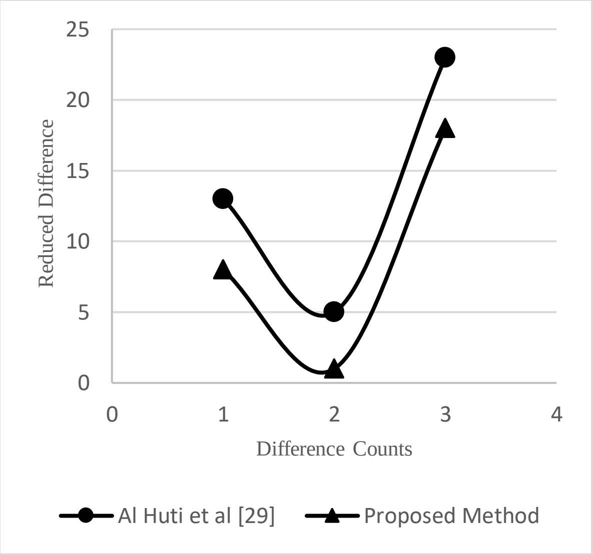
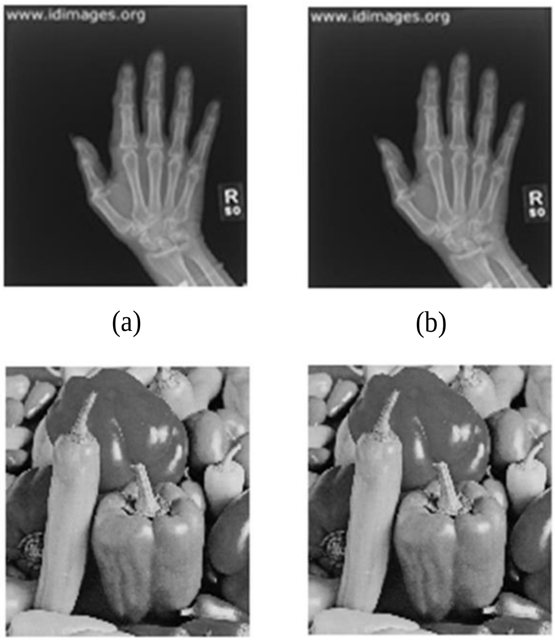


Fig. 6. Variation of the reduced difference using the proposed method and the one implemented by Al_Huti et al. [29].



(a) (b)

(c) (d)

Fig. 7. An example of original images (a) Hand image before hiding data (b) Hand image after hiding 16569 bits (c) Pepper image before hiding data (d) Pepper image after hiding 16569 bits using the proposed method.

TABLE III
Comparison between The Method of Al_Huti et al. [29] and the Proposed One by Using General Grayscale Images [30]

Images	Al_Huti et al. [29]		Proposed method	
	Capacity (bits)	PSNR (dB)	Capacity (bits)	PSNR (dB)
Girl	16569	41.84	16569	42.54
	37629	36.80	37629	37.77
	90000	32.35	90000	33.54
	147762	30.41	147762	31.63
	196508	27.16	196508	30.25
Aeroplane	16569	41.06	16569	41.59
	37629	38.69	37629	39.09
	90000	31.23	90000	32.16
	147762	27.45	147762	28.65
	196508	25.89	196508	27.36
Lena	16569	46.58	16569	48.27
	37629	40.33	37629	42.03
	90000	33.15	90000	34.96
	147762	29.60	147762	31.47
	196508	28.30	196508	29.98
Pepper	16569	33.88	16569	34.55
	37629	32.74	37629	32.79
	90000	29.66	90000	30.04
	147762	28.14	147762	28.82
	196508	27.00	196508	27.72
Elaine	16569	41.11	16569	42.16
	37629	37.30	37629	38.54
	90000	32.16	90000	33.57
	147762	29.89	147762	31.22
	196508	28.69	196508	29.99

TABLE IV
Comparison between Method of Al_Huti et al. [29] and the Proposed Method by Using Medical Grayscale Images [31]

Images	Al_Huti et al. [29]		Proposed method	
	Capacity (bits)	PSNR (dB)	Capacity (bits)	PSNR (dB)
Lung	16569	46.33	16569	46.92
	37629	44.47	37629	45.02
	90000	41.96	90000	44.03
	147762	40.48	147762	41.31
	196508	38.41	196508	39.14
Hand	16569	42.00	16569	43.46
	37629	41.95	37629	43.33
	90000	40.76	90000	41.89
	147762	38.03	147762	38.94
	196508	37.61	196508	38.55
Abdominal	16569	43.95	16569	44.62
	37629	42.38	37629	42.99
	90000	40.78	90000	41.26
	147762	38.03	147762	39.96
	196508	37.81	196508	38.39
Head	16569	42.45	16569	42.77
	37629	40.45	37629	40.81
	90000	35.43	90000	36.71
	147762	32.57	147762	33.89
	196508	31.63	196508	32.86
Leg	16569	47.21	16569	48.37
	37629	43.84	37629	44.84
	90000	40.43	90000	41.24
	147762	39.59	147762	40.26
	196508	38.21	196508	38.79

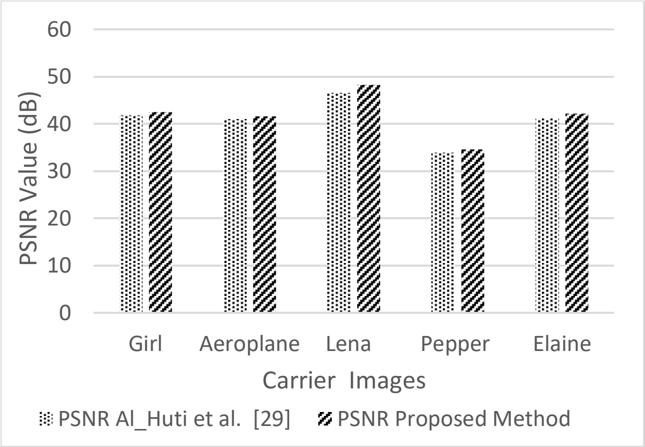


Fig. 8. PSNR variation after hiding 16569 bits in general images

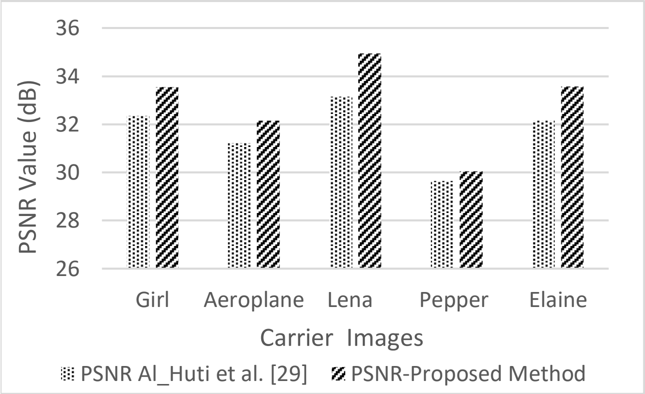


Fig. 9. PSNR variation after hiding 90000 bits in general images

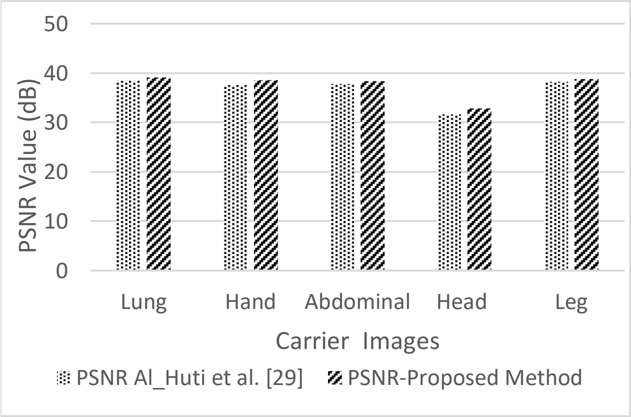


Fig. 10. PSNR variation after hiding 196508 bits in medical images

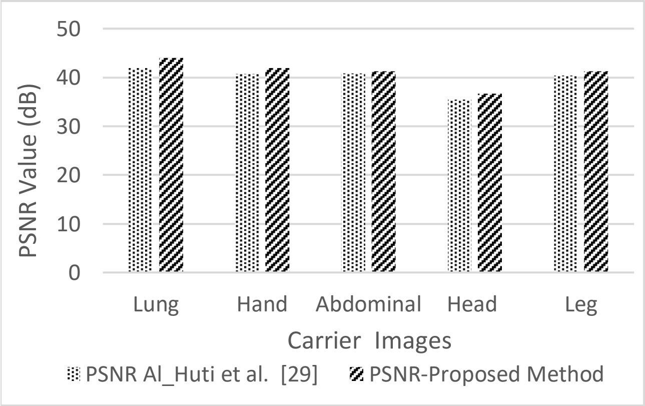


Fig. 11. PSNR variation after hiding 90000 bits in medical images

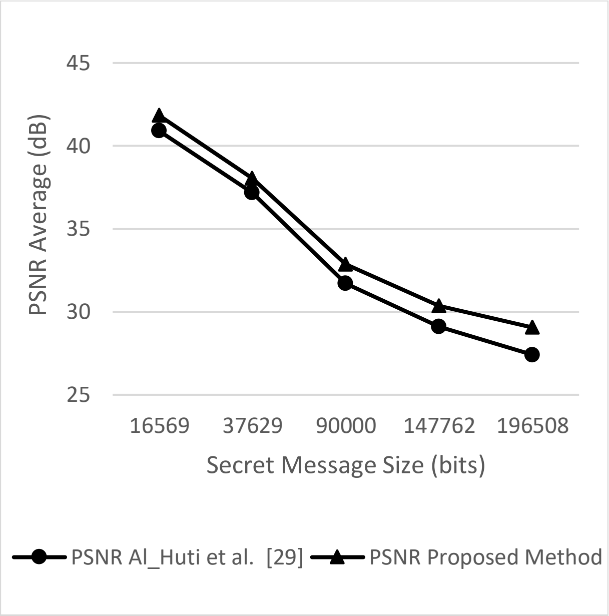


Fig. 12. The Overall PSNR average for all general cover images with respect to each secret message size

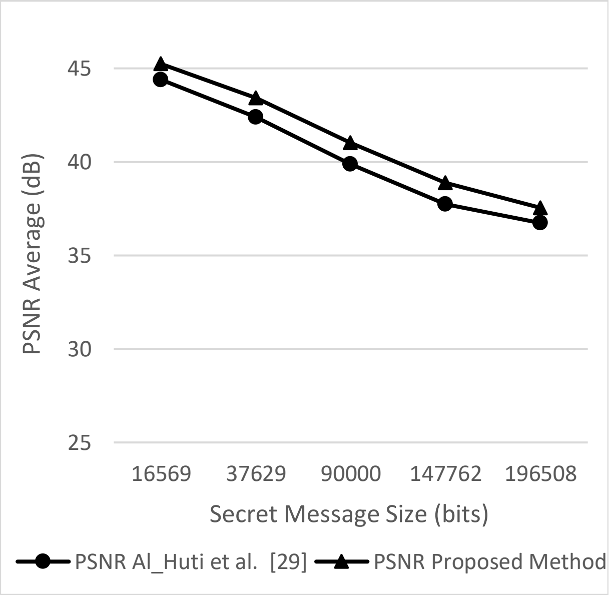


Fig. 13. The Overall PSNR average for all medical cover images with respect to each secret message size

secret data. From Fig. 7 we can see that the visual quality of the cover image is maintained. Besides, if we look at both Figs. 7(a) and 7(b) as well as 7(c) and 7(d), they are almost similar which makes the proposed method to be judged highly invisible, i.e., it is really difficult to identify the difference between them (the original and the stego images), which means that there is a high similarity between the stego image and the original cover image. Moreover, due to the fact that the reduced difference expansion scheme effectively reduces the difference values, this keeps good quality of the stego image. However, the results presented in Table III and Table IV show that the quality depends on the number of secret bits concealed in the cover image.

Again, from the experimental results, it could be seen that after concealing five different sizes of the secret message into all images, good PSNR is achieved and this results in a good visual quality of the stego image. Based on the overall results, it can be inferred that the proposed method has greatly enhanced the previous one [29]. This enhancement allows high quality application for data hiding which is highly desirable. Nevertheless, as mentioned above, the value of PSNR goes down after hiding more secret bits in the cover image. That is, after concealing 196508 bits, the PSNR value slightly decreases. The highest PSNR (48.37 dB, see Table IV) is obtained after concealing 16569 bits in Leg medical image while the lowest PSNR is obtained from Aeroplane image (27.36 dB, see Table III) after hiding 196508 bits. For Leg medical image, the idea is that values obtained after computing the difference between pixels' pairs were further reduced to the possible smallest values which results in a good PSNR. For Aeroplane image, it means that although the difference values were further reduced using the proposed RDE, the values generated after the reduction process are still large compared to the ones from Leg. In addition, if there is a high disparity between the neighboring pixels in each block, it results in large difference values which may reduce the quality. Considering all sizes of the secret message, the PSNR from some images tend to be close to each other which implies that the difference values obtained after reduction are almost in the same range. Nonetheless, as the trade-off, there is always a slight change in the quality of stego image whenever the payload capacity is increased or decreased.

As a result, this proposed method can be highly preferable to individuals willing to conceal low or medium payload capacity with low perceptibility and high security level. The reason is that it will be difficult to suspect the existence of the secret data in the stego image while being transmitted to the intended recipients via the internet. Moreover, this will also increase the level of confidentiality and privacy between the communicating parties. The results' visualization about the performance of the scheme in [29] and the proposed one can be viewed from Figs. 8, 9, 10, 11,12 and 13. From Fig. 8 and 9 we could see that after concealing both sizes, the proposed method outperforms Al_Huti et al.'s scheme [29] in terms of the visual quality. Moreover, Figs. 10 and 11 show that for all secret message sizes, our method is still achieving good PSNR compared to the previous one. Figs. 12 and 13 depict the overall PSNR average for all images with respect to each secret message for both general and medical images. The PSNR average is computed based on five secret message sizes which are used to evaluate the distortion level (or changes) encountered in the cover image after hiding each message size. For example, for Girl, Aeroplane, Lena, Pepper and Elaine cover images, the PSNR average (in this case 41.829 dB) after hiding the first secret message size (16569 bits) is obtained by computing the average of five PSNR values (PSNR from each cover image, i.e., PSNR_Girl= 46.92 dB, PSNR_Aeroplane = 41.59 dB, PSNR_Lena = 48.27 dB, PSNR_Pepper = 34.55 dB and PSNR_Elaine = 42.16 dB). For medical cover images the process is similar. Surprisingly, the overall good PSNR average is achieved in medical images. This is because these images are characterized by a high redundancy which permits data to be concealed without much distorting them. The proposed approach has not only improved the quality but also the number of bits that can be hidden in the image. Generally, this proposed approach can be suitable for all users depending on the needed embedding capacity.

V. Conclusion and Future Work

Digital image steganography is one of the interesting research areas in information hiding. If it is used properly, the reliable communication, data security and the privacy of the communicating parties can be well maintained. This paper presents a new method developed based on pixel block, reduced difference expansion and constant base point for hiding secret data in both general (non-medical) and medical grayscale images that achieves good PNSR and good embedding capacity. That is, by considering the quality, this proposed method provides better results for corresponding secret size. The quality of the cover image is varying proportionally with respect to the payload capacity, i.e., increasing the capacity results in degrading the cover image. Since the payload capacity and the quality of the stego image are critical factors to be considered while concealing data in any cover media, in our future work we will focus on increasing the embedding capacity while conserving the quality of the cover image.

Acknowledgment

The research work is sponsored by ITS and the Ministry of Research, Technology and Higher Education, Republic of Indonesia.

References

[1] Y.-Y. Tsai, D.-S. Tsai and C.-L. Liu, "Reversible data hiding scheme based on neighboring pixel differences," *Digital Signal Processing*, vol. 23, pp. 919-927, 2013. https://doi.org/10.1016/j.dsp.2012.12.014

[2] D.-C. Lou, M.-C. Hu and J.-L. Liu, "Multiple layer data hiding scheme for medical images," *Computer Standards & Interfaces*, vol. 31, p. 329–335, 2009. https://doi.org/10.1016/j.csi.2008.05.009

[3] G. Gao, . X. Wan, . S. Yao, . Z. Cui, C. Zhou and X. Sun, "Reversible data hiding with contrast enhancement and tamper localization for medical images," *Information Sciences*, vol. 385–386, p. 250–265, 2017. https://doi.org/10.1016/j.ins.2017.01.009

[4] B. Datta, U. Mukherjee and S. K. Bandyopadhyay, "LSB Layer Independent Robust Steganography using Binary Addition," in *International Conference on Computational Modeling and Security (CMS)*, 2016. https://doi.org/10.1016/j.procs.2016.05.188

[5] G. Swain, "A steganographic method combining LSB substitution and PVD in a block," in *International Conference on Computational Modelling and Security (CMS 2016)*, 2016. https://doi.org/10.1016/j.procs.2016.05.174

[6] M. Tayel, A. Gamal and H. Shawky, "A Proposed Implementation Method of an Audio Steganography Technique," in *2016 18th International Conference on Advanced Communication Technology (ICACT)*, 2016. https://doi.org/10.1109/ICACT.2016.7423320

[7] J. Blackledge and A. Al-Rawi, "Steganography Using Stochastic Diffusion for the Covert Communication of Digital Images," *IAENG International Journal of Applied Mathematics*, vol. 41, pp. 270 - 298, 2011.

[8] S. H. El-sayed, S. . F. El-Zoghdy and S. O. Faragallah, "Adaptive Difference Expansion-Based Reversible Data Hiding Scheme for Digital Images," *Arabian Journal for Science and Engineering*, vol. 41, p. 1091–1107, 2016. https://doi.org/10.1007/s13369-015-1956-7

[9] M. S. Subhedar and V. H. Mankar, "Current status and key issues in image steganography: A survey," *Computer Science Review,* vol. 13–14, p. 95–113, 2014. https://doi.org/10.1016/j.cosrev.2014.09.001

[10] M. Hussain, A. W. A. Wahab, A. . T. Ho, N. Javed and K.-H. Jung, "A data hiding scheme using parity-bit pixel value differencing and improved rightmost digit replacement," *Signal Processing: Image Communication,* vol. 50, p. 44–57, 2017. https://doi.org/10.1016/j.image.2016.10.005

[11] S. Weng, Y. Liu, J.-S. Pan and N. Cai, "Reversible data hiding based on flexible block-partition and adaptive block-modification strategy," *Journal of Visual Communication and Image Representation,* vol. 41, p. 185–199, 2016. https://doi.org/10.1016/j.jvcir.2016.09.016

[12] S. Agrawal and . M. Kumar, "Mean value based reversible data hiding in encrypted images," *Optik - International Journal for Light and Electron Optics,* vol. 130, p. 922–934, 2017. http://doi.org/10.1016/j.ijleo.2016.11.059

[13] E. Ghasemi, J. Shanbehzadeh and N. Fassihi, "High Capacity Image Steganography usingWavelet Transform and Genetic Algorithm," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 2011.

[14] R. Tanwar and M. Bisla, "Audio steganography," in *2014 International Conference on Optimization Reliabilty, and Information Technology (ICROIT),* 2014. https://doi.org/10.1109/icroit.2014.6798347

[15] M. B. Andra, T. Ahmad and T. Usagawa, "Medical Record Protection With Improved GRDE Data Hiding Method on Audio Files," *Engineering Letter*s, vol 25 no 2, pp 112-124, 2017.

[16] M. Thangamani and J. A. Ibrahim. S, "Knowledge Exploration in Image Text Data using Data Hiding Scheme," in *Proceedings of the International MultiConference of Engineers and Computer Scientists(IMECS),* 2017.

[17] A. Arham, . H. . A. Nugroho and T. B. Adji, "Multiple layer data hiding scheme based on difference expansion of quad," *Signal Processing,* vol. 137, p. 52–62, 2017. https://doi.org/10.1016/j.sigpro.2017.02.001

[18] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Processing,* vol. 13, pp. 1147 - 1156, 2004. https://doi.org/10.1109/TIP.2004.828418

[19] L. Xin, W. Qiao-yan, Z. Ze-li and Z. Jie, "A Novel Steganographic Method with Four-Pixel Differencing and Modulus Function," *Fundamenta Informaticae,* Vols. 118, no. 3, pp. 281-289, 2012. http://dx.doi.org/10.3233/FI-2012-714

[20] W.-C. Kuo, J.-J. Li , C.-C. Wang , L.-C. Wuu and Y.-C. Huang, "An Improvement Data Hiding Scheme Based on Formula Fully Exploiting Modification Directions and Pixel Value Differencing Method," in *2016 11th Asia Joint Conference on Information Security (AsiaJCIS),* 2016. https://doi.org/10.1109/asiajcis.2016.20

[21] Himakshi, R. K. Singh, H. . K. Verma and C. K. Singh, "Bi-directional pixel-value differencing approach for RGB color image," in *2013 Sixth International Conference on Contemporary Computing (IC3),* 2013. https://doi.org/10.1109/IC3.2013.6612248

[22] G. Swain and S. K. Lenka, "Pixel value differencing steganography using correlation of target pixel with neighboring pixels," in *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT),* 2015. https://doi.org/10.1109/ICECCT.2015.7226029

[23] X. Wang, J. Ding and Q. Pei, "A novel reversible image data hiding scheme based on pixel value ordering and dynamic pixel block partition," *Information Sciences,* vol. 310 , p. 16–35, 2015. https://doi.org/10.1016/j.ins.2015.03.022

[24] T. Ahmad, . M. Holil, W. Wibisono and I. R. Muslim, "An improved Quad and RDE-based medical data hiding method," in *2013 IEEE International Conference on, Computational Intelligence and Cybernetics (CYBERNETICSCOM),* 2013. https://doi.org/10.1109/CyberneticsCom.2013.6865798

[25] A. Tyagi, R. Roy and S. Changder, "High Capacity Image Steganography Based on Pixel Value Differencing and Pixel Value Sum," in *2015 Second International Conference on Advances in Computing and Communication Engineering (ICACCE),* 2015. https://doi.org/10.1109/ICACCE.2015.92

[26] M. Kalita and T. Tuithung, "A novel steganographic method using 8-neighboring PVD (8nPVD) and LSB substitution," in *2016 International Conference on Systems, Signals and Image Processing (IWSSIP),* 2016. https://doi.org/10.1109/IWSSIP.2016.7502756

[27] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 13, pp. 890 - 896, 2003. https://doi.org/10.1109/TCSVT.2003.815962

[28] M. Holil and T. Ahmad, "Secret Data Hiding by Optimizing General Smoothness Difference Expansion-Based Method," *Journal of Theoretical and Applied Information Technology,* vol. 72 No.2, 2015.

[29] M. H. A. Al_Huti, T. Ahmad and S. Djanali, "Increasing the capacity of the secret data using pixels blocks and adjusted RDE-based on grayscale images," in *International Conference on Information and Communication Technology and Systems*, Surabaya, Indonesia, 2015. https://doi.org/10.1109/ICTS.2015.7379903

[30] "Califonia UUo, “SIPI Image Database,“," [Online]. Available: http://sipi.usc.edu/database/database.php?volume=misc&image=11. [Accessed 22 December 2016].

[31] "Library Ed, “Partners Infectious Disease Images“.," [Online]. Available: http://www.idimages.org/images/browse/Image Technique/.. [Accessed 26 January 2017].

Authors’ Biography

Pascal Maniriho received his Bachelor of Technology with Honors in Information and Communication Technology from Umutara Polytechnic, Rwanda, in September 2013. He is currently pursuing his Master of Informatics at Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. His research interests include data hiding, network and database security, ad hoc networks, wireless sensor networks, big data analysis and pattern classification. He is a member of IAENG.

Tohari Ahmad has obtained his Bachelor, Master and Ph.D. degree from Institut Teknologi Sepuluh Nopember (Indonesia), Monash University (Australia) and RMIT University (Australia), respectively. All are in computer science and information technology.

He is now a researcher at Department of Informatics, Institut Teknologi Sepuluh Nopember, Indonesia. His research interest is in data hiding, biometric security and information security. He is member of IEEE, ACM, IAENG.