




Improved center-folding based directional pixel value ordering for reversible data hiding scheme

Sudipta Meikap¹ · Biswapati Jana² 

Received: 27 April 2020 / Revised: 23 August 2020 / Accepted: 2 September 2020 /
Published online: 8 October 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In the context of Reversible Data Hiding scheme (RDH), twin image-based methods have been widely used in recent years. Pixels of any cover image block are organized in ascending order and then modify the largest and smallest pixel to embed hidden information exploiting Pixel Value Ordering (PVO) techniques. The DPVO (Directional PVO) has been utilized in three orientations such as a horizontal, vertical and crosswise line applied one after other. Meanwhile, center folding method compresses the valuable hidden information and then embedded it within pixels of dual stego-images through averaging. The proposed scheme embed more than two data bits positions in any row of the image block which is possible by introducing a new parameter α , which was not reported by other PVO based RDH schemes. The Embedding Capacity (EC) is improved without compromising visual quality when the secret information is embedded using the center folding strategy combining with the DPVO scheme. The proposed method uses different image block sizes to demonstrate the relationship between the data hiding rate with visual quality of the stego image. The experimental outcomes indicate that the suggested method is suitable to embed high amount of hidden data with a good visual quality, that can be assured by comparing with other state-of-the-art methods. The intended result highlighted some impressive sublime features in the field of image identification, manipulation and forgery detection in which technical life stunts. This system profits enormously from numerous aspects of government and the private sector including education, economic protection, defence, intellectual property rights.

Keywords Reversible data hiding · Center-folding · Directional pixel-value-ordering · Embedding capacity · Steganalysis · Steganographic attacks

✉ Sudipta Meikap
sudiptameikap@gmail.com

Biswapati Jana
biswapatijana@gmail.com

¹ Department of Computer Science, Hijli College, Paschim Medinipur, West Bengal, 721306, India

² Department of Computer Science, Vidyasagar University, Midnapore, West Bengal, 721102, India

1 Introduction

With the improvement of information technology, secret data communication is raising over the internet for common users. During message transfer the secret data may be damaged, illegally observed, copied fraudulently, or stolen from sender to receiver or vice versa. Data hiding plays an important role in multimedia security. It is useful in various purposes such as copyright protection, covert communication, content authentication, forensic tracking, tamper detection and many other human centric applications. Due to the presence of hidden secret information, a cover image has some distortion that can't be rectified in comparison with the original cover image. This distorted image affects the visual quality which is imperceptible from human eyes. This slight error is acceptable for the data hiding technique while it is unacceptable for some real field areas like remote sensing, medical and military images. According to whether the cover image pixels can be recovered or not after data extraction, current data hiding schemes are classified into two categories: reversible and irreversible. The scheme of reversible data hiding usually exploits the techniques of histogram shifting, prediction error, and difference expansion etc. After data extraction, irreversible methods are inadequate to reconstruct the real cover image whereas reversible methods can reconstruct the real image. In Reversible Data Hiding (RDH), not only the secret message but also the cover image is recovered from the stego image which is essential in many real life applications such as military, legal and medical images. RDH is generally a fragile approach for verification of quality and integrity, and is assessed by its efficiency of capacity-distortion. Recent times the demand for reliably secure, high-capacity transmission of information via hiding data is increasing. But shielding data will be a double-edged sword, since serious and criminal attackers would use it. Eavesdroppers can use this to disrupt social order, threaten public security and engage in illegal activity. Design of any defense scheme may not be enough, but guaranteeing protection is of paramount importance. When an eavesdropper causes secret information to be detected within a media then the information protection system will fail. In this paper, a reversible data hiding method has been proposed using center-folding combining with Directional Pixel Value Ordering (DPVO) to solved the issue related to secure hidden communication.

The rest of the paper is organized as follows. Literature review is explored in Section 2, motivations and objectives of the paper is mentioned in Section 2.1, contribution of the paper is enlisted in Section 2.2, the introduction of Peng et al.'s [23] and Lu et al.'s [17] algorithms are displayed in Section 3. The proposed data hiding procedures are discussed with numerical illustration in Section 4. The experimental results with performances are exposed in Section 5. Analyzing results with some steganographic attacks are exhibited in Section 6. Section 7 draws the conclusions.

2 Literature review

In recent years, several reversible data hiding schemes have been reported. Among them, Different Expansion (DE) is a very significant and popular technique suggested by Tian [29] in which secret data bits are concealed by modification a pair of pixel. Alattar [1], generalized Tian's approach in 2004, by performing $(n - 1)$ bits hiding within n pixels. The general DE method used reversible integer transform to embed secret data into a digital image. In multilayered embedding applications, this approach might degrade the image quality of the stego image. In 2008, Kim et al. [8] suggested a new DE transform using location map

which is smaller than Tian's location map. Thodi and Rodríguez [28] developed Prediction Error Expansion (PEE), which is the extension of DE, where pixel's prediction error is considered to embed secret data instead of their difference, in such a way that the hiding capacity is increased twice compared with DE and reduce distortion due to much smaller value of perdition error. After that, many researchers developed, improved PEE based data embedding approach in various ways, for example using a map of the point of overflow based on payload [7], sorting and predictions [26], adaptive embedding [13], context incorporation [5], etc. Recently, Pixel Value Ordering (PVO) with Prediction Error Expansion (PEE) based information hiding approaches have been described by Li et al. [14] where minimal and maximal pixels are moderated during message embedding. The utilization of PVO in prediction error expansion helps to reduce the shifting pixels in an image. Use the second highest and second smallest pixels, the PVO method provided the estimation errors and integrates the hidden data into minimally and maximally weighted pixel resolution in a blockwise fashion. Hence, Li et al.'s RDH provides better visual quality. In order to further enhance the results, Peng et al. [23] proposed the notion of the relative location of the second peak value and the second low value with respect to highest and lowest pixel value in the frame, respectively, when evaluating the inaccuracy of estimation. Peng et al. [23] improved the PVO method presented by Li et al. [14]. They modified the Li et al.'s [14] difference amongst the first and second minimum or maximum value of a block by new difference technique as well as new histogram modification technique. Improving PVO method, Qu and Kim [25] presented Pixel-based Pixel Value Ordering (PPVO) scheme. This approaches uses pixels as foundation rather than regions where neighbour pixels are applied as reference. The minimal and maximal pixels are moderated for secret message bit embedding. An improved PVO based RDH method depended on multiple histograms is presented by Ou et al. [22]. This is a method of modification in adaptive histogram which is appropriate for histogram generation.

Lu et al. [17] suggested RDH scheme through center folding which reduces the secret symbol's value. Many of the reversible data hiding techniques used to conceal secret data into hidden symbols form. If the value of the hidden symbols is too high, then difference between the stego-pixel with the original values will be high. Lu et al. [17] brought down the value of the secret symbols by folding technique, and embedding the folded hidden symbols in dual image, by an average process to improve the visual quality of the image. Further, Lu et al. [18] designed a multilayered architecture, where the hidden data is encoded into code pairs, then embedded within the interpolated pixel in such a way that it effectively reduce the image distortion. Meikap and Jana suggested a data hiding basing on interpolation scheme with Pixel Value Ordering (PVO). They modified PVO scheme and suggested generalized PVO with Direction called Directional PVO (DPVO) with varying block size. The scheme works for three directions that is horizontal, vertical and diagonal to increase the embedding capacity. Lee et al. [11] developed a lossless privacy preservation strategy, using centralized differential expansion to mask more hidden data into finer cover image areas and later he encoded a hidden message using the pixel center point position to get the stego-pixels. In order to preserve the image quality degradation, Lou et al. [15] have suggested Reduced Expansion of Differences (RDE) process. Lou's scheme is not only reversible but also satisfies small processing costs with an embedding scheme for high capacity data. To get high payload, it utilized 2 copies of images through averaging formula. This method folds the secrets and then inserted into two stego-images. Meikap et al. [19, 20] presented RDH based PVO schemes which improved the payload. But, the challenges is to raise the payload by frequently embedding into overlapped pixels in different directions within an image block after fold the secret data and embed it into pixels. To overcome this challenges,

we have proposed an improved center folding based information hiding technique through directional PVO (DPVO) with different block size.

2.1 Motivations and objectives of the paper

For adequate data hiding technique, the motivations of the proposed work are as follows:

- (i) So far, most PVO based data hiding schemes hides at most two positions in a row of image block, that is the maximal and/or minimal position by comparing with the second maximal and/or minimal pixel. Hence, the motivation is to embed secret data bits in more than two positions in a row of the image block.
- (ii) So far, many secret data hiding methods have been reported which utilizes single images. Hence, the motivation is to investigate PVO based data hiding schemes for multiple image with good visual quality.
- (iii) PVO based scheme may be utilized to communicate secrets among sender and receiver. So far, it is not studied the secret data extraction from tampered stego images. This motivate us to investigate various tampered detection and extraction from stego images.
- (iv) So far, it has been observed that the embedding capacity (EC) of PVO or center folding based technique was limited. This motivate us to improve embedding capacity in bits through PVO or center folding based technique.
- (v) It has been observed that, data bits are embedded in the same pixel repeatedly is rare in PVO based data hiding scheme. This motivate us to embed secret data bits in same pixel repeatedly in such a way that we can easily retrieve secret data with out any fail.
- (vi) The most challenging task on data hiding scheme is to retrieve the accurate secret data from overlapped stego pixel without any distortion. This motivate us to investigate data hiding scheme on overlapped pixel with multiple time embedding with successful extraction.
- (vii) The security of the PVO based data hiding scheme is more challenging task which was not evaluated by earlier researchers. This motivate us to design a secured PVO based data hiding schemes.

The objectives of the proposed investigation are mentioned as follows:

- (i) *Embedding capacity*: Capacity of embedded data bits should be high while keeping good visual quality. The objective of the proposed scheme is to embed more than two position in a row of any image block. In addition, high embedding is possible due to dual image with image interpolation schemes.
- (ii) *Imperceptibility*: For stego image, it is necessary that anyone can't perceive in mind the presence of secret message. So, imperceptibility is required for data hiding scheme. It is possible due to dual image and image interpolation.
- (iii) *Reversibility*: After conceal the important secret message into cover image, it is responsible to extract the original secrets from the stego image as well as recover the original cover image otherwise the receiver can't get the actual secret message. To embed and extract secret from more than one image is difficult assignment. Hence, our objective is to design dual image based reversible data hiding scheme with PVO and center folding scheme with good visual quality.
- (iv) *Robustness*: The data hiding method must be maintained in such a fashion that, it should not be affected by various steganographic attacks. The proposed scheme is

designed to achieve more than one time embedding in a same pixel. It has been possible for DPVO schemes which operated three directions that is horizontal, vertical and diagonal one after another.

- (v) *Tamper detection*: A cover image may be corrupted or modified by any adversary. It should be discarded by the receiver. So, it is important to detect the presence of any tamper in stego image. So, the objective is to study tampered detection after embed secret data which is worth for medical and military image processing.
- (vi) *Enhanced security*: Security should be increased to protect from various steganographic attacks. In existing PVO based schemes, there are some security loophole which need to acquire in a certain level for practical applications. Here, data set number, size of image block and number of image block are required to extract secret data which treat as secret keys without which it is impossible to retrieve desired secret message.

2.2 Contribution of the paper

The benefactions of this paper are described below:

- (i) *Imperceptibility*: The proposed scheme achieve good visual quality with high embedding capacity. This is possible for data embedding within interpolated dual image. The PSNR is above 51 dB while $p = 2$, where p is the set of secret data bits.
- (ii) *Reversibility*: The proposed algorithm is reversible because we can successfully take out both the hidden message and the cover image from dual marked images without any distortion. This is possible for maintaining maximum and minimum pixel modification even after the data embedding the pixel order remains the same.
- (iii) *Embedding capacity*: It gives average embedding capacity above 6,00,000 bits where $p = 2$ while keeping good image quality. The data embedding is performed in two stages on dual image. In the first stage, center folding technique has been applied and in second stage DPVO has been encountered.
- (iv) *Robustness*: To examine the robustness of our proposed scheme, we evaluate RS analysis, SD, CC, SSIM, NCC, SC, NAE, BER, Q-index and several attacks like noise of salt and pepper, cropping, copy-move forgery and opaque. It has been observed that all those importance parameters of statistical evaluation offer promising results.
- (v) *Enhanced security*: The proposed method uses a set of secret bits as the value of p , the number of image block as N and size of image block as π which are treated as secret keys and shared between sender and receiver as well as the secret data bits are embedded within dual images. The hidden data bits are hard to retrieve without simultaneous two stego images with correct values of secret keys p , N , and π which are hard to guess by adversary and computationally hard by recent computing devices.

3 Preliminaries

Here, reversible data hiding with PVO of Peng et al.'s scheme [23] and center folding with dual image of Lu et al.'s scheme [17] are concisely described.

3.1 Review of Peng et al.'s IPVO [23]

In 2014, Peng et al. [23] introduced an information hiding method called Improved Pixel Value Ordering (IPVO). In this method, two extremely valued pixels (i. e. minimal and

maximal) are modified for message embedding. The 1st and 2nd smallest pixels are utilized to find out the Prediction-Error (PE) for embedding in minimum modification. The n th and $(n - 1)$ th largest pixels are used to find out the Prediction-Error (PE) for embedding in maximum modification.

3.1.1 Embedding and extraction process based on minimum-modification

The details of message embedding in the minimum modification is as follows: The image is separated into several blocks and then sorted the pixels of each block in increasing sequence. Then compute

$$ER_{\min} = x_b - x_c \text{ where } \begin{cases} b = \min(\sigma(1), \sigma(2)), \\ c = \max(\sigma(1), \sigma(2)). \end{cases} \tag{1}$$

Now, the revised minimal pixel $x_{\sigma(1)}$ is gained by

$$x' = \begin{cases} x_{\sigma(1)} - a, & \text{if } ER_{\min} = 1 \\ x_{\sigma(1)} - 1, & \text{if } ER_{\min} > 1 \\ x_{\sigma(1)} - a, & \text{if } ER_{\min} = 0 \\ x_{\sigma(1)} - 1, & \text{if } ER_{\min} < 0 \end{cases} \tag{2}$$

where the bits $a \in \{0, 1\}$ are inserted with pixel. The numerical instance of embedding is displayed in Fig. 1. The revised value of X is (y_1, \dots, y_n) , where $y_{\sigma(1)} = x'$ and $y_i = x_i$ for every $i \neq \sigma(1)$. The mapping σ keeps unaltered in decoder side. The extraction of bits and restoration of image processes are gained from the sorted pixels (y_1, \dots, y_n) . Calculate the value of $ER'_{\min} = y_b - y_c$, where (b, c) is described in (1).

If $ER'_{\min} \leq 0$, then $y_b \leq y_c$. As $\sigma(1) < \sigma(2)$ then $b = \sigma(1)$ with $c = \sigma(2)$:

- If $ER'_{\min} \in \{-1, 0\}$, it ensures the presence of concealed data bit $a = -ER'_{\min}$. The minimal pixel is $x_{\sigma(1)} = y_b + a$;
- If $ER'_{\min} < -1$, it ensures the absence of concealed message. The minimal pixel is $x_{\sigma(1)} = y_b + 1$.

If $ER'_{\min} > 0$, then $y_b > y_c$. So, $b = \sigma(2)$, $c = \sigma(1)$ and here, $\sigma(2) < \sigma(1)$:

- If $ER'_{\min} \in \{2, 1\}$, it ensures the presence of concealed message $a = ER'_{\min} - 1$ and the minimal pixel is $x_{\sigma(1)} = y_c + a$;
- If $ER'_{\min} > 2$, then absent hidden message and the minimal pixel is $x_{\sigma(1)} = y_c + 1$.

The insertion and removal process in maximum modification is discussed in [23].

The main limitation of the Peng et al.'s scheme [23] is that it can only insert at most two data bits in a row of a image block. To increase the embedding capacity, we have developed

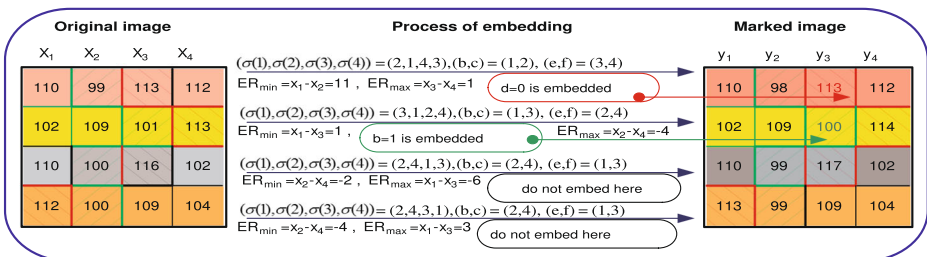


Fig. 1 Numerical example of IPVO scheme proposed by Peng et al. [23] for data embedding

the proposed scheme which embed more than two data bits depending on the image block size.

3.2 Review of Lu et al.'s scheme [17]

Lu et al. [17] described a RDH method exploiting a center folding technique incorporated with a dual image. This scheme squeezes the hidden symbol through applying the center folding method before inserting it within dual marked images. Each bits of l from original hidden information is assembled as a set and turned into a secret decimal symbol h . To keep away from distortion, the range of secret decimal symbol is changed from $M = \{0, 1, \dots, 2^l - 1\}$ to $M' = \{-2^{l-1}, -2^{l-1} + 1, \dots, 0, \dots, 2^{l-1} - 2, 2^{l-1} - 1\}$ by using folding method. h'_s can be computed by using the equation discussed in [17] where 2^{l-1} is median values. When $h < 2^{l-1}$, the value of h' will be smaller than 0 and was indicated as a negative sign. When $h = 2^{l-1}$, the value of h' was indicated as 0. When $h > 2^{l-1}$, the hidden value h' was indicated as a positive sign and it will be greater than 0.

These folded hidden symbol (h') was embedded into two marked images through averaging technique. The h' was equally divided into two parts h'_1 and h'_2 . These are inserted into real pixel $x_{i,j}$ to construct dual marked images $x'_{i,j}$ and $x''_{i,j}$. In Lu's scheme [17], the value range of the secret data is divided into two parts. One part consists of the negative folded values that is less than 0, and the other part consists of the folded positive values which is greater than 0. The largest value in each part is still very large, which can cause huge distortion. To remove the huge distortion one may use re-encoding by second phase embedding which has been designed in the proposed scheme. Here we have divided the data embedding stage into two phases: In first phases, we try to embed secret data through center folding with a proper data set number and then in second stage DPVO has been applied for embedding in overlapped pixel with repeated time.

4 Proposed method

This section represents the secret message inserting and taking out process through center folding and Directional PVO (DPVO) techniques where secrets are embedded to form stego images. The proposed method made up of two stages: (1) Embedding Procedure and (2) Extraction Procedure.

4.1 Notations

The following notations are used in this paper:

Notations	Descriptions
CI	Cover Image
IMI	Interpolated Image
CI_1	Interpolated marked image –1 after first embedding
CI_2	Interpolated marked image –2 after first embedding
CI'_1	Interpolated marked image –1 after second embedding
CI'_2	Interpolated marked image –2 after second embedding
w	Width of Cover Image
h	Hight of Cover Image
D_{pix}	Interpolated Pixel
p	A set of secret bits
N	Number of image block

π	Size of image block
d_s	Hidden symbol in decimal form
Q	Hidden decimal symbol's range
Q'	Modified hidden decimal symbol's range
d'_s	Folded hidden symbol
d'_{s1}	Folded hidden symbol for image –1
d'_{s2}	Folded hidden symbol for image –2
$pixel_{i,j}$	Pixel of cover image
$pixel'_{i,j}$	Stego pixel of image –1
$pixel''_{i,j}$	Stego pixel of image –2
α	A parameter for maintaining pixel order
d_{min_f}	Pixel differences between minimum pixels
d_{max_f}	Pixel differences between maximum pixels
d'_{min_f}	Modified pixel differences between minimum pixels
d'_{max_f}	Modified pixel differences between maximum pixels
D	Hidden Message
EC	Embedding Capacity
EC ₁	Embedding Capacity in image –1
EC ₂	Embedding Capacity in image –2
SC	Structure Content
NAE	Normalized Absolute Error
BER	Bit Error Rate
CC	Correlation Coefficient
SD	Standard Deviation
SSIM	Structural SIMilarity
NCC	Normalized Cross-Correlation
PSNR	Peak Signal-to-Noise Ratio
PSNR ₁	Peak Signal-to-Noise Ratio for image –1
PSNR ₂	Peak Signal-to-Noise Ratio for image –2

4.2 Embedding procedure

The secret data are embedded in two phases. In the first phase, center folding scheme has been used and in second phase Directional PVO has been utilized. The cover image represents to $CI = \{pixel_{1,1}, pixel_{1,2}, pixel_{1,1}, \dots, pixel_{w,h}\}$, where w and h denoted image width and height, respectively. We expand this image through the interpolation method shown in Fig. 2 by averaging the neighbor pixels using the following (3).

$$\begin{cases} D_{pix1} = (pixel_1 + pixel_2)/2 \\ D_{pix2} = (pixel_1 + pixel_3)/2 \\ D_{pix3} = (pixel_1 + pixel_2 + pixel_3 + pixel_4)/4 \\ D_{pix4} = (pixel_2 + pixel_4)/2 \\ D_{pix5} = (pixel_3 + pixel_4)/2 \end{cases} \tag{3}$$

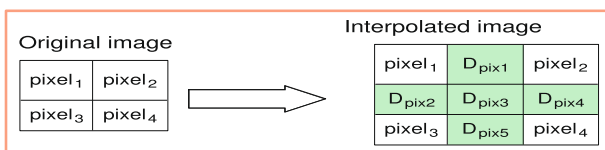


Fig. 2 Construction of interpolated pixels with green color that represents interpolated row and column

If image size is $(v \times v)$ then the interpolated image will be $(v + (v - 1)) \times (v + (v - 1))$, where interpolated row/column is $(v - 1)$. One is added to both row and column to form an even number of row and column and copying the last row and column value. The hidden information accepts each bit of p as a set and modifies into a hidden symbol d_s in decimal form. To keep away from image distortion, compress the hidden symbol d_s and modify the hidden decimal symbol's range from $Q = \{0, 1, \dots, 2^p - 1\}$ to $Q' = \{-2^{p-1}, -2^{p-1} + 1, \dots, -1, 0, 1, \dots, 2^{p-1} - 2, 2^{p-1} - 1\}$. The folded hidden symbol d'_s can be calculated as follows:

$$d'_s = d_s - 2^{p-1} \tag{4}$$

where 2^{p-1} represents intermediary values. By using (4), d_s is changed into folded d'_s .

- If $d_s = 2^{p-1}$, then the value of d'_s is indicated as 0.
- If $d_s > 2^{p-1}$, then the value of d'_s is indicated as a positive sign and it will be greater than 0.
- If $d_s < 2^{p-1}$, then the value of d'_s is indicated as a negative sign and it will be smaller than 0.

The secret data embedding process can be accomplished by

$$\begin{cases} d'_{s1} = \lfloor \frac{d'_s}{2} \rfloor, \\ d'_{s2} = \lceil \frac{d'_s}{2} \rceil. \end{cases} \tag{5}$$

where d'_{s1} and d'_{s2} are derived from d'_s . These values (d'_{s1}, d'_{s2}) are inserted into pixel value $pixel_{i,j}$ to construct dual interpolated marked (stego) pixels as $pixel'_{i,j}$ and $pixel''_{i,j}$. Thus, two interpolated marked images are constructed using the equations given below.

$$\begin{cases} pixel'_{i,j} = pixel_{i,j} + d'_{s1}, \\ pixel''_{i,j} = pixel_{i,j} - d'_{s2}, \end{cases} \tag{6}$$

The secret message bits hide into image pixels that belong in between 2^{p-1} and $256 - 2^{p-1}$.

Now, first phase message embedding is completed successfully. Then we have started the second phase of message embedding through the Directional PVO (DPVO) method. In this second phase, dual interpolated marked (stego) images are breaking into non-overlapping blocks where all pixels are arranged in ascending ordered. There is a possibility to change the order of pixel due to modification in more than two pixel values in a row, which may create problem during data extraction. To maintain the rank of sorting order, we have proposed a new parameter α . We subtract a new parameter α from 1st, 2nd and so on smallest pixels and add the α to 1st, 2nd and so on largest pixel values. The value of α is dependent on the block size. If the block size is increases the maximum value of α will be increases. After each inclusion or exclusion with the pixel, the α decreases by one until it reaches to zero value. If the block size is v , then interpolated block is $(2v - 1)$ where $(v - 2)$ is the maximal value and 0 is the minimum value of α . In this context, we have proposed Lemma-1 which has been used during data embedding.

Lemma 1 *If the size of block is $(2v - 1)$ and sorted pixels in increasing manner is $(pixel_1, pixel_2, pixel_3, \dots, pixel_{2v-3}, pixel_{2v-2}, pixel_{2v-1})$, then the changed pixel value is $(pixel_1 - \alpha_{(v-2)}, pixel_2 - \alpha_{((v-2)-1)}, pixel_3 - \alpha_{((v-2)-2)}, \dots, pixel_{v-1} - \alpha_{((v-2)-(v-2))}, pixel_v, pixel_{v+1} + \alpha_{((v-2)-(v-2))}, \dots, pixel_{2v-3} + \alpha_{((v-2)-2)}, pixel_{2v-2} +$*

$\alpha_{((v-2)-1)}$, $pixel_{2v-1} + \alpha_{(v-2)}$, where, the value of $\alpha_{((v-2)-(v-2))} = ((v-2) - (v-2))$. The maximal and minimal value are $(v-2)$ and 0 respectively.

Example 1 Let block size is (2×2) , so size of interpolated block is (3×3) . The α 's maximal value is 0 (as $(v-2) = 0$). It is added and subtracted to and from the pixel of maximal and minimal respectively. This activity still going on until $(\alpha) = 0$ which is denoted in Fig. 3.

Hidden data are inserted into the pixels of image block in distinct way (i.e. Horizontal, Vertical and Diagonal). The procedure is defined below:

4.2.1 Embedding in minimum-modification

In every row of a block X , assume that n pixels ($pixel_1, \dots, pixel_n$) are sorted in rising up order to get $(pixel_{\sigma(1)}, \dots, pixel_{\sigma(n)})$. We compute

$$d_{\min_f} = pixel_b - pixel_c, \text{ where } \begin{cases} c = \max(\sigma((2) + f), \sigma((1) + f)), \\ b = \min(\sigma((2) + f), \sigma((1) + f)), \\ f = (0, 1, 2, 3, \dots, \text{fix}(n/2) - 1). \end{cases} \quad (7)$$

To round the data toward zero, we use $\text{fix}()$.

- When $\sigma((2) + f) < \sigma((1) + f)$, the value of $b = \sigma((2) + f)$ with $c = \sigma((1) + f)$, where $d_{\min_f} > 0$.
- When $\sigma((1) + f) < \sigma((2) + f)$, the value of $b = \sigma((1) + f)$ with $c = \sigma((2) + f)$, where $d_{\min_f} \leq 0$.

The d_{\min_f} is changed using following formula

$$d'_{\min_f} = \begin{cases} d_{\min_f} - 1, & \text{if } d_{\min_f} < 0 \\ d_{\min_f} - D, & \text{if } d_{\min_f} = 0 \\ d_{\min_f} + D, & \text{if } d_{\min_f} = 1 \\ d_{\min_f} + 1, & \text{if } d_{\min_f} > 1 \end{cases} \quad (8)$$

where $D \in \{0, 1\}$ are inserted into pixel. In each operation, α is changed. The modified minimum value is derived by

$$pixel' = pixel_{\sigma((2)+f)} - \alpha - |d'_{\min_f}| = \begin{cases} (pixel_{\sigma((1)+f)} - \alpha) - 1, & \text{if } d_{\min_f} < 0 \\ (pixel_{\sigma((1)+f)} - \alpha) - D, & \text{if } d_{\min_f} = 1 \\ (pixel_{\sigma((1)+f)} - \alpha) - D, & \text{if } d_{\min_f} = 0 \\ (pixel_{\sigma((1)+f)} - \alpha) - 1, & \text{if } d_{\min_f} > 1 \end{cases} \quad (9)$$

Assume the changed value of $X=(cpixel_1, cpixel_2, \dots, cpixel_n)$ in each row, where $cpixel_{\sigma((1)+f)} = pixel'$ and $cpixel_i = pixel_i$ for all $i \neq \sigma((1)+f)$ and the changed value

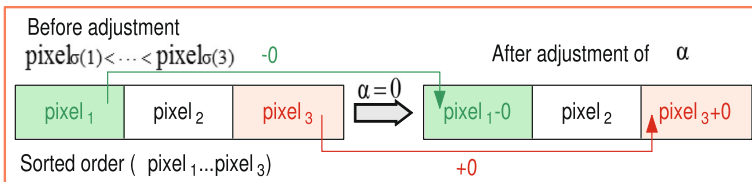


Fig. 3 Block diagram for data adjustment using Lemma 1 during data embedding

of $X=(CR_1, CR_2, \dots, CR_n)$ in each column, where $CR_{\sigma((1)+f)} = cr'$ and $CR_i = cr_i$ for all $i \neq \sigma((1) + f)$.

4.2.2 Embedding in maximum-modification

Algorithm 1 Embedding procedure.

Input: The cover image ($CI_{w \times h}$). p bits of hidden message(D) makes a group. The α is derived by Lemma 1.

Output: The interpolated image: $IMI_{(2w-1) \times (2h-1)}$, the dual interpolated marked images: $CI'_{1(2w-1) \times (2h-1)}, CI'_{2(2w-1) \times (2h-1)}$, number of block (N), the block size (π).

Step-1: Take cover image and make interpolated image $IMI_{(2w-1) \times (2h-1)}$ using (3);

Step-2: Take the value of p and converted into decimal based hidden symbol d_s ;

Step-3: Compute the folded hidden symbol d'_s using (4);

Step-4: Calculate the value of d'_{s1} and d'_{s2} using (5) which are inserted into pixel value $pixel_{i,j}$ to make dual marked pixel $pixel'_{i,j}$ and $pixel''_{i,j}$ using (6) and created dual interpolated marked images $CI_{1(2w-1) \times (2h-1)}$ and $CI_{2(2w-1) \times (2h-1)}$;

Step-5: Making N blocks whose size is π and each containing n pixels from interpolated marked images by dividing method;

Step-6:

```

for  $CI_{1(2w-1) \times (2h-1)}$  to  $CI_{2(2w-1) \times (2h-1)}$  do
  for  $block_1$  to  $block_N$  do
    Pixels are sorted in ascending order from each row;
    for  $row_1$  to  $row_\pi$  do
      for pair of 1st to  $(\pi/2)$ th minimal and maximal pixel do
        |  $dataembedding()$ ;
      end
    end
    Pixels are sorted in ascending order from each column;
    for  $col_1$  to  $col_\pi$  do
      for pair of 1st to  $(\pi/2)$ th minimal and maximal pixel do
        |  $dataembedding()$ ;
      end
    end
    Pixels are sorted in ascending order from first diagonal;
    for  $diagonal1$  do
      for pair of 1st to  $(\pi/2)$ th minimal and maximal pixel do
        |  $dataembedding()$ ;
      end
    end
    Pixels are sorted in ascending order from second diagonal;
    for  $diagonal2$  do
      for pair of 1st to  $(\pi/2)$ th minimal and maximal pixel do
        |  $dataembedding()$ ;
      end
    end
  end
end
    
```

Step-7: Construct interpolated marked images $CI'_{1(2w-1) \times (2h-1)}$ and $CI'_{2(2w-1) \times (2h-1)}$ with N and π ;

Step-8: Stop;

procedure: $dataembedding()$

Execute $d_{min_f} = pixel_b - pixel_c$ and $d_{max_f} = pixel_d - pixel_e$, where (b, c, d, e, f) are depicted in (7) and (10);

The changed pixel $pixel'$ from minimum modification is derived from (9);

The changed pixel $pixel'$ from maximum modification is derived from (12);

end procedure

The data inserting into pixels for Maximum-Modification is discussed below: Calculate,

$$d_{\max_f} = pixel_d - pixel_e \text{ where } \begin{cases} e = \max(\sigma((n - 1) - f), \sigma((n) - f)), \\ d = \min(\sigma((n - 1) - f), \sigma((n) - f)), \\ f = (0, 1, 2, 3, \dots, \text{fix}(n/2) - 1). \end{cases} \quad (10)$$

- When $\sigma((n) - f) < \sigma((n - 1) - f)$, then value of $c = \sigma((n) - f)$ with $e = \sigma((n - 1) - f)$. Here, $d_{\max_f} > 0$.
- When $\sigma((n - 1) - f) < \sigma((n) - f)$, then value of $c = \sigma((n - 1) - f)$ with $e = \sigma((n) - f)$. Here, $d_{\max_f} \leq 0$.

The d_{\max_f} is changed to d'_{\max_f} by following formula

$$d'_{\max_f} = \begin{cases} d_{\max_f} - 1, & \text{if } d_{\max_f} < 0 \\ d_{\max_f} + D, & \text{if } d_{\max_f} = 1 \\ d_{\max_f} - D, & \text{if } d_{\max_f} = 0 \\ d_{\max_f} + 1, & \text{if } d_{\max_f} > 1 \end{cases} \quad (11)$$

where $D \in \{0, 1\}$ are inserting bits. The changed maximum pixel value $pixel'$ is derived by

$$pixel' = pixel_{\sigma((n-1)-f)} + \alpha + |d'_{\max_f}| = \begin{cases} (pixel_{\sigma((n)-f)} + \alpha) + 1, & \text{if } d_{\max_f} < 0 \\ (pixel_{\sigma((n)-f)} + \alpha) + D, & \text{if } d_{\max_f} = 1 \\ (pixel_{\sigma((n)-f)} + \alpha) + D, & \text{if } d_{\max_f} = 0 \\ (pixel_{\sigma((n)-f)} + \alpha) + 1, & \text{if } d_{\max_f} > 1 \end{cases} \quad (12)$$

Assume the changed value of $X=(cpixel_1, cpixel_2, \dots, cpixel_n)$ in each row, where $cpixel_{\sigma((n)-f)} = pixel'$ and $cpixel_i = pixel_i$ for all $i \neq \sigma((n)-f)$ and the changed value of $X=(CR_1, CR_2, \dots, CR_n)$ in each column, where $CR_{\sigma((n)-f)} = cr'$ and $CR_i = cr_i$ for all $i \neq \sigma((n) - f)$. After embedding horizontally, then the above minimum and maximum process follow in vertical and diagonals ways to form the final stego images. In the second phase of the data embedding procedure, the maximum and minimum pixels have been changed in three distinct directions one after another. The overall hidden message embedding process is displayed in Fig. 4. Figure 4a depicts the original cover image and after interpolation it is converted to interpolated cover image as (b). After that, this interpolated image used for data embedding through the center folding method which produces dual marked images as shown in (c) and (d). These two images as (c) and (d) are used for second phase data embedding using DPVO method which produces final dual stego images as (e) and (f) respectively. The suggested secrets embedding algorithm is following in Algorithm 1. We have taken one cover image and generate interpolated image following step-1. In step-2, we have choose p as a set of secret data and then converted it in equivalent decimal value. Construct folded hidden symbols and embedded within dual images is described in step-3 and step-4. First stage data embedding has been completed using center folding scheme and dual intermediate stego image is generated. Now, in step-5, image is partitioned into non-overlapping N blocks and step-6 describe data hiding procedure using DPVO.

4.3 Numerical demonstration of embedding activity

We present a numerical example to explain the suggested embedding activity. For input, we use a (2×2) real cover image. Then, construct an interpolated cover image of size (3×3)

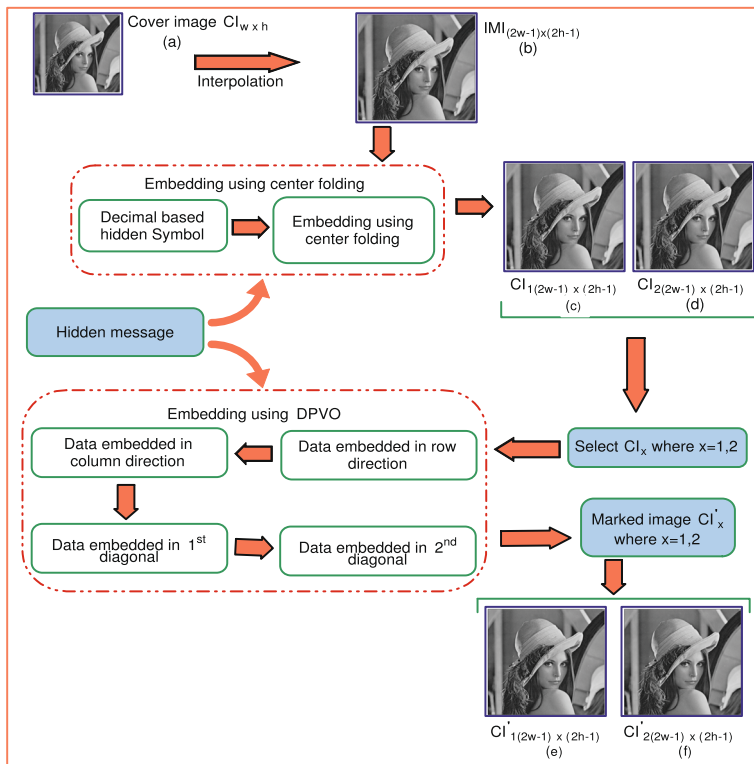


Fig. 4 Overall data embedding process in proposed improved center folding based DPVO scheme

by (3) as IMI from the cover image. Using the center folding method, dual marked images CI_1 and CI_2 of size (3×3) are constructed.

Let the hidden message is ‘00100001000110001110010110001000’ which we like to hide. We take here, the value of p is 3. Using (4), (5) and (6), the dual images are constructed shown in Fig. 5.

Then, the second phase of message embedding takes place into the dual marked images. This process takes place through three distinct directions (i. e. row, column, first and second diagonal) one after another. Finally, two marked images CI'_1 and CI'_2 are made as shown in Fig. 5.

4.4 Extraction procedure

In data extraction, reverse order direction will occur. This process establishes from the diagonal direction of the interpolated marked image. α is added and subtracted to and from the minimal and maximal pixel of 1st, 2nd and so on respectively, which is dependent on the Lemma 2:

Lemma 2 Assume, the size of image block is $(2v - 1)$ and ordered pixels in increasing manner is $(pixel'_1, pixel'_2, pixel'_3, \dots, pixel'_{2v-3}, pixel'_{2v-2}, pixel'_{2v-1})$, then the changed

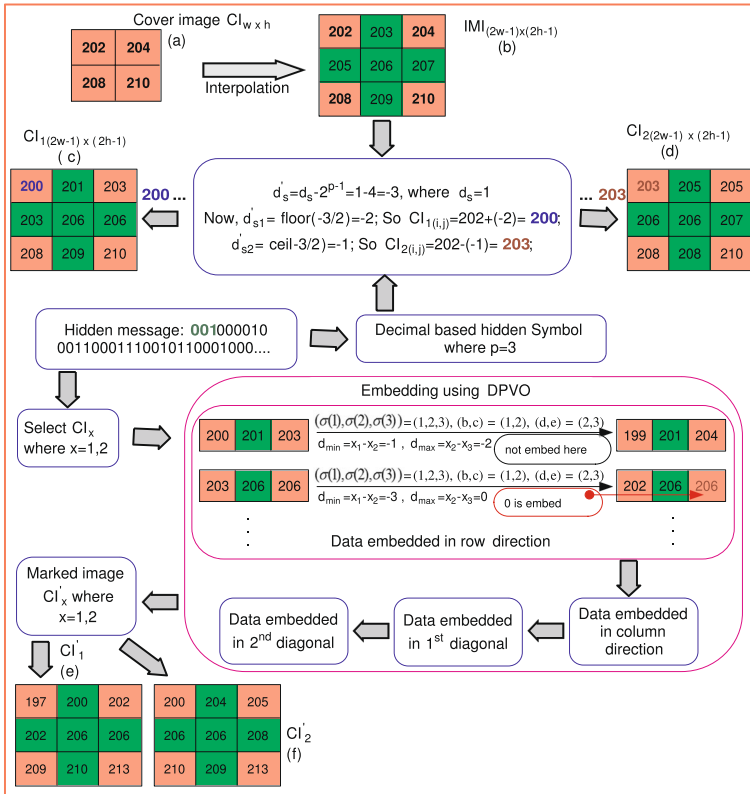


Fig. 5 Numerical example of two phase data embedding

pixel value is $(pixel'_1 + \alpha_{(v-2)}, pixel'_2 + \alpha_{((v-2)-1)}, pixel'_3 + \alpha_{((v-2)-2)}, \dots, pixel'_{v-1} + \alpha_{((v-2)-(v-2))}, pixel'_v, pixel'_{v+1} - \alpha_{((v-2)-(v-2))}, \dots, pixel'_{2v-3} - \alpha_{((v-2)-2)}, pixel'_{2v-2} - \alpha_{((v-2)-1)}, pixel'_{2v-1} - \alpha_{(v-2)})$. Whenever, the $\alpha_{((v-2)-(v-2))} = ((v-2) - (v-2))$. The maximal and minimal values are $(m-2)$ and 0 respectively.

Example 2 Let block size is (2×2) , so size of interpolated block is (3×3) . The α 's maximal value is $(v-2) = 0$, that is added and subtracted to and from the pixel of minimal and maximal respectively. This activity still going on until $(\alpha) = 0$, which is denoted in Fig. 6.

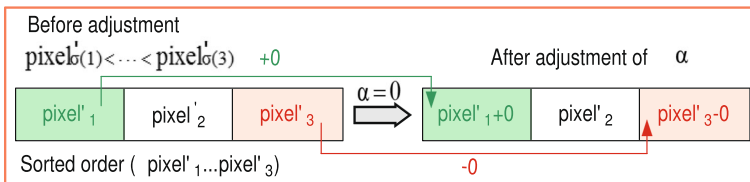


Fig. 6 Block diagram for data adjustment using Lemma 2 during data extraction

Algorithm 2 Extraction procedure.

Input: The dual interpolated images $CI'_{1(2w-1)\times(2h-1)}$ and $CI'_{2(2w-1)\times(2h-1)}$.
Output: The recover data D , the real cover image ($CI_{w\times h}$).
for $CI'_{1(2w-1)\times(2h-1)}$ **to** $CI'_{2(2w-1)\times(2h-1)}$ **do**
 Step-1: Split interpolated marked image into N blocks with size of π . Here, N , p and π are undisclosed key. Each block holds n pixels;
 Step-2:
 for $block_N$ **to** $block_1$ **do**
 Pixels are sorted in ascending order from second diagonal;
 for $diagonal2$ **do**
 for pair of $(\pi/2)$ th to 1st maximal and minimal pixel **do**
 | $datarestoring()$;
 end
 end
 Pixels are sorted in ascending order from first diagonal;
 for $diagonal1$ **do**
 for pair of $(\pi/2)$ th to 1st maximal and minimal pixel **do**
 | $datarestoring()$;
 end
 end
 Pixels are sorted in ascending order from each column;
 for col_π **to** col_1 **do**
 for pair of $(\pi/2)$ th to 1st maximal and minimal pixel **do**
 | $datarestoring()$;
 end
 end
 Pixels are sorted in ascending order from each row;
 for row_π **to** row_1 **do**
 for pair of $(\pi/2)$ th to 1st maximal and minimal pixel **do**
 | $datarestoring()$;
 end
 end
 end
 return D ;
end
Step-3: Recover dual interpolated images CI_1 and CI_2 and store hidden message;
Step-4: Calculate the folded hidden symbol d'_s from (13);
Step-5: Compute the decimal based hidden symbol d_s through the (14) and derive the hidden message;
Step-6: Construct real pixel values from $CI_{1(2w-1)\times(2h-1)}$ and $CI_{2(2w-1)\times(2h-1)}$ by (15);
Step-7: Remove the interpolate rows and columns from interpolated image $IMI_{(2w-1)\times(2h-1)}$. Finally reconstruct the real cover image $CI_{w\times h}$;
Step-8: Stop;

procedure: $datarestoring()$
Determine $d'_{max_f} = cpixel_d - cpixel_e$ and $d'_{min_f} = cpixel_b - cpixel_c$, where (b, c, d, e, f) are discussed in (10) and (7);
The maximal pixels are regain through the extraction in maximum modification;
The minimal pixels are regain through the extraction in minimum modification;
end procedure

4.4.1 Extraction in minimum-modification

Hidden data extraction and rebuild of image are carried out through minimum pixel modification. Assume the changed value is $(cpixel_1, cpixel_2, \dots, cpixel_n)$. The mapping σ lasts unchanged. We calculate $d'_{min_f} = cpxel_b - cpxel_c$, where (b, c, f) are discussed in (7).

- When $d'_{\min_f} \leq 0$, then $cpixel_b \leq cpixel_c$. Here, $b = \sigma((1) + f)$, $c = \sigma((2) + f)$ with $\sigma((1) + f) < \sigma((2) + f)$:
 - When $d'_{\min_f} \in \{-1, 0\}$, it ensures the presence of concealed data bit $D = -d'_{\min_f}$. Restored minimal pixel is $pixel_{\sigma((1)+f)} = (cpixel_b + \alpha) + D$;
 - When $d'_{\min_f} < -1$, then, no hidden message present and restored the minimal pixel is $pixel_{\sigma((1)+f)} = (cpixel_b + \alpha) + 1$.
- When $d'_{\min_f} > 0$, then $cpixel_b > cpixel_c$. Here, $c = \sigma((1) + f)$, $b = \sigma((2) + f)$ with $\sigma((1) + f) > \sigma((2) + f)$:
 - When $d'_{\min_f} \in \{2, 1\}$, it ensures the presence of concealed data bit $D = d'_{\min_f} - 1$. Recovered minimal pixel is $pixel_{\sigma((1)+f)} = (cpixel_c + \alpha) + D$;
 - if $d'_{\min_f} > 2$, then, no hidden message present and the original minimal pixel is $pixel_{\sigma((1)+f)} = (cpixel_c + \alpha) + 1$.

4.4.2 Extraction in maximum-modification

Assume the changed values are $(cpixel_1, cpixel_2, \dots, cpixel_n)$. The mapping σ lasts unchanged. We calculate $d'_{\max_f} = cpixel_d - cpixel_e$ where (d, e, f) is discussed in (10).

- When $d'_{\max_f} \leq 0$, then $cpixel_d \leq cpixel_e$. Here, $\sigma((n - 1) - f) < \sigma((n) - f)$ with $d = \sigma((n - 1) - f)$, $e = \sigma((n) - f)$:
 - When $d'_{\max_f} \in \{-1, 0\}$, it ensures the presence of concealed data bit $D = -d'_{\max_f}$. Recovered maximum pixel $pixel_{\sigma((n)-f)} = (cpixel_e - \alpha) - D$;
 - When $d'_{\max_f} < -1$, then, no hidden message present and restored the maximum pixel is $pixel_{\sigma((n)-f)} = (cpixel_e - \alpha) - 1$.
- When $d'_{\max_f} > 0$, then, $cpixel_d > cpixel_e$. Now, $\sigma((n - 1) - f) > \sigma((n) - f)$ with $d = \sigma((n) - f)$, $e = \sigma((n - 1) - f)$:
 - When $d'_{\max_f} \in \{2, 1\}$, it ensures the presence of concealed data bit $D = d'_{\max_f} - 1$. The original pixel(maximum) is $pixel_{\sigma((n)-f)} = (cpixel_d - \alpha) - D$;
 - When $d'_{\max_f} > 2$, then, no hidden data present and recovered the maximum pixel $pixel_{\sigma((n)-f)} = (cpixel_d - \alpha) - 1$.

This extraction procedure begins from 2nd diagonal of the marked block. After that, 1st diagonal line, vertical line, and horizontal line directions are utilized to take out hidden data from both interpolated marked images. Now, we recover the previous-valued pixel from both stego pixel $pixel'_{i,j}$ and $pixel''_{i,j}$ of CI'_1 and CI'_2 marked images respectively by following formulas

$$d'_s = pixel'_{i,j} - pixel''_{i,j} \tag{13}$$

$$d_s = d'_s + 2^{p-1} \tag{14}$$

where p is a number of hidden bits makes a group. The real pixel $pixel_{i,j}$ is rebuild as follows

$$pixel_{i,j} = \left\lfloor \frac{pixel'_{i,j} + pixel''_{i,j}}{2} \right\rfloor \tag{15}$$

We can restore the real cover image $CI_{w \times h}$ from $IMI_{(2w-1) \times (2h-1)}$ interpolated image by eliminating all interpolated rows and columns. The message extraction and image rebuild are explored in Fig. 7. In Fig. 7a and b depicts the stego images for data extraction process. These (a) and (b) images are utilized for first data extraction using DPVO method which produces (c) and (d) images respectively. After that, (c) and (d) images goes through second extraction process using the center folding method which produces the interpolated image. The original cover image shown in (e) that is constructed through elimination of interpolate row and column from the interpolated image. The extraction algorithm is explored in Algorithm 2. Here, at first dual stego images are partitioned into non-overlapping N blocks and DPVO has been applied which are enlisted in step-2. Within step-3, first phase of hidden data extraction has been done and intermediate dual stego images are recovered. Step-4 to step-6 describe second phase data extraction through center folding technique and finally original cover image is reconstructed as per step-7.

4.5 Numerical demonstration of extraction activity

We consider a numerical illustration to explain the extraction activity. For an input, we use two interpolated marked (stego) images $CI'_1_{(2w-1) \times (2h-1)}$ and $CI'_2_{(2w-1) \times (2h-1)}$ shown in Fig. 8.

The first stage of extraction process passes through the second and first diagonal, column and row direction one after other of each interpolated stego images. Collect all the pixels and message from the stego images. This process makes $CI_1_{(2w-1) \times (2h-1)}$ and $CI_2_{(2w-1) \times (2h-1)}$ marked images shown in Fig. 8. Then, using (13), (14) and (15), again restore the hidden message and real interpolated cover image $IMI_{(2w-1) \times (2h-1)}$

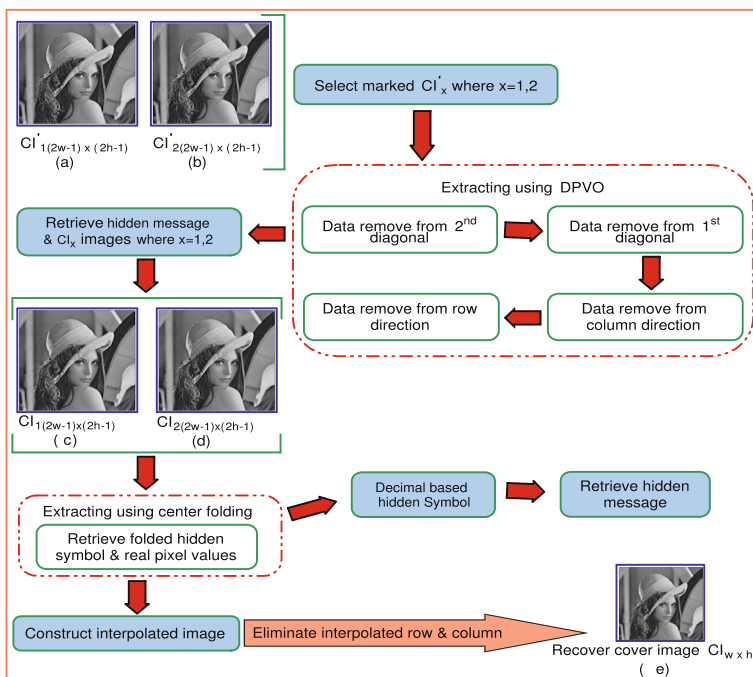


Fig. 7 Overall data extracting process in proposed improved center folding based DPVO scheme

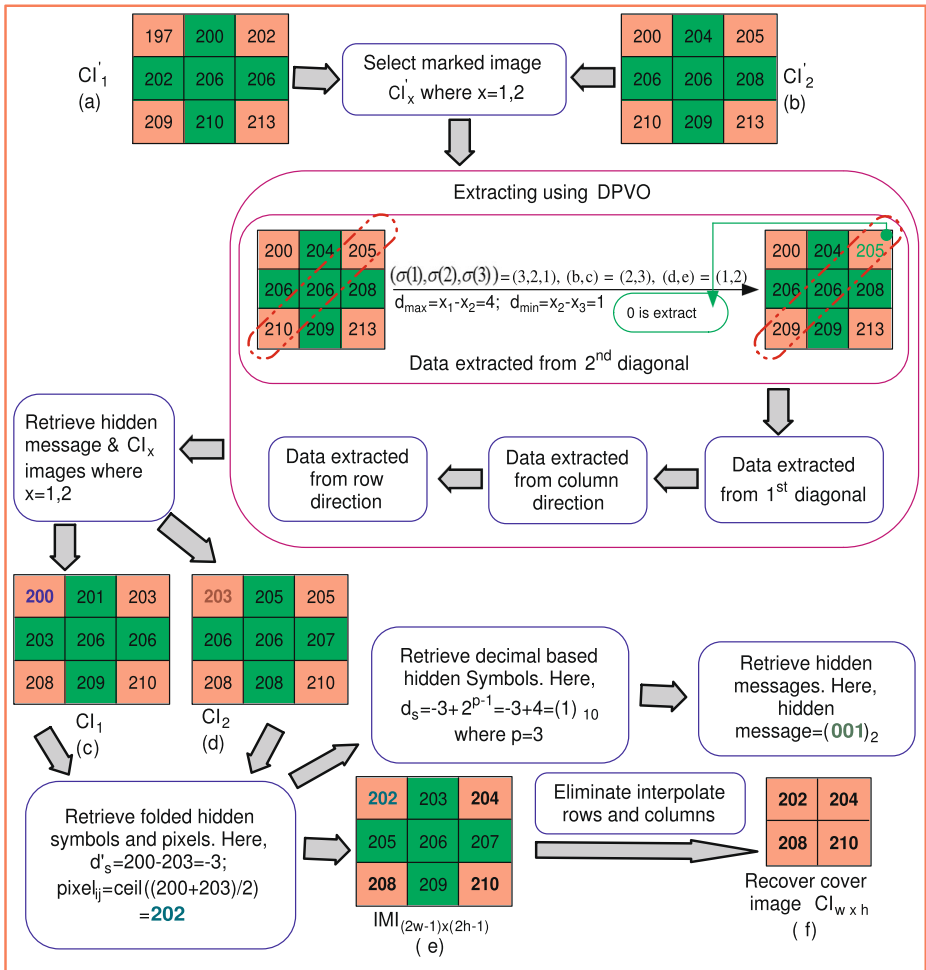


Fig. 8 Numerical example of data extraction

from dual marked images $Cl_1_{(2w-1) \times (2h-1)}$ and $Cl_2_{(2w-1) \times (2h-1)}$. Finally, the hidden message ‘00100001000110001110010110001000’ is restored as explored in Fig. 8.

5 Experiment results and comparisons

This paper used different standard gray-scale images from distinct databases. Only 10 images were used from USC-SIPI [31], 24 images were used from Kodak [9], 20 images were used from the Berkeley Segmentation Dataset and Benchmark [30] and 20 images were used from the National Library of Medicine [27] to examine the method. Only 10 standard images from each database are shown in Fig. 9. The image size of our scheme is (256×256) . Distinct datasets are utilized for the best study of the efficacy of the submitted method. Method of message embedding and removal techniques are evaluated through MATLAB R2014a (8.3.0.532). The payload with image quality is measured by PSNR (dB)

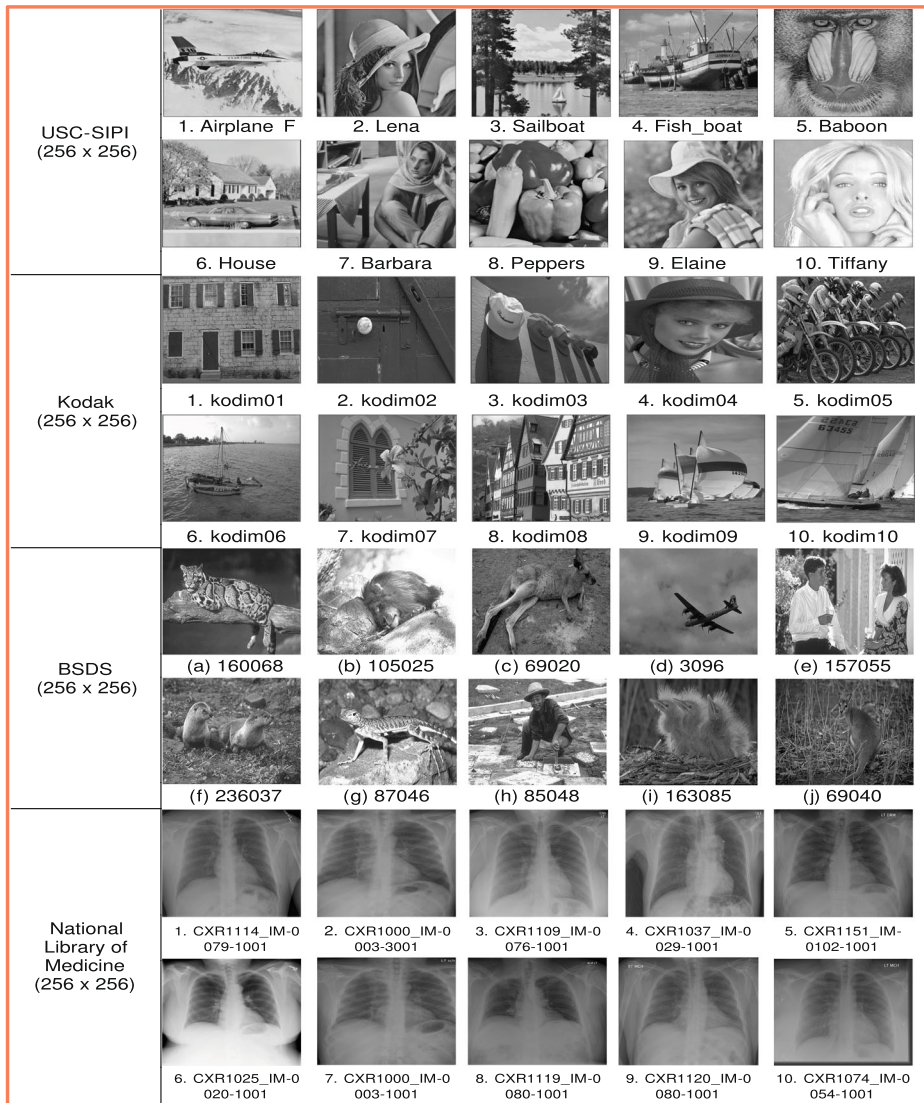


Fig. 9 The standard grayscale images from four distinct sets are used as message embedding and extracting procedure

to determine the difference between real and marked images. The PSNR is derived via mean squared error (MSE). The MSE is obtained by the following equation

$$MSE = \frac{1}{w \times h} \times \sum_{i=1}^w \sum_{j=1}^h (c'_{i,j} - c_{i,j})^2 \tag{16}$$

The PSNR (dB) is obtained by

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \tag{17}$$

We test our suggested method through structure content (SC) technique. This method evaluates the structural similarity of the cover and stego images. It is obtained by the following equation

$$SC = \frac{\sum_{i=1}^w \sum_{j=1}^h (c'_{i,j})^2}{\sum_{i=1}^w \sum_{j=1}^h (c_{i,j})^2} \quad (18)$$

The value of SC is nearer to 1, means the cover and stego images are similar. A larger value of SC presents poor quality of image. Normalized absolute error (NAE) checks the distance of stego from cover image. It is computed by

$$NAE = \frac{\sum_{i=1}^w \sum_{j=1}^h (|c_{i,j} - c'_{i,j}|)}{\sum_{i=1}^w \sum_{j=1}^h (c_{i,j})} \quad (19)$$

We also evaluate the proposed scheme by evaluating Q-index(Q) and bit error rate (BER). BER finds the differences amongst original secrets and recovered secrets. It is obtained by

$$BER = \frac{1}{W \times H} \left[\sum_{i=1}^W \sum_{j=1}^H S_{i,j} \oplus S'_{i,j} \right] \quad (20)$$

where the size of secret image is $S_{(W \times H)}$ and after recover from stego image, it is $S'_{(W \times H)}$.

After hiding 6,29,955 bits, we achieve the quality measured by PSNR of the Lena image is 52.01 dB given in Table 1. To evaluate number of hidden message bits inserted into stego image, we calculate Bit per pixel (BPP). The BPP obtained by

$$BPP = \frac{EC}{2 \times w \times h} \quad (21)$$

where embedding capacity is EC. The increasing of BPP shows the increasing ability of data hiding shown in Table 1.

The suggested method has been applied on ten (10) USC-SIPI gray-scale images through SC, NAE, Q-index, BER techniques. Table 2 displays several measurements of the suggested method. It shows the SC and NAE value are 0.9859 and 0.00762 respectively, while $p = 2$. When $p = 3$, then, more data are inserted into the cover image which causes slight changes in SC and NAE values (0.9814 and 0.00812 respectively). It also shows in Table 2 that all SC values are close to 1 and NAE values are very small. This means that the suggested method performs well in these measurements.

The data insertion activity is depending on the value of p with the block size. It is noticed that the scheme is best for a small value of p and larger block size (π) compared with other maximum value of p and small value of block size (π) but bit capacity is lower shown in Table 1. For example, when $p = 2$, then, 6,29,955 bits are inserted with an average PSNR 52.01 (dB) and BPP 1.20. When $p = 3$, then, 9,04,197 bits are inserted with average PSNR 48.95 (dB) and BPP 1.72 whereas, 11,54,588 bits are inserted with average PSNR 46.70 (dB) and BPP 2.20 when $p = 4$ for Lena image.

The exploratory results in terms of PSNR of distinct databases are looked into in Table 3 when $p = 2$. This table describes the PSNR is above 45 dB when the inserted message is over 5,80,000 bits.

The comparisons measured in PSNR (dB) among several PVO-based technique and proposed scheme are listed in Table 4. The proposed algorithm increases secrets embedding capacity (EC) contrast with another PVO-based schemes. The outcome of suggested technique is differ from other PVO-based method explained by Chang et al. [2–4], Lee et al. [10, 12], Qin et al. [24], Lu et al. [16, 17]. The hidden message capacity is 379909, 379909, 511317, 343396, 101302, 347145, 379909 and 117765 bits greater than Chang

Table 1 The secrets embedding(EC) with distinct pay load (bits) of 10 USC-SIPI gray-scale images with average PSNR (dB) and bit per pixel (BPP)

Cover Image(CI)	p	PSNR ₁	PSNR ₂	Average PSNR	EC ₁ (CI_1')	EC ₂ (CI_2')	EC = EC ₁ + EC ₂	Bit per pixel(BPP)
Lena	2	51.26	52.76	52.01	3,38,321	2,91,634	6,29,955	1.20
	3	48.13	49.78	48.95	4,92,765	4,11,432	9,04,197	1.72
	4	46.34	47.06	46.70	5,35,876	6,18,712	11,54,588	2.20
	5	42.12	41.97	42.04	7,81,310	8,48,523	16,29,833	3.10
Barbara	2	52.32	51.48	51.90	2,86,158	3,47,022	6,33,180	1.20
	3	48.60	48.91	48.75	4,62,540	4,21,205	8,83,745	1.68
	4	40.32	39.11	39.71	6,11,230	6,52,638	12,63,868	2.41
Airplane F16	5	36.02	35.31	35.66	7,05,301	7,39,012	14,44,313	2.75
	2	51.17	49.80	50.48	3,14,713	3,89,238	7,03,951	1.34
	3	49.89	48.63	49.26	4,76,012	5,86,008	10,62,020	2.02
Baboon	4	47.12	45.54	46.33	5,82,120	6,88,654	12,70,774	2.42
	5	42.43	41.44	41.93	8,11,200	8,93,635	17,04,835	3.25
	2	52.71	51.65	52.18	3,42,714	3,95,134	7,37,848	1.40
Peppers	3	47.71	48.05	47.88	4,22,004	4,76,513	8,98,517	1.71
	4	44.41	42.53	43.47	7,02,954	5,55,014	12,57,968	2.39
	5	39.53	38.65	39.09	8,49,104	8,63,277	17,12,381	3.26
Elaine	2	51.76	52.16	51.96	3,17,430	2,86,857	6,04,287	1.15
	3	47.12	49.17	48.14	4,51,302	4,02,518	8,53,820	1.62
	4	43.19	42.16	42.67	5,08,211	5,92,721	11,00,932	2.02
	5	40.17	39.03	39.60	8,28,332	9,42,451	17,70,783	3.37
Fishing boat	2	51.19	52.47	51.83	2,97,976	2,66,623	5,64,599	1.07
	3	47.37	46.92	47.14	3,87,743	4,42,912	8,30,655	1.58
	4	45.31	43.98	44.64	5,16,974	5,78,452	10,95,426	2.08
	5	41.53	39.76	40.64	6,69,027	8,13,624	14,82,651	2.82
House	2	51.89	52.45	52.17	3,29,756	2,54,532	5,84,288	1.11
	3	47.83	46.80	47.31	4,01,121	4,47,075	8,48,196	1.61
	4	43.19	41.09	42.14	5,56,034	6,02,901	11,58,935	2.21
	5	41.42	40.06	40.74	7,56,721	7,84,323	15,41,044	2.93
Sailboat on lake	2	52.04	50.31	51.17	2,91,882	3,61,636	6,53,518	1.24
	3	49.92	48.74	49.33	3,74,560	4,42,314	8,16,874	1.55
	4	43.67	43.88	43.77	5,82,219	5,63,747	11,45,966	2.18
	5	39.36	40.29	39.82	8,25,112	8,80,045	17,05,157	3.25
Tiffany	2	49.50	48.79	49.14	2,83,014	3,27,152	6,10,166	1.16
	3	46.53	45.76	46.14	4,11,176	4,75,152	8,86,328	1.69
	4	42.75	42.17	42.46	5,86,028	5,43,311	11,29,339	2.15
	5	39.61	39.12	39.36	7,49,218	7,42,330	14,91,548	2.54
Tiffany	2	50.19	52.26	51.22	3,28,656	2,47,109	5,75,765	1.09
	3	46.83	46.12	46.47	4,02,739	4,36,882	8,39,621	1.60
	4	43.53	42.61	43.07	5,08,531	5,96,207	11,04,738	2.10
	5	40.02	39.60	39.81	7,74,112	6,89,340	14,63,452	2.79

Table 2 The different measurement techniques of the suggested scheme on 10 USC-SIPI gray-scale images

Cover Image(CI)	p	Average PSNR	Average SC	Average NAE	Average Q-index	BER
Lena	2	52.01	0.9859	0.00762	0.9883	0.0114
	3	48.95	0.9814	0.00812	0.9829	0.0129
Barbara	2	51.90	0.9768	0.00187	0.9887	0.1130
	3	48.75	0.9691	0.00341	0.9786	0.1169
Airplane F16	2	50.48	0.9844	0.00659	0.9831	0.1066
	3	49.26	0.9798	0.00906	0.9817	0.1098
Baboon	2	52.18	0.9881	0.00682	0.9889	0.1147
	3	47.88	0.9813	0.00759	0.9817	0.1158
Peppers	2	51.96	0.9785	0.00187	0.9981	0.1051
	3	48.14	0.9659	0.00562	0.9885	0.1088
Elaine	2	51.83	0.9843	0.00257	0.9946	0.1159
	3	47.14	0.9806	0.00417	0.9873	0.1189
Fishing boat	2	52.17	0.9926	0.00674	0.9891	0.1076
	3	47.31	0.9901	0.00759	0.9874	0.1143
House	2	51.17	0.9785	0.00654	0.9872	0.1237
	3	49.33	0.9711	0.00722	0.9806	0.1249
Sailboat on lake	2	49.14	0.9831	0.00587	0.9856	0.1207
	3	46.53	0.9701	0.00659	0.9882	0.1198
Tiffany	2	50.19	0.9984	0.00162	0.9994	0.1187
	3	46.83	0.9910	0.00259	0.9917	0.1283

et al. [2], Chang et al. [3], Lee et al. [12], Lee et al. [10], Chang et al. [4], Qin et al. [24], Lu et al. [16] and Lu et al. [17] respectively for image Lena while value of p is 3 presented in Table 4. Figure 10 describes the differentiation graph with secrets payload for several images. It is inspected that the outcome of this paper is superior compared to another existing PVO in terms of payload while quality of image is unchanged. It is also cleared that the quality of image in PSNR (dB) is greater than other existing schemes shown in Fig. 11.

In our proposed scheme, we generate dual images from a single cover image. We embed secrets using center folding, and PVO techniques where the set of secret code (p), the number of image block (N), and the size of image block (π) plays an important role. Here, dual images are created using the center folding method. Using the proposed scheme, there must need the above-said parameters while secrets are extracted from dual stego images. At the receiver end, if we change the image block size (π) rather than original then the extracted bits are exchanged.

We also evaluate our proposed method with an additional UCID image database [21]. We select 400 images of (256×256) from each of UCID, BSDS, National Library of Medicine image databases randomly. The particulars provided in the distinct columns of Table 5 is: Image Database, Bit rate, True Positives (TP), False Positives (FP), True Negatives (TN), False Negatives (FN) and accuracy (in percentage). It is remembered that there are only 400 cover images and 400 stego (marked) images are utilized for the experiments given below. Here, $TP + FN = 400$ and $TN + FP = 400$ always.

Table 3 The results evaluated by PSNR (dB) with payload in bits of different databases

Database	Image	PSNR ₁	PSNR ₂	EC (bits)
BSDS	69040	45.11	46.53	6,69,719
	157055	47.05	45.89	5,93,432
	236037	45.34	46.83	6,11,289
	105025	47.22	45.87	5,97,862
	69020	47.01	46.49	6,83,521
	87046	46.64	45.39	6,05,012
	85048	45.01	46.64	6,18,605
	3096	47.46	46.06	5,81,407
	160068	45.13	47.67	6,58,307
	163085	46.09	47.72	6,32,129
Kodak	kodim01	44.19	47.43	6,73,464
	kodim02	46.65	46.72	5,94,126
	kodim03	47.08	45.74	6,32,760
	kodim04	46.87	45.43	6,62,830
	kodim05	47.35	46.86	6,12,432
	kodim06	45.03	46.49	5,94,211
	kodim07	45.99	48.32	6,14,876
	kodim08	47.56	45.54	5,96,567
	kodim09	44.18	46.93	6,34,521
	kodim10	46.87	46.17	6,23,486
National Library of Medicine	CXR1000.IM-0003-3001	45.86	46.95	6,65,870
	CXR1025.IM-0020-1001	45.42	47.73	6,74,223
	CXR1000.IM-0003-1001	46.83	45.99	6,38,510
	CXR1037.IM-0029-1001	46.89	45.99	6,22,441
	CXR1074.IM-0054-1001	46.99	46.23	6,43,548
	CXR1114.IM-0079-1001	47.23	46.92	6,01,467
	CXR1109.IM-0076-1001	46.08	46.98	6,85,092
	CXR1120.IM-0080-1001	46.59	45.53	6,14,201
	CXR1119.IM-0080-1001	46.56	47.81	6,47,316
CXR1151.IM-0102-1001	47.64	44.99	6,18,463	

6 Steganographic attacks

Steganalysis is a skill of finding a secret message in suspected image. Although, steganographic skill does not perform perfect security. So, it leaves a hint of data inserting in stegogramme. It offers a fruitful way of recognizing the presence of secrets into cover media. A steganalyst performs this action using several ways which are classified as: Targeted steganalysis and Blind steganalysis. The structural attack, statistical attack, and visual attack, etc. are examples of targeted steganalysis while one of the important blind steganalysis methods is Regular Singular (RS) analysis suggested by J. Fridrich [6]. We examined the marked images using RS analysis [6]. The statistical attacks and histogram attacks are studied to evaluate the innocuousness as well as robustness.

Table 4 Comparison among the other schemes and the proposed scheme with image quality in PSNR(dB) as well as embedding capacity(EC) in bits

Schemes	Measure	Lena	Baboon	Peppers	Barbara	Fishing boat
Chang et al. [2]	PSNR ₁	45.12	45.11	45.14	45.13	45.12
	PSNR ₂	45.13	45.13	45.15	45.11	45.13
	Avg. PSNR	45.13	45.12	45.15	45.12	45.13
	EC	5,24,288	5,24,148	5,23,356	5,24,288	5,24,284
Chang et al. [3]	PSNR ₁	48.13	48.14	48.11	48.14	48.13
	PSNR ₂	48.14	48.13	48.14	48.11	48.12
	Avg. PSNR	48.14	48.14	48.13	48.13	48.13
	EC	5,24,288	5,24,288	5,24,288	5,24,288	5,24,288
Lee et al. [12]	PSNR ₁	51.14	51.14	51.14	51.14	51.14
	PSNR ₂	54.16	54.14	54.17	54.16	54.16
	Avg. PSNR	52.65	52.64	52.66	52.65	52.65
	EC	3,92,880	3,93,486	3,92,796	3,93,026	3,93,040
Lee et al. [10]	PSNR ₁	49.76	49.77	49.75	49.75	49.76
	PSNR ₂	49.56	49.56	49.56	49.58	49.57
	Avg. PSNR	49.66	49.67	49.66	49.67	49.67
	EC	5,60,801	5,60,686	5,60,572	5,61,223	5,61,255
Chang et al. [4]	PSNR ₁	39.89	39.91	39.94	39.89	39.89
	PSNR ₂	39.89	39.91	39.94	39.89	39.89
	Avg. PSNR	39.89	39.91	39.94	39.89	39.89
	EC	8,02,895	8,02,524	7,99,684	8,02,888	8,02,716
Qin et al. [24]	PSNR ₁	52.11	52.04	51.25	52.12	52.11
	PSNR ₂	41.58	41.56	41.52	41.58	41.57
	Avg. PSNR	46.85	46.80	46.39	46.85	46.84
	EC	5,57,052	5,57,096	5,57,245	5,57,339	5,57,194
Lu et al. [16]	PSNR ₁	49.20	49.21	49.19	49.22	49.20
	PSNR ₂	49.21	49.20	49.21	49.20	49.21
	Avg. PSNR	49.21	49.21	49.20	49.21	49.21
	EC	5,24,288	5,24,204	5,24,192	5,24,288	5,24,284
Lu et al. ($k = 2$) [17]	PSNR ₁	49.89	49.89	49.89	49.89	49.89
	PSNR ₂	52.90	52.87	52.92	52.90	52.90
	Avg. PSNR	51.40	51.38	51.41	51.40	51.40
	EC	5,24,288	5,24,172	5,23,780	5,24,288	5,24,286
Lu et al. ($k = 3$) [17]	PSNR ₁	46.17	46.17	46.21	46.18	46.16
	PSNR ₂	47.54	47.55	47.59	47.55	47.54
	Avg. PSNR	46.86	46.86	46.90	46.87	46.85
	EC	7,86,432	7,86,042	7,77,474	7,86,432	7,86,429
Proposed method ($p = 2$)	PSNR ₁	51.26	52.71	51.76	52.32	51.89
	PSNR ₂	52.76	51.65	52.16	51.48	52.45
	Avg. PSNR	52.01	52.18	51.96	51.90	52.17
	EC	6,29,955	7,37,848	6,04,287	6,33,180	5,84,288

Table 4 (continued)

Schemes	Measure	Lena	Baboon	Peppers	Barbara	Fishing boat
Proposed method ($p = 3$)	PSNR ₁	48.13	47.71	47.12	48.60	47.83
	PSNR ₂	49.78	48.05	49.17	48.91	46.80
	Avg. PSNR	48.95	47.88	48.14	48.75	47.31
	EC	9,04,197	8,98,517	8,53,820	8,83,745	8,48,196

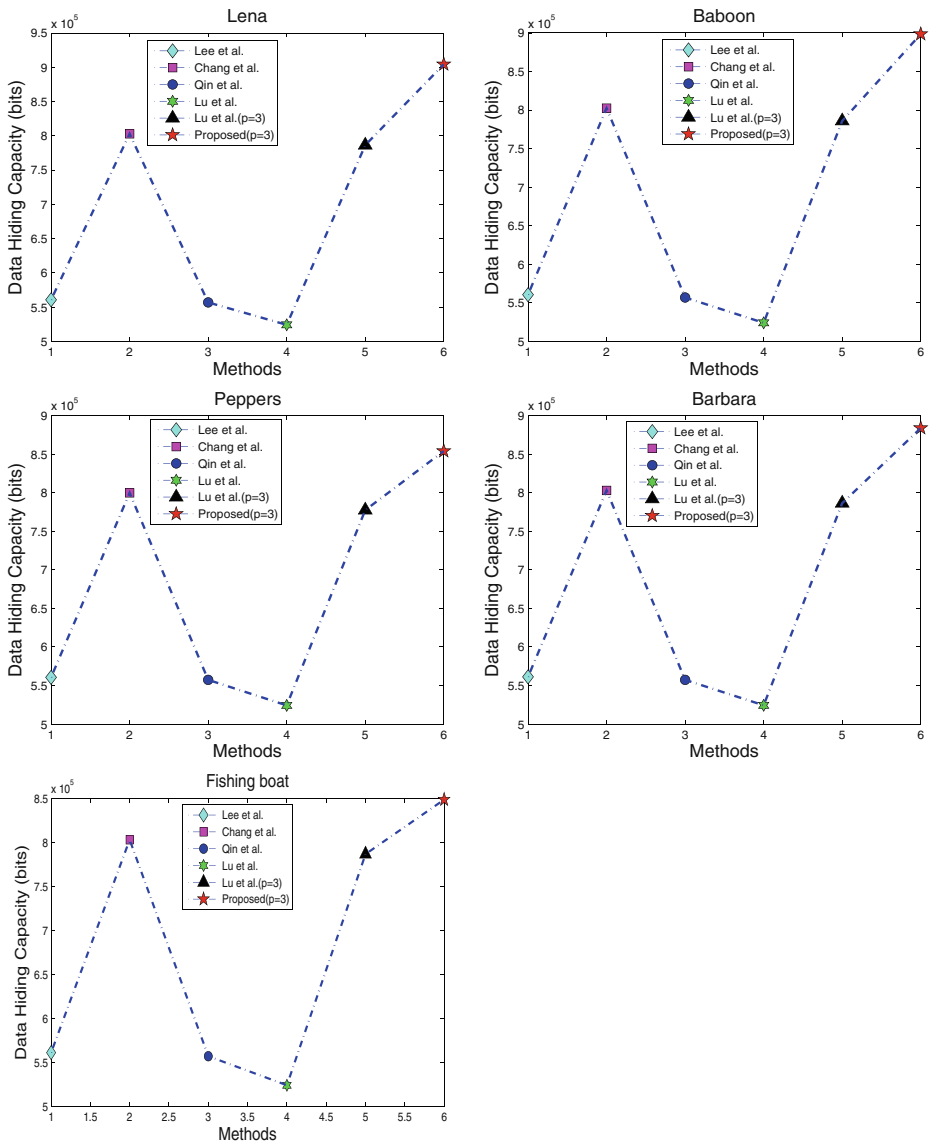


Fig. 10 Comparisons of bit capacity among the methods of Lee et al. [10], Chang et al. [4], Qin et al. [24], Lu et al. [16, 17] and proposed method with $p = 3$

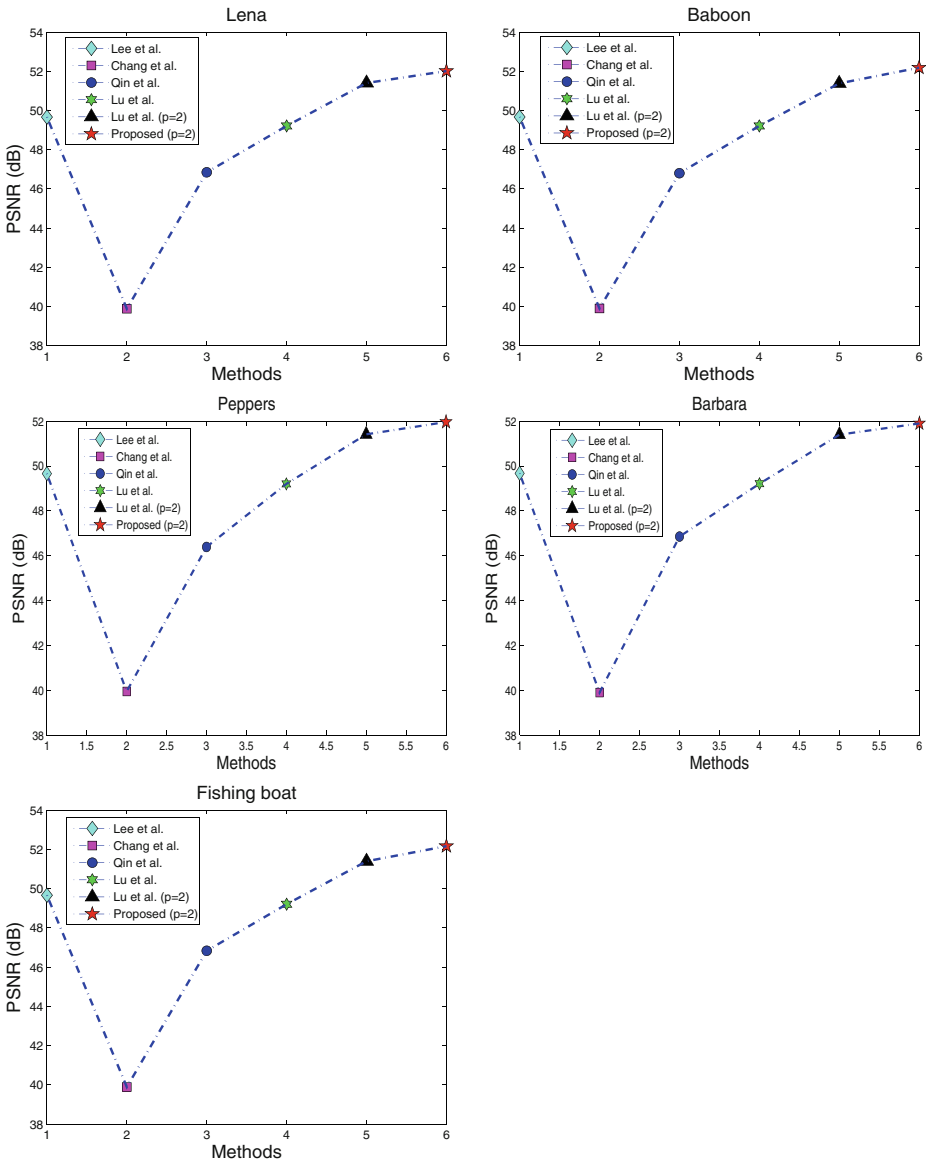


Fig. 11 Comparisons of image quality in PSNR(dB) among the methods of Lee et al. [10], Chang et al. [4], Qin et al. [24], Lu et al. [16, 17] and proposed method with $p = 2$

6.1 RS analysis

The marked images are tested using RS analysis [6] for the further analysis. The value of RS is nearer to 0 describes less modification with predicted more secure. Table 6 enlisted that both of R_M , R_{-M} and S_M , S_{-M} are nearby same. So, $R_M \cong R_{-M}$ with $S_M \cong S_{-M}$ rules are fulfilled for the marked images which are more secure in opposition to RS attack.

Table 5 True positive(TP), false positive(FP), true negative(TN), false negative(FN) and accuracy(ACC) values of the proposed method for 400 test images of each data set using different bit rate

Image Database	p	Bit rate (%)	TP	FP	TN	FN	Accuracy (%)
UCID	3	100	399	36	364	1	95.37
		50	376	108	292	24	83.5
		25	367	117	283	33	81.25
BSDS	3	100	399	2	398	1	99.62
		50	397	14	386	3	97.87
		25	399	45	355	1	94.25
National	3	100	399	3	397	1	99.5
Library of		50	382	23	377	18	94.87
Medicine		25	391	36	364	9	94.37

6.2 Histogram attack

The histogram of the cover and marked images for Lena image are shown in Fig. 12. The marked images are derived from interpolated image engaging higher secrets hiding. Also, it is noticed that the histogram shape maintains after message embedding. This ensures the minimum distortion and here is absence of step pattern that establish the robustness in opposition to histogram attacks.

6.3 Statistical attack

In this analysis, we evaluate the robustness of the suggested scheme using hiding the secrets. This method is shielded from several attacks because of containing none of the adverse

Table 6 RS analysis of marked images

Image Database	Cover Image(CI)	Marked Images	R_M	R_{-M}	S_M	S_{-M}	RS value
USC-SIPI	Lena	CI'_1	34582	32912	31413	33204	0.0524
		CI'_2	34786	32869	31754	33067	0.0485
	Peppers	CI'_1	35065	32117	31537	33156	0.0685
		CI'_2	35192	32453	33290	31245	0.0698
Kodak	kodim01	CI'_1	54316	52593	50812	52643	0.0338
		CI'_2	54514	52765	50298	52614	0.0387
	kodim02	CI'_1	55756	54657	51654	52831	0.0211
		CI'_2	55178	54798	51368	52360	0.0128
BSDS	3096	CI'_1	26170	24920	23816	24911	0.0469
		CI'_2	26560	24189	23376	24322	0.0664
	163085	CI'_1	27452	26152	24985	25772	0.0398
		CI'_2	27819	26559	24863	25626	0.0384
National Library of Medicine	CXR1000.IM-0003-3001	CI'_1	30822	28426	25645	26731	0.0616
		CI'_2	30652	28573	25923	26879	0.0536
	CXR1000.IM-0003-1001	CI'_1	30169	28467	25621	26870	0.0528
		CI'_2	31231	29385	26543	27467	0.0479

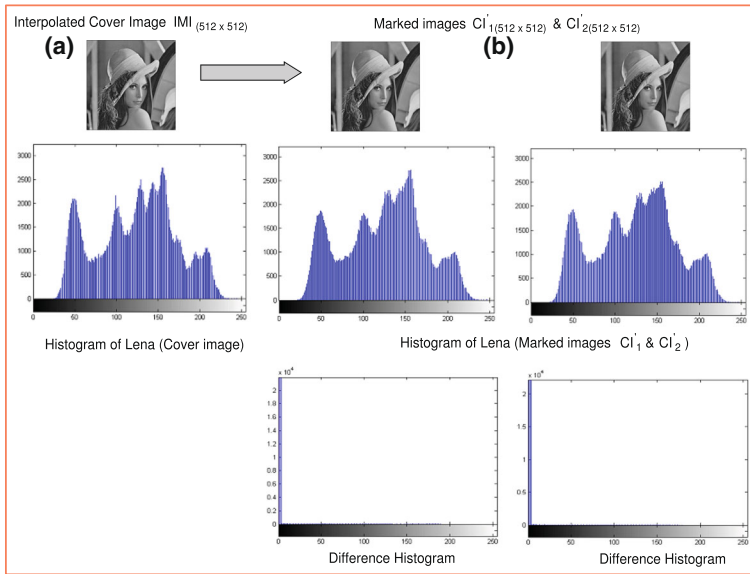


Fig. 12 Histogram difference of cover and marked image for Lena in USC-SIPI data base [31]

effects. We calculate the correlation coefficient (CC) and standard deviation (SD) displayed in Tables 7 and 8, respectively. Here, the CC and SD are determined through the interpolated real image together with the stego. The result shows that the CC is nearby 1, which establishes maximum security for secrets. It also displays in Table 8 that, the SD is nearby to 0, which establishes maximum security for secrets. The suggested scheme gained beneficial power against attacks. The cover image together with the secrets is regained from dual stego images without any kind of data loss.

6.4 Structural similarity index and normalized cross-correlation

The process for examining the similarity between the real and the stego images is known as the structural similarity (SSIM) index. This method is built upon 3 keys as contrast, structural with luminance key. The SSIM is the combination of three keys. To observed some patterns in-between real and stego images, there needs of normalised cross-correlation (NCC). It is utilized for block matching fields, image registration with video as well as image compression, etc. The SSIM and NCC values are explored in Table 9.

6.5 Tamper detection and image recovery

For the robustness of our proposed scheme, we evaluated PSNR, SD, CC in the presence of several noise levels such as noise from salt and pepper, cropping, and copy-move forgery attacks. Figures 13, 14 and 15 displays the experimental outcomes after using salt and pepper, cropping as well as copy-move forgery attack respectively, with distinct noise intensity. After secrets taking out, the extracted secret image is marginally changed where the location of tamper in the restored image is recognized successfully. The results SD and CC show the robustness of the proposed scheme.

Table 7 The correlation coefficient(CC) between the original and marked image

Image Database	Cover Image(CI)	p	EC (bits)	CC		
				$IMI \& CI'_1$	$IMI \& CI'_2$	
USC-SIPI	Lena	2	6,29,955	0.9995	0.9995	
		3	9,04,197	0.9992	0.9992	
		4	11,54,588	0.9989	0.9998	
		5	16,29,833	0.9985	0.9985	
		2	6,04,287	0.9995	0.9995	
	Peppers	3	8,53,820	0.9991	0.9991	
		4	11,00,932	0.9989	0.9989	
		5	17,70,783	0.9983	0.9983	
		2	6,73,464	0.9994	0.9994	
		3	8,90,514	0.9991	0.9991	
	Kodak	kodim01	4	11,40,936	0.9987	0.9994
			5	14,71,865	0.9984	0.9984
			2	5,94,126	0.9994	0.9994
			3	8,63,748	0.9991	0.9991
			4	11,08,314	0.9988	0.9988
kodim02		5	14,28,837	0.9986	0.9986	
		2	5,81,407	0.9996	0.9996	
		3	8,62,119	0.9992	0.9992	
		4	11,08,327	0.9988	0.9988	
		5	14,16,042	0.9986	0.9986	
BSDS	3096	2	6,32,129	0.9995	0.9995	
		3	8,70,655	0.9993	0.9993	
		4	11,40,657	0.9990	0.9990	
		5	14,32,813	0.9986	0.9986	
		2	6,65,870	0.9995	0.9995	
	National Library of Medicine	CXR1000_IM- 0003-3001	3	8,61,239	0.9991	0.9991
			4	11,37,861	0.9988	0.9988
			5	14,79,468	0.9986	0.9986
			2	6,38,510	0.9995	0.9995
			3	8,52,538	0.9993	0.9993
CXR1000_IM- 0003-1001		4	11,20,659	0.9989	0.9989	
		5	14,55,210	0.9986	0.9986	

7 Conclusions and future works

This paper proposed an improved center folding technique through Directional PVO for reversible data hiding with different sizes of p (set of data) and image block. The algorithms for data inserting and removing are developed in a manner so that huge messages are inserted and removed to and from images respectively. The suggested scheme carries out reliable message communication as inserting the secrets among dual images. Using our method, we get an average PSNR value above 51 dB and an average embedding capacity

Table 8 The standard deviation(SD) between the original and marked images

Image	Cover	p	EC	IMI	SD				
					CI'_1	Difference	CI'_2	Difference	
Database	Image(CI)		(bits)						
USC-SIPI	Lena	2	6,29,955		39.2969	0.0315	39.2991	0.0337	
		3	9,04,197	39.2654	39.3327	0.0673	39.3394	0.0740	
		4	11,54,588		39.4981	0.2327	39.4916	0.2262	
		5	16,29,833		39.8524	0.5870	39.8537	0.5883	
		2	6,04,287		43.7499	0.0310	43.7476	0.0287	
	Peppers	3	8,53,820	43.7189	43.7813	0.0624	43.7866	0.0677	
		4	11,00,932		44.1067	0.3878	44.1026	0.3837	
		5	17,70,783		44.3832	0.6643	44.3856	0.6667	
		2	6,73,464		41.3917	0.0313	41.3959	0.0355	
		3	8,90,514	41.3604	41.4233	0.0629	41.4294	0.0690	
	Kodak	kodim01	4	11,40,936		41.6702	0.3098	41.6813	0.3209
			5	14,71,865		41.9278	0.5674	41.9279	0.5675
			2	5,94,126		37.2568	0.0289	37.2574	0.0295
			3	8,63,748	37.2279	37.2915	0.0636	37.2936	0.0657
			4	11,08,314		37.4562	0.2283	37.4587	0.2308
kodim02		5	14,28,837		37.7038	0.4759	37.7016	0.4737	
		2	5,81,407		39.6809	0.0287	39.6828	0.0306	
		3	8,62,119	39.6522	39.7185	0.0663	39.7204	0.0682	
		4	11,08,327		39.9287	0.2765	39.9215	0.2693	
		5	14,16,042		40.1807	0.5285	40.1852	0.5330	
BSDS	3096	2	6,32,129		26.3086	0.0380	26.3012	0.0306	
		3	8,70,655	26.2706	26.3389	0.0683	26.3391	0.0685	
		4	11,40,657		26.5721	0.3015	26.5749	0.3043	
		5	14,32,813		26.8210	0.5504	26.8156	0.5450	
		2	6,65,870		41.2112	0.0282	41.2135	0.0305	
	National Library of Medicine	CXR1000_IM-0003-3001	3	8,61,239	41.1830	41.2477	0.0647	41.2481	0.0651
			4	11,37,861		41.5187	0.3357	41.5116	0.3286
			5	14,79,468		41.8245	0.6415	41.8176	0.6346
			2	6,38,510		38.4631	0.0242	38.4675	0.0286
			3	8,52,538	38.4389	38.5076	0.0687	38.5021	0.0632
CXR1000_IM-0003-1001		4	11,20,659		38.6429	0.2040	38.6488	0.2099	
		5	14,55,210		38.9502	0.5113	38.9527	0.5138	

above 6,00,000 bits when $p = 2$. The proposed method carries out satisfactory outcomes as well as better performance contrast with other PVO works and it is more reliable against many steganographic attacks. In non-overlapped-blocks, the size of image block (π), and the set of secret code (p) plays an important role in our suggested method. If p increases and π decreases, then many secrets are inserted into the cover image. Moreover, a pixel will be embedded at most 4 times when $p \geq 1$. If we created overlapped-blocks, then a pixel will be embedded at most 7 times. Increasing overlapped-block size improves PSNR in dB of stego image. So, increased overlapped-block size and value of p enhances the secret

Table 9 The data bit capacity(EC) of four different image databases with structural similarity(SSIM) index and normalized cross-correlation(NCC)

Image Database	Cover Image(CI)	p	Data (bits)	IMI vs CI'_1		IMI vs CI'_2		
				SSIM	NCC	SSIM	NCC	
USC-SIPI	Lena	2	6,29,955	0.9964	0.9994	0.9964	0.9994	
		3	9,04,197	0.9960	0.9991	0.9961	0.9991	
		4	11,54,588	0.9952	0.9988	0.9954	0.9988	
		5	16,29,833	0.9945	0.9980	0.9945	0.9980	
		Peppers	2	6,04,287	0.9968	0.9982	0.9968	0.9982
	Kodak	kodim01	3	8,53,820	0.9961	0.9976	0.9963	0.9976
			4	11,00,932	0.9956	0.9971	0.9958	0.9971
			5	17,70,783	0.9949	0.9960	0.9951	0.9960
			2	6,73,464	0.9962	0.9994	0.9963	0.9994
			3	8,90,514	0.9958	0.9991	0.9958	0.9991
kodim02		4	11,40,936	0.9945	0.9989	0.9945	0.9989	
		5	14,71,865	0.9872	0.9988	0.9872	0.9988	
		2	5,94,126	0.9988	0.9992	0.9988	0.9992	
		3	8,63,748	0.9985	0.9989	0.9985	0.9989	
		4	11,08,314	0.9972	0.9986	0.9972	0.9986	
BSDS	3096	5	14,28,837	0.9959	0.9982	0.9959	0.9982	
		2	5,81,407	0.9982	0.9989	0.9981	0.9989	
		3	8,62,119	0.9968	0.9987	0.9970	0.9987	
		4	11,08,327	0.9954	0.9981	0.9954	0.9981	
		5	14,16,042	0.9942	0.9976	0.9942	0.9976	
	163085	2	6,32,129	0.9984	0.9988	0.9982	0.9988	
		3	8,70,655	0.9971	0.9982	0.9971	0.9982	
		4	11,40,657	0.9962	0.9982	0.9963	0.9982	
		5	14,32,813	0.9956	0.9963	0.9955	0.9963	
		National Library of Medicine	CXR1000.IM-0003-3001	2	6,65,870	0.9983	0.9989	0.9983
3	8,61,239			0.9976	0.9984	0.9975	0.9984	
4	11,37,861			0.9966	0.9979	0.9967	0.9979	
5	14,79,468			0.9954	0.9958	0.9956	0.9958	
2	6,38,510			0.9978	0.9984	0.9976	0.9984	
CXR1000.IM-	3		8,52,538	0.9956	0.9978	0.9956	0.9978	
	4		11,20,659	0.9943	0.9962	0.9945	0.9962	
	5		14,55,210	0.9892	0.9854	0.9895	0.9954	

message as well as the visual quality of stego image. This technique can be implemented as a host signal to other media types such as audio, video, and animated image. It can handle vast quantities of hidden data bits being stored and forwarded. This can also be applied in different contexts, including transform, compress, and random domain. This can also be a valuable guide for professionals in the area of fraud identification and copyright protection.

Cover Image (256 x 256) (C ₁)	Secret Image (276 x 276)	Marked Images (512 x 512) (C ₁ & C ₂)		Tampered marked Images (512 x 512) (C ₁ & C ₂)		Recovered Secret Image	Recover Cover Image	Statistical Analysis
								Difference of SD between IMI & CI=49.57-23.65 =25.92 CC between IMI & CI=0.72
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Salt & pepper (0.01)	PSNR= 18.32 dB	PSNR= 28.09 dB	
								Difference of SD between IMI & CI=49.57-24.19 =25.38 CC between IMI & CI=0.68
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.01)	No attack	PSNR= 16.57 dB	PSNR= 28.71 dB	
								Difference of SD between IMI & CI=49.57-24.25 =25.32 CC between IMI & CI=0.59
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.01)	Salt & pepper (0.01)	PSNR= 16.75 dB	PSNR= 26.58 dB	
								Difference of SD between IMI & CI=49.57-25.38 =24.19 CC between IMI & CI=0.70
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Salt & pepper (0.1)	PSNR= 15.23 dB	PSNR= 26.92 dB	
								Difference of SD between IMI & CI=49.57-25.86 =23.71 CC between IMI & CI=0.57
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.1)	No attack	PSNR= 15.90 dB	PSNR= 24.69 dB	
								Difference of SD between IMI & CI=49.57-26.07 =23.50 CC between IMI & CI=0.51
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.1)	Salt & pepper (0.1)	PSNR= 16.82 dB	PSNR= 25.72 dB	
								Difference of SD between IMI & CI=49.57-26.78 =22.79 CC between IMI & CI=0.61
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Salt & pepper (0.3)	PSNR= 13.04 dB	PSNR= 23.14 dB	
								Difference of SD between IMI & CI=49.57-24.31 =25.26 CC between IMI & CI=0.58
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.3)	No attack	PSNR= 12.87 dB	PSNR= 24.91 dB	
								Difference of SD between IMI & CI=49.57-25.16 =24.41 CC between IMI & CI=0.46
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Salt & pepper (0.3)	Salt & pepper (0.3)	PSNR= 11.08 dB	PSNR= 19.83 dB	

Fig. 13 Cover image and data restoration using different level of salt and pepper noise on Sailboat_on_lake image

Cover Image (256 x 256) (CI)	Secret Image (276 x 276)	Marked Images (512 x 512) (CI ₁ & CI ₂)		Tampered marked Images (512 x 512) (CI ₁ & CI ₂)		Recovered Secret Image	Recover Cover Image	Statistical Analysis
								Difference of SD between IMI & CI= $49.57-24.64=24.93$ CC between IMI & CI= 0.70
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Cropping (10%)	PSNR= 18.42 dB	PSNR= 29.09 dB	
								Difference of SD between IMI & CI= $49.57-26.31=23.26$ CC between IMI & CI= 0.65
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (10%)	No attack	PSNR= 16.37 dB	PSNR= 29.64 dB	
								Difference of SD between IMI & CI= $49.57-25.78=23.79$ CC between IMI & CI= 0.63
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (10%)	Cropping (10%)	PSNR= 16.12 dB	PSNR= 27.83 dB	
								Difference of SD between IMI & CI= $49.57-26.43=23.14$ CC between IMI & CI= 0.69
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Cropping (20%)	PSNR= 17.42 dB	PSNR= 27.96 dB	
								Difference of SD between IMI & CI= $49.57-24.39=25.18$ CC between IMI & CI= 0.60
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (20%)	No attack	PSNR= 15.86 dB	PSNR= 25.74 dB	
								Difference of SD between IMI & CI= $49.57-25.46=24.11$ CC between IMI & CI= 0.56
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (20%)	Cropping (20%)	PSNR= 17.29 dB	PSNR= 26.78 dB	
								Difference of SD between IMI & CI= $49.57-24.35=25.22$ CC between IMI & CI= 0.62
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	Cropping (30%)	PSNR= 14.24 dB	PSNR= 25.08 dB	
								Difference of SD between IMI & CI= $49.57-24.71=24.86$ CC between IMI & CI= 0.61
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (30%)	No attack	PSNR= 12.89 dB	PSNR= 24.67 dB	
								Difference of SD between IMI & CI= $49.57-26.47=23.10$ CC between IMI & CI= 0.56
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	Cropping (30%)	Cropping (30%)	PSNR= 12.08 dB	PSNR= 20.53 dB	

Fig. 14 Cover image and data restoration using different level of cropping on Sailboat_on_lake image

Cover Image (256 x 256) (C ₁)	Secret Image (276 x 276)	Marked Images (512 x 512) (C ₁ & C ₂)		Tampered marked Images (512 x 512) (C ₁ & C ₂)		Recovered Secret Image	Recover Cover Image	Statistical Analysis
								Difference of SD between IMI & CI=49.57-26.67 =22.90 CC between IMI & CI=0.68
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	CopyMove (10%)	PSNR= 18.72 dB	PSNR= 29.09 dB	
								Difference of SD between IMI & CI=49.57-26.45 =23.12 CC between IMI & CI=0.63
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (10%)	No attack	PSNR= 17.61 dB	PSNR= 29.64 dB	
								Difference of SD between IMI & CI=49.57-24.84 =24.73 CC between IMI & CI=0.59
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (10%)	CopyMove (10%)	PSNR= 17.12 dB	PSNR= 27.83 dB	
								Difference of SD between IMI & CI=49.57-25.31 =24.26 CC between IMI & CI=0.58
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	CopyMove (20%)	PSNR= 18.22 dB	PSNR= 27.96 dB	
								Difference of SD between IMI & CI=49.57-25.33 =24.24 CC between IMI & CI=0.62
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (20%)	No attack	PSNR= 18.07 dB	PSNR= 25.74 dB	
								Difference of SD between IMI & CI=49.57-26.42 =23.15 CC between IMI & CI=0.49
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (20%)	CopyMove (20%)	PSNR= 17.89 dB	PSNR= 26.78 dB	
								Difference of SD between IMI & CI=49.57-25.71 =23.86 CC between IMI & CI=0.64
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	No attack	CopyMove (30%)	PSNR= 15.34 dB	PSNR= 25.08 dB	
								Difference of SD between IMI & CI=49.57-25.87 =23.70 CC between IMI & CI=0.57
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (30%)	No attack	PSNR= 15.89 dB	PSNR= 24.67 dB	
								Difference of SD between IMI & CI=49.57-26.57 =23.00 CC between IMI & CI=0.51
Sailboat	Shape Image	PSNR=49.50 dB	PSNR=48.79 dB	CopyMove (30%)	CopyMove (30%)	PSNR= 14.27 dB	PSNR= 20.53 dB	

Fig. 15 Cover image and data restoration using copy-move forgery on Sailboat_on_lake image

References

1. Alattar AM (2004) Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Trans Image Process* 13(8):1147–1156
2. Chang CC, Kieu TD, Chou YC (2007) Reversible data hiding scheme using two steganographic images. In: *TENCON 2007–2007 IEEE Region 10 conference*. IEEE, pp 1–4
3. Chang CC, Chou YC, Kieu TD (2009) Information hiding in dual images with reversibility. In: *2009 Third international conference on multimedia and ubiquitous engineering*. IEEE, pp 145–152
4. Chang CC, Lu TC, Horng G, Huang YH, Hsu YM (2013) A high payload data embedding scheme using dual stego-images with reversibility. In: *2013 9th International conference on information, communications & signal processing*. IEEE, pp 1–5
5. Coltuc D (2011) Improved embedding for prediction-based reversible watermarking. *IEEE Trans Inf Forensics Secur* 6(3):873–882
6. Fridrich J, Goljan M, Du R (2001) Reliable detection of LSB steganography in color and grayscale images. In: *Proceedings of the 2001 workshop on multimedia and security: new challenges*, pp 27–30
7. Hu Y, Lee HK, Li J (2008) DE-based reversible data hiding with improved overflow location map. *IEEE Trans Circ Syst Video Technol* 19(2):250–260
8. Kim HJ, Sachnev V, Shi YQ, Nam J, Choo HG (2008) A novel difference expansion transform for reversible data embedding. *IEEE Trans Inf Forensics Secur* 3(3):456–465
9. Kodak Lossless True Color Image Suite, <http://r0k.us/graphics/kodak/>. Accessed 11 Dec 2017
10. Lee CF, Huang YL (2013) Reversible data hiding scheme based on dual stegano-images using orientation combinations. *Telecommun Syst* 52(4):2237–2247
11. Lee CC, Wu HC, Tsai CS, Chu YP (2008) Adaptive lossless steganographic scheme with centralized difference expansion. *Pattern Recognit* 41(6):2097–2106
12. Lee CF, Wang KH, Chang CC, Huang YL (2009) A reversible data hiding scheme based on dual steganographic images. In: *Proceedings of the 3rd international conference on ubiquitous information management and communication*, pp 228–237
13. Li X, Yang B, Zeng T (2011) Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection. *IEEE Trans Image Process* 20(12):3524–3533
14. Li X, Li J, Li B, Yang B (2013) High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion. *Signal Process* 93(1):198–205
15. Lou DC, Hu MC, Liu JL (2009) Multiple layer data hiding scheme for medical images. *Comput Stand Interfaces* 31(2):329–335
16. Lu TC, Tseng CY, Wu JH (2015) Dual imaging-based reversible hiding technique using LSB matching. *Signal Process* 108:77–89
17. Lu TC, Wu JH, Huang CC (2015) Dual-image-based reversible data hiding method using center folding strategy. *Signal Process* 115:195–213
18. Lu TC, Huang SR, Huang SW (2020) Reversible hiding method for interpolation images featuring a multilayer center folding strategy. *Soft Comput*. <https://doi.org/10.1007/s00500-020-05129-7>
19. Meikap S, Jana B (2018) Directional PVO for reversible data hiding scheme with image interpolation. *Multimed Tools Appl* 77(23):31281–31311
20. Meikap S, Jana B (2019) Directional pixel value ordering based secret sharing using sub-sampled image exploiting Lagrange polynomial. *SN Appl Sci* 1(6):645
21. Nottingham Trent University, UCID Image Database. <http://jasoncantarella.com/downloads/ucid.v2.tar.gz>. Accessed 6 Nov 2019
22. Ou B, Li X, Wang J (2016) Improved PVO-based reversible data hiding: a new implementation based on multiple histograms modification. *J Vis Commun Image Represent* 38:328–339
23. Peng F, Li X, Yang B (2014) Improved PVO-based reversible data hiding. *Digit Signal Process* 25:255–265
24. Qin C, Chang CC, Hsu TJ (2015) Reversible data hiding scheme based on exploiting modification direction with two steganographic images. *Multimed Tools Appl* 74(15):5861–5872
25. Qu X, Kim HJ (2015) Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding. *Signal Process* 111:249–260
26. Sachnev V, Kim HJ, Nam J, Suresh S, Shi YQ (2009) Reversible watermarking algorithm using sorting and prediction. *IEEE Trans Circ Syst Video Technol* 19(7):989–999
27. The National Library of Medicine presents MedPix®, <https://openi.nlm.nih.gov/gridquery.php?q=&it=x>. Accessed 11 Dec 2017
28. Thodi DM, Rodríguez JJ (2007) Expansion embedding techniques for reversible watermarking. *IEEE Trans Image Process* 16(3):721–730

29. Tian J (2003) Reversible data embedding using a difference expansion. *IEEE Trans Circ Syst Vid Technol* 13(8):890–896
30. University of California, Berkeley, “The Berkeley Segmentation Dataset and Benchmark”, http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/BSR/BSR_bsd500.tgz. Accessed 11 Dec 2017
31. University of Southern California, “The USC-SIPI Image Database”, <http://sipi.usc.edu/database/database.php?volume=misc>. Accessed 11 Dec 2017

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.