Enhanced payload and trade-off for image steganography via a novel pixel digits alteration

Vinita Verma[1] · Sunil K. Muttoo[1] · V. B. Singh[2]

Abstract
Many high capacity image steganography algorithms have been proposed but at the cost of lower imperceptibility. To fill the gap, this paper proposes a novel RGB image steganography technique that not only enhances the hiding capacity but also largely improves the trade-off between capacity and imperceptibility on maximum embedding rate. The technique encrypts the secret data (text, grayscale or a color image) into a sequence of two-digit decimal values. A digit in the sequence is hidden in a cover pixel by altering the pixel's digits to the nearest pixel value possible. Most common spatial hiding techniques are the Least Significant Bit (LSB) substitution and Pixel Value Differencing (PVD) which hide data in the form of bits. The proposed linear time digit based hiding technique outperforms LSB substitution, PVD or their variants in terms of both capacity and imperceptibility. It also outperforms the existing digit hiding methods in color images. Moreover, the method is secure against the two detection schemes of Weighted Stego-Image steganalysis.

1 Introduction

Steganography is a method of hiding secret data within a non-secret data (cover) without any visual distortion. Based on the cover, steganography is classified as image, text, audio or video

---

✉ Vinita Verma
vinitaducs@gmail.com

Sunil K. Muttoo
drskmuttoo@gmail.com

V. B. Singh
vbsinghdcacdu@gmail.com

1   Department of Computer Science, University of Delhi, Delhi, India

2   Department of Computer Science, Delhi College of Arts and Commerce, University of Delhi, Delhi, India

steganography. Image steganography mainly uses spatial and a transform domain to hide the data in LSBs and frequency coefficients of the pixels respectively. LSB substitution and PVD are the most common spatial hiding techniques. PVD uses the difference between pixel values to hide more data bits in noisy relative to smooth areas. However, algorithms hiding binary data show low embedding capacity and the ones with large hiding capacity result in lower imperceptibility [1].

To enhance payload and imperceptibility, variants of LSB substitution and PVD have been proposed. Seven-way PVD [15] was proposed using nine-pixel blocks to increase both capacity and Peak Signal-to-Noise Ratio (PSNR). PVD combined with modulus function [18], hiding data in R and B planes, increased payload and PSNR. Nine-pixel differencing with modified LSB [20] readjusted the pixel values to minimize distortion. Nonadaptive PVD resulted in a higher capacity than the adaptive method [21]. Five and eight-way PVD with modified LSB [22] were proposed to get high bit rates, lesser distortion and address falloff boundary problem (pixel exceeding upper (255) or lower (0) range value after substitution). LSB combined with eight directional PVD [23] enhanced capacity, PSNR and evaded steganalysis. Pixel indicator based adaptive LSB and PVD [27] on scrambled RGB images increased capacity by 50%. An adaptive method using LSB and PVD [5] for grayscale images improved both capacity and PSNR. Multi-pixel differencing and LSB [10] embedded data in noisy and smooth blocks of the grayscale image respectively. Construction based data hiding [13] transformed the message into a fingerprint image directly. It resulted in accurate extraction and hard detection of the message. Image steganography robust to JPEG compression [26] was presented by hiding data in channel compressed version of the original image. Data were extracted with exact accuracy. Overlapping block-based PVD [16] obtained from combining color components improved both capacity and PSNR. QVD (quotient value differencing) and MFPVD (modulus function PVD) [24] were combined obtaining higher payload and PSNR respectively. QVD combined with LSB [25] achieved the highest capacity in the available literature but at the cost of reduced PSNR. A literature review of spatial domain steganography techniques follows in [7].

The proposed technique, in this paper, hides data digits instead of binary data. To the best of our knowledge, the proposed technique has achieved the second-best capacity with the highest PSNR relative to the technique [25]. The rest of the paper is organized as follows. Section 2 describes the related work. The methodology of the proposed technique is discussed in Section 3. Section 4 presents a comparative analysis of the proposed technique with existing spatial hiding techniques. Section 5 performs security analysis of the proposed technique. The conclusion is presented in Section 6.

2 Related work

Digit-by-digit data hiding using modulus function [28] was proposed for grayscale images. The method resulted in a higher capacity and produced better results than LSB substitution. Pixel Value Modification (PVM) [14] was proposed for RGB images using a modulus three function. The method resulted in increased hiding capacity. PVM method was enhanced [12] and presented using varying modulus functions. However, the hiding capacity depended on the modulus function. Hiding of ASCII values corresponding to characters, in the least significant digits of the pixels [19] was proposed using medical cover images. However, only pixel values 0–249 were used for embedding. Parity-bit PVD was combined with improved Right Most

Digit Replacement (iRMDR) [8], increasing both capacity and PSNR for grayscale images. RMDR with adaptive LSB [9] was proposed, resulting in the high capacity and good visual symmetry for grayscale images. Generalization of the concept of a single-digit sum [3] was presented for supporting the number system with any base to enable hiding of the data bits in pixels.

The proposed method in comparison to the existing digit hiding methods, embeds 393,216 bytes (3,145,728 bits), 512 × 512 grayscale or a 256 × 512 color image in a 512 × 512 color image, relatively achieving higher capacity. Indeed, the trade-off between capacity and imperceptibility (39 dB approximate PSNR at 3,145,728 bits or 4bpp) obtained using the proposed method has not been found in the available literature.

3 Proposed method

The method proposed in this paper compresses ASCII values of text characters or pixel values of data image into two-digit decimal values. The compression provides encryption and enables large hiding. Sections 3.1 and 3.2 present the overview of the procedure to hide text and data image respectively. After conversion, the individual digits of the data decimal values are hidden sequentially in cover pixels using the method provided in Section 3.3. Section 3.4 illustrates mathematically that the changes in cover pixel values after hiding data are minimal. Data extraction is described in Section 3.5.

3.1 Text hiding procedure

Step 1:   the text is converted into ASCII format.
Step 2:   ASCII values 32–126 (x) are compressed to range 00–95 using a function,

$$F(x) = \left\lfloor \left(\frac{x-lb}{ub-lb}\right)*max \right\rfloor; lb = 32, ub = 126, max = 95$$

The control characters Carriage return, new line, horizontal and vertical tab are converted to 96, 97, 98 and 99 respectively. Thus, two-digit value is obtained corresponding to every character.

Step 3:   a sequence or a string S of digits is created using the obtained two-digit values. The first digit in S is embedded in the first blue, the second digit in first green and third in first red pixel. The next three digits in S are inserted in second-pixel components. Following this way, digits in S are embedded sequentially into the pixel components.

A character (two-digit value) is embedded in two pixels. Therefore, XxY RGB image (XxYx3 pixels) can hide (XxYx3)/2 bytes using the proposed method. For example, 512 × 512 RGB image can hide total 393,216 bytes or 3,145,728 bits.

3.2 Data image hiding

Data image pixels are compressed to range 00–64 using respective six MSBs, which are sufficient to recover the image in acceptable detail. Therefore, the data comprises of pixel values 00–64. This is followed by Step 3 in Section 3.1 to hide the compressed grayscale

Fig. 1 Individual digits of a pixel value

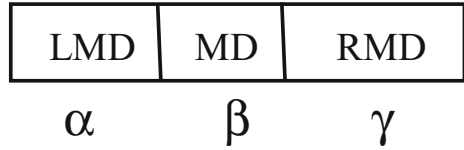| LMD	MD	RMD

$$\alpha \qquad \beta \qquad \gamma$$

image. For color data image, a pixel (00–64) of each plane is inserted into two consecutive pixels of the corresponding cover plane. For example, individual digits of the 1st red pixel are embedded into 1st and 2nd red pixels of the cover image, 2nd red pixel into 3rd and 4th red cover pixels. Following the process for the other two planes, X x (Y/2) or (X/2) xY color image can be embedded in an XxY color image.

3.3 Hiding digit in a pixel value

Pixel values lie in range 000–255. Fig. 1 shows the maximum three digits of a pixel p (Most Significant Digit or leftmost digit (LMD) α, a middle digit (MD) β, Least Significant Digit or rightmost digit (RMD) γ). The pixel value p is split into individual digits, using place values as $p = 100\alpha + 10\beta + \gamma$. Based upon the difference between the rightmost digit and digit d to be
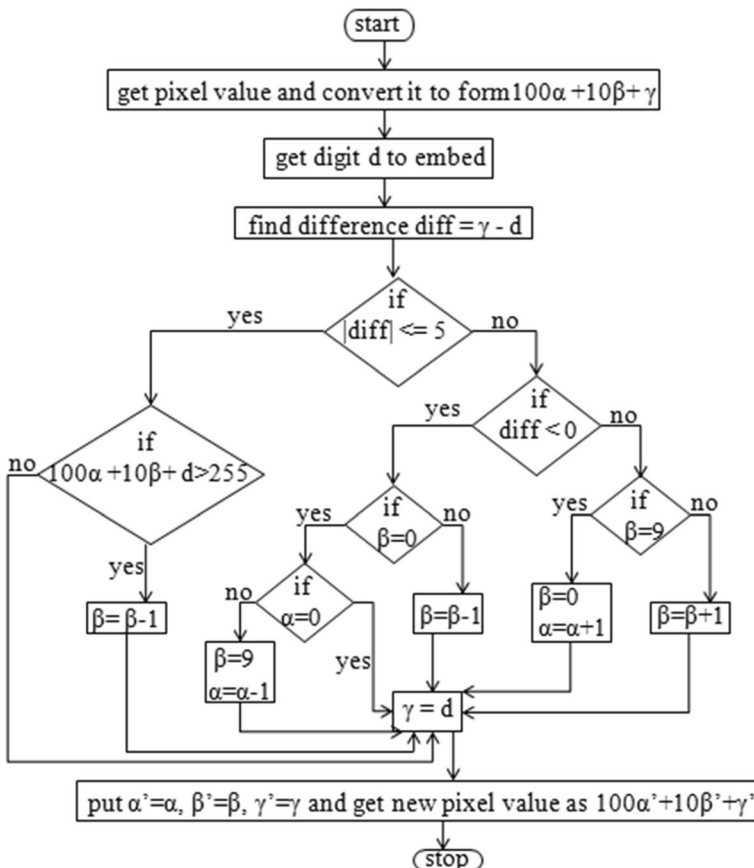

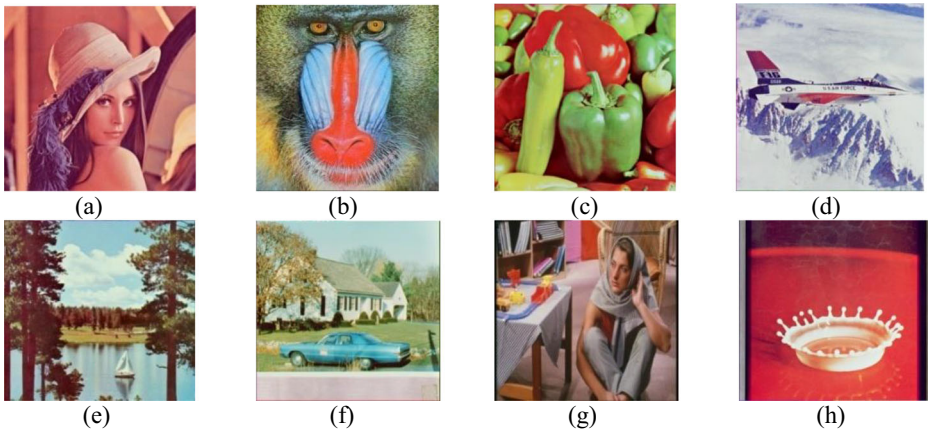**Fig. 2** Procedure to hide a digit in a pixel value

Fig. 3 512 × 512 color cover images: **a** lena **b** baboon **c** pepper **d** jet **e** boat **f** house **g** barbara **h** pot

embedded, new pixel value p' is computed as p' = 100α' + 10β' + γ'. *The proposed method replaces the rightmost digit by d, followed by changing the middle and/or leftmost digit if required, to bring minimal change in the pixel value.* When the difference between RMD and d is up to value 5, RMD is replaced by d without any change in either MD or LMD. In a rare case of a modified value obtained beyond 255, the middle digit is decreased by 1 to solve the falloff boundary problem. However, when the difference between RMD and d is more than 5, i.e. change in the pixel p by value more than 5, then MD and/or LMD are also changed to ensure minimal changes in cover pixels. In such a case, the method uses the positive and negative difference as an indicator to increase and decrease the pixel value respectively. To increase the pixel value, MD (β) is increased by 1 after replacing RMD by d. Middle digit β ∈ {0, …, 8} upon increasing changes to β' ∈ {1, …, 9}. But, increasing MD equal to 9 changes it to 0 and increases LMD by 1 as in simple addition. To decrease the pixel value, MD is decremented by 1. MD equal to 0 upon decreasing changes to 9 and LMD decreases by 1 as in mathematical subtraction. The threshold for comparison has been selected to be 5 since it resulted in minimal value changes. Fig. 2 illustrates the manipulation of individual digits of a pixel value to hide a digit in the rightmost digit of the pixel. The method thus uses the rightmost digit or least significant digit replacement.
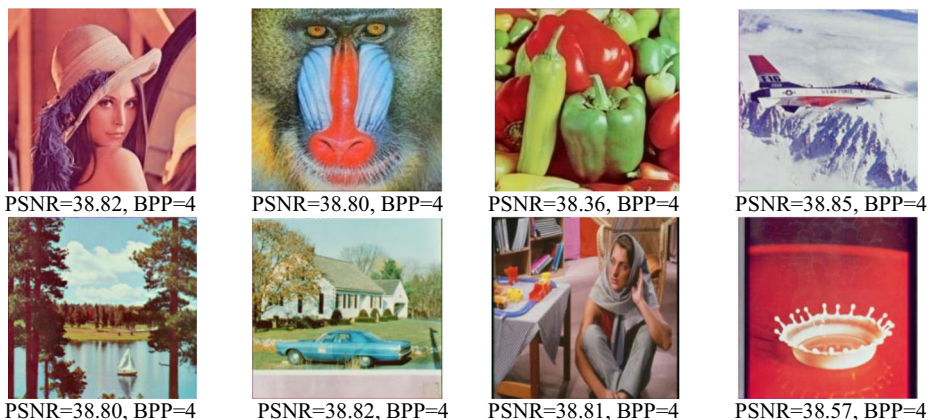

Fig. 4 Stego images on hiding 3,145,728 bits in respective cover images using the proposed technique

3.4 Changes in pixel values after hiding data

Amount of distortion to the cover pixels using the proposed method has been computed mathematically corresponding to each case of Fig. 2. Let original pixel be $p = 100\alpha + 10\beta + \gamma$ and new pixel value be $p' = 100\alpha' + 10\beta' + \gamma'$.

case 1:   if $|\gamma - d| <= 5$, put $\gamma' = d, \beta' = \beta, \alpha' = \alpha$ (1)

  $p - p' = 100\alpha + 10\beta + \gamma - (100\alpha' + 10\beta' + \gamma')$

  $= 100\alpha + 10\beta + \gamma - (100\alpha + 10\beta + d)$

⇒ **$p - p' = \gamma - d$ or $|p - p'| <= 5$**   using (1)

  for example, **let p = 212** where $\alpha = 2, \beta = 1, \gamma = 2$ and **let d = 3**

  since $|\gamma - d| = 1$, $p' = 100\alpha + 10\beta + d$ or $100(2) + 10(1) + 3$

                 p' = 213

  case 1a: if $100\alpha + 10\beta + d > 255$, put $\beta' = \beta - 1$

    $p - p' = 100\alpha + 10\beta + \gamma - (100\alpha + 10(\beta - 1) + d)$

    $p - p' = \gamma - d + 10$ (2)

    since $p' > 255$, $p - p' < 0$

⇒ $\gamma - d < 0$ or $\gamma - d \,\varepsilon\, \{-5, \ldots, -1\}$ ∵ $p' > 255$ on substituting γ by d   (3)

    ∴ $p - p' \,\varepsilon\, \{5, \ldots, 9\}$ using (2) and (3)

    for example, **let p = 251** where $\alpha = 2, \beta = 5, \gamma = 1$ and **let d = 6**

    $p' = 100(2) + 10(5) + 6$ or 256

    since $256 > 255$, **p' = 246** on decrementing β

case 2:   if $|\gamma - d| > 5$ & $\gamma - d < 0$

    since $|\gamma - d| > 5$, $\gamma - d > 5$ or $\gamma - d < -5$

⇒ $\gamma - d \,\varepsilon\, \{-9, -8, -7, -6\}$ ∵ $\gamma - d < 0$ (4)

  case 2a: if $\beta \,\varepsilon\, \{1, \ldots, 9\}$, put $\gamma' = d, \beta' = \beta - 1, \alpha' = \alpha$

    $p - p' = 100\alpha + 10\beta + \gamma - (100\alpha + 10(\beta - 1) + d)$

      $= \gamma - d + 10$ (5)

$\Rightarrow$ **p - p' ε {1, 2, 3, 4}**   using (4) and (5)

for example, **let p = 240** where $\alpha = 2$, $\beta = 4$, $\gamma = 0$ and **let d=9**

p' = 100(2) + 10(4 - 1) + 9 or **p' = 239**

case 2b: if β = 0 & α ≠ 0, put γ' = d, β' = 9, α' = α - 1

p - p' = $100\alpha + 10(0) + \gamma - (100(\alpha - 1) + 10(9) + d)$

p - p' = $\gamma - d + 10$ (6)

$\Rightarrow$ **p - p' ε {1, 2, 3, 4}**   using (4) and (6)

for example, **let p = 100** where $\alpha = 1$, $\beta = 0$, $\gamma = 0$ and **let d=8**

p' = 100(1 - 1) + 10(9) + 8 or **p' = 98**

case 2c: if β = 0 & α = 0, put γ' = d, β' = 0, α' = 0

p - p' = $100(0) + 10(0) + \gamma - (100(0) + 10(0) + d)$

p - p' = $\gamma - d$

$\Rightarrow$ **p - p' < 0 or p' > p by maximum 9**   using (4)

for example, **let p = 3** where $\alpha = 0$, $\beta = 0$, $\gamma = 3$ and **let d=9**

p' = 100(0) + 10(0) + 9 or **p' = 9**

case 3:   **if |γ - d| > 5 & γ - d > 0**

$\Rightarrow$ $\gamma - d > 5$ or $\gamma - d \,\varepsilon\, \{6, 7, 8, 9\}$ (7)

case 3a: if β ε {0, . . . ,8}, put γ' = d, β' = β + 1, α' = α

p - p' = $100\alpha + 10\beta + \gamma - (100\alpha + 10(\beta + 1) + d)$

p - p' = $\gamma - d - 10$ (8)

$\Rightarrow$ **p - p' ε {-4, -3, -2, -1}**   using (7) and (8)

for example, **let p = 129** where $\alpha = 1$, $\beta = 2$, $\gamma = 9$ and **let d=0**

p' = 100(1) + 10(2+1) + 0 or **p' = 130**

case 3b: if β = 9, put γ' = d, β' = 0, α' = α + 1

p - p' = $100\alpha + 10(9) + \gamma - (100(\alpha + 1) + 10(0) + d)$

p - p' = $\gamma - d - 10$ (9)

$\Rightarrow$ **p - p' ε {-4, -3, -2, -1}**   using (7) and (9)

for example, **let p = 99** where $\alpha = 0$, $\beta = 9$, $\gamma = 9$ and **let d=1**

p' = 100(0+1) + 10(0) + 1 or **p' = 101**

This shows that pixel values change by the values in range 0–4 or 0–5 in almost all cases. Change in a pixel value by maximum value 9 is rare, either in solving falloff boundary problem or a single-digit cover pixel value.

3.5 Data extraction

A digit is extracted from stego pixel using modulus 10 on the pixel value. To recover text or grayscale image, digits are extracted sequentially and grouped into two-digit values. Values (y) 00–95 are converted back to ASCII values using ⌈ (y*(ub - lb)/max + lb) ⌉ and 96–99 are converted to respective control characters to recover the text. Values 00–64 are used to recover the grayscale image. For color data image, planes are recovered, extracting pixels (00–64) from every two consecutive pixels of the corresponding stego plane.

4 Results and analysis

The proposed technique has used standard 512×512 color test images of JPG format as cover images obtained from the SIPI image database (http://sipi.usc.edu/database/database.php?volume=misc). Cover images are shown in Fig. 3 (a - h). Fig. 4 presents stego images with PSNR values resulting from embedding 393,216 bytes (3,145,728 bits) in the respective cover images using the proposed technique. PSNR measures imperceptibility in dB using Eq. (10) between cover C and stego S for XxY images.

$$PSNR = 10*\log_{10} \frac{X*Y*255*255}{\sum_{i=1}^{X} \sum_{j=1}^{Y} \left(C_{ij} - S_{ij} \right)^2} \tag{10}$$

QI (Quality Index), a measure of similarity is computed using Eq. (11), where C_m and S_m are the average pixel values.

Table 1 Results of existing LSB + PVD techniques

image 512x512x3	PVD + modulus [18]			seven-way + one-way PVD [15]				overlapping block-based PVD [16]		
	PSNR	capacity	bpp	PSNR	capacity	QI	bpp	PSNR	capacity	bpp
lena	37.13	810,757	1.546	41.73	1,901,149	.9993	2.41	31.01	1,976,671	2.51
baboon	34.95	918,877	1.753	33.77	2,243,218	.9957	2.85	32.29	2,219,715	2.82
pepper	36.74	812,986	1.551	40.20	1,806,166	.9992	2.29	30.10	1,783,210	2.27
jet	36.36	818,887	1.562	41.98	1,909,595	.9989	2.42	35.66	1,753,707	2.23
boat	36.03	851,837	1.625	37.91	1,991,005	.9988	2.53	33.11	2,130,772	2.71
house	36.57	834,866	1.592	38.98	1,977,403	.9986	2.51	34.59	2,079,088	2.64
Average	36.30	841,368	1.605	39.09	1,971,423	.9984	2.50	32.79	1,990,527	2.53

Table 2 Results of other LSB + PVD techniques

Image 512x512x3	MFPVD [24]				non-adaptive PVD [21]				five-way PVD with M-LSB type 2 [22]			
	PSNR	capacity	QI	bpp	PSNR	capacity	QI	bpp	PSNR	capacity	QI	bpp
Lena	44.16	2,027,335	.999	2.58	39.51	2,446,290	.999	3.11	39.58	2,451,046	.999	3.12
baboon	40.84	2,303,950	.999	2.93	35.31	2,685,772	.997	3.42	31.72	2,816,768	.993	3.58
pepper	43.19	2,027,680	.999	2.58	35.08	2,449,935	.997	3.12	35.45	2,459,762	.998	3.13
jet	44.19	2,033,855	.999	2.59	38.73	2,422,730	.998	3.08	39.73	2,454,060	.998	3.12
boat	42.62	2,105,840	.999	2.68	36.68	2,527,123	.999	3.21	34.16	2,565,007	.997	3.26
house	42.96	2,094,865	.999	2.66	36.63	2,482,279	.998	3.16	36.46	2,533,362	.997	3.22
Average	42.99	2,098,921	.999	2.67	36.99	2,502,355	.998	3.18	36.18	2,546,668	.997	3.24

Table 3 Results of the other and the proposed technique

image 512x512x3	eight directional PVD with M-LSB type 2 [22]				proposed technique			
	PSNR	capacity	QI	bpp	PSNR	capacity	QI	bpp
Lena	39.38	2,533,551	.999	3.22	38.82	3,145,728	.9965	4
baboon	31.23	2,939,376	.992	3.74	38.80	3,145,728	.9942	4
pepper	34.1	2,544,392	.997	3.24	38.36	3,145,728	.9948	4
jet	39.2	2,538,801	.998	3.23	38.85	3,145,728	.9466	4
boat	33.17	2,659,795	.997	3.38	38.80	3,145,728	.9894	4
house	36.66	2,625,804	.998	3.34	38.82	3,145,728	.9829	4
Average	35.62	2,640,286	.997	3.36	38.74	3,145,728	.9841	4

$$QI = \frac{4*C_m*S_m*\left\{\sum_{i=1}^{X}\sum_{j=1}^{Y}\left(C_{ij}-C_m\right)\left(S_{ij}-S_m\right)\right\}}{\left\{\sum_{i=1}^{X}\sum_{j=1}^{Y}\left(C_{ij}-C_m\right)^2 + \sum_{i=1}^{X}\sum_{j=1}^{Y}\left(S_{ij}-S_m\right)^2\right\}*\left\{C_m^2 + S_m^2\right\}} \tag{11}$$

The results of the existing LSB + PVD techniques and the proposed technique are presented in Table 1 to Table 3. In Table 2, for MFPVD [24], PSNR is provided for 840,000 bits. However, for the proposed technique, PSNR has been computed for 3,145,728 bits. The proposed technique has achieved relatively the higher capacity of 4bpp with considerably good PSNR of 38.74 dB (39 dB approximately). Comparison of the proposed technique with QVD techniques for a payload of 700,000 bits has been presented in Table 4. QVD [24] and the proposed technique has attained the same capacity of 4bpp. To compare the two, the median capacity for both the techniques has been computed as the median is more robust to outliers. The median capacity obtained for proposed and QVD technique [24] is 3,145,728 bits and 3,140,444 bits respectively. Therefore, both the mean capacity (as shown in Table 4) and the median capacity for the proposed technique is higher relative to [24].

The results of Tables 1, 2, 3 and 4 have been summarized in Figs. 5 and 6 respectively. Fig. 5 shows that the proposed technique has outperformed the existing LSB + PVD techniques in terms of capacity, attaining 4bpp. Additionally, PSNR of about 39 dB at 4bpp using the proposed technique is fairly high and comparable to that of MFPVD [24] which shows lower capacity. The proposed technique also attaining higher median and mean capacity has outperformed QVD [24] as shown in Fig. 6. However, QVD + LSB [25] has achieved the highest capacity of 4.55bpp in the available literature. Thus, to the best of our knowledge, the proposed technique has attained the second-best capacity (3,145,728 bits with 4bpp) in the available literature and the highest PSNR of 45.18 dB (for 700,000 bits). The trade-off (about 39 dB PSNR at 4bpp) has not been found in the available literature. The proposed digit hiding technique has outperformed the existing digit based hiding techniques also as shown in Table 5.

The efficiency of the proposed technique for different embedding rates of 20%, 40%, 60%, 80% and 100% over a set of cover images (Fig. 3) has been presented in Fig. 7. It shows approximate 46 dB PSNR at embedding rate of 0.2 (629,144 bits), approximate 43 dB at 0.4 rate (1,258,288 bits) and about 41 dB at rate of 0.6 (1,887,440 bits). However, PSNR decreases as the embedding rate increases, reaching about 38.74 dB

Table 4 Results of existing QVD techniques and the proposed technique for 700,000 bits

| image 512x512x3	QVD technique [24]				LSB + QVD technique [25]				proposed technique			
	PSNR	capacity	QI	bpp	PSNR	capacity	QI	bpp	PSNR	capacity	QI	bpp
Lena	37.85	3,137,551	.9985	3.99	33.16	3,573,202	.9956	4.54	45.45	3,145,728	.9994	4
baboon	33.19	3,178,789	.9950	4.04	32.41	3,615,390	.9941	4.59	45.34	3,145,728	.9990	4
tiffany	37.54	3,137,090	.9967	3.99	33.71	3,572,814	.992	4.54	44.40	3,145,728	.9977	4
pepper	35.34	3,140,279	.9978	3.99	33.21	3,576,879	.9965	4.55	45.05	3,145,728	.9991	4
jet	38.28	3,140,610	.9975	3.99	32.74	3,576,372	.9912	4.55	45.37	3,145,728	.9863	4
boat	35.04	3,148,861	.9978	4.01	33.14	3,585,039	.9966	4.56	45.34	3,145,728	.9981	4
house	36.57	3,142,681	.9976	3.99	32.94	3,578,488	.9947	4.55	45.39	3,145,728	.9958	4
pot	37.29	3,137,725	.9989	3.99	33.13	3,572,977	.9971	4.54	45.08	3,145,728	.9979	4
Average	36.39	3,145,448	.9975	4.00	33.06	3,581,395	.9947	4.55	45.18	3,145,728	.9967	4
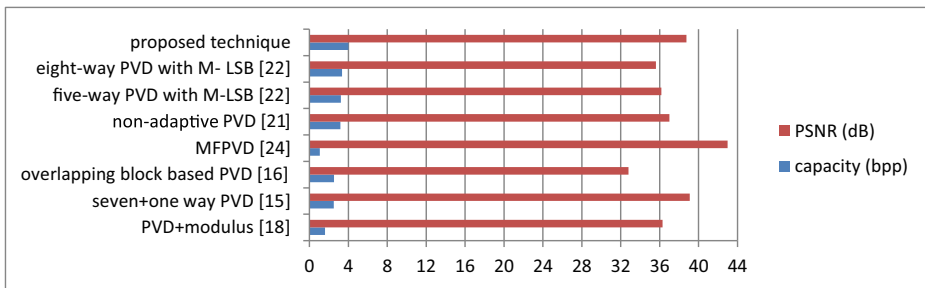
Fig. 5 comparison of capacity and PSNR for different techniques

over 100% rate (3,145,728 bits). At respective embedding rates, the proposed technique has resulted in considerably high PSNR.

The proposed technique has been extended to also hide an image in a color image. Fig. 8 (a1 - h1) presents data images: 512 × 512 grayscale and 256 × 512 color images. Table 6 presents the result of hiding these images in Lena and Baboon. It shows fairly good PSNR of about 40.6 dB and 38.8 dB on hiding grayscale and color images respectively. Figs. 9 and 10 show stego images with PSNR values on hiding grayscale (Fig. 8 (a1 - d1)) and color data images (Fig. 8 (e1 - h1)) respectively in Lena and Baboon.

For the proposed technique, time complexity depends upon the size n of secret data and embedding a data unit alters the pixel digits that takes O (1). Thus, the time complexity of O(n) is obtained. Table 7 compares the time complexity of the proposed technique with that of other techniques. Also, the embedding and extraction time for 700,000 bits is provided in the table.

5 Security analysis

To evaluate the security of the proposed method, modified Weighted Stego-Image (WS) steganalysis [2, 11] have been used. Fridrich introduced standard WS method [6] to estimate the number of random embedding changes, made using LSB steganography in
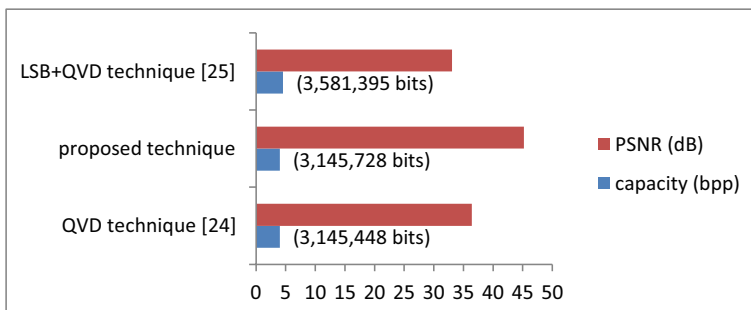

Fig. 6 Comparison of PSNR (for 700,000 bits) and maximum capacity achieved using different techniques

Table 5 Comparison of the proposed technique with existing digit hiding techniques

Image 512 × 512 × 3	Payload (bytes)	PSNR		
		PVM [14]	enhanced PVM modulus 10 [12]	proposed technique
Lena	83,654	45.0926	41.2968	45.6307
baboon	91,286	39.0633	40.8535	45.1470
pepper	78,612	42.0500	41.5641	45.5300
jet	81,851	43.9345	41.2968	45.6596

the spatial domain. It outperformed both RS and Sample Pairs Analysis (SPA) for nearly 100% embedding rate. Ker and Bohme [11] improved the standard method and special-ized standard WS to detect sequential embedding, outperforming structural detectors. Bohme further enhanced WS on a never-compressed and JPEG pre-compressed covers [4] and proposed WS variant [17] to detect content-adaptive embedding. Specialized WS to detect sequential embedding [11] was upgraded in [2]. Both the upgraded and specialized WS methods have been used for the security analysis of the proposed technique that also embeds data sequentially.

Suppose that a cover image C consists of n samples with n = XxY and C = (c_1, …, c_n). A payload of length q ≤ n is embedded by LSB substitution, resulting in a stego image S = (s_1, …, s_n). Let $\overline{S}$ be the stego image with every sample's LSB flipped using $\overline{s}_i = s_i + 1 - 2 (s_i \mod 2)$ and for z ε [0,1], let S^z be the weighted stego image formed using Eq. (12).

$$s_i^z = z\overline{s}_i + (1-z)s_i \qquad (12)$$

As per Theorem 1 [6], weighted stego image is closest to the cover when the difference between S^z and C is measured using Euclidean L^2 norm. Thus, embedding rate $r = \frac{q}{n}$ can be estimated as r' from the stego image using Eq. (13) that finds z which minimizes the distance between cover and weighted stego image.
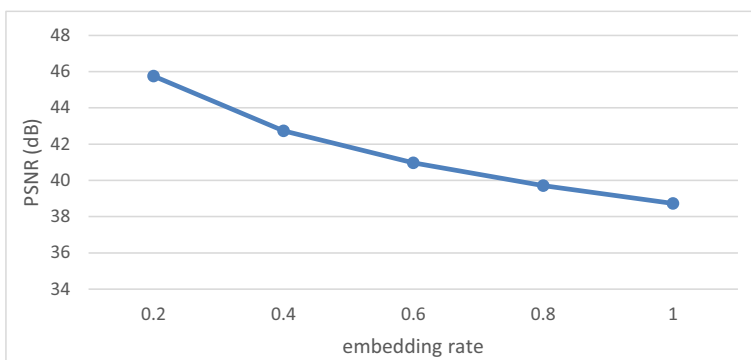

Fig. 7 Efficiency of the proposed technique over different embedding rates

Fig. 8 512 × 512 grayscale and 256 × 512 color data images

$$r' = \mathrm{arg}min_z E(z) = \sum_{i=1}^{n} \left(s_i^z - c_i\right)^2 \qquad (13)$$

However, the cover is unavailable for steganalysis, WS method estimates c_i using F(s_i), the function of the neighbors of s_i excluding s_i. Then the embedding rate is estimated using Eq. (14) which on differentiating, estimates r using Eq. (15). In [6], cover pixel predictor F(s_i) estimated c_i as a mean of the four closest neighbors of s_i. However, this predictor resulted in more accuracy in flat areas than noisy areas of the image. Thus, to improve performance, weights w_i for each stego pixel (relatively less weight for pixels in noisy areas) were introduced in expression (15) such that $_i$ $w_i = 1$. It yielded estimation Eq. (16) (unweighted expression (15) corresponds to each $w_i = 1/n$).

$$r' = \mathrm{arg}min_z E(z) = \sum_{i=1}^{n} \left(s_i^z - F(s_i)\right)^2 \qquad (14)$$

$$r' = \mathrm{arg}min_z E(z) = \frac{2}{n} \sum_{i=1}^{n} (s_i - F(s_i))\left(s_i - \overline{s_i}\right) \qquad (15)$$

$$r' = \mathrm{arg}min_z \sum_{i=1}^{n} w_i\left(s_i^z - F(s_i)\right)^2 = 2\sum_{i=1}^{n} w_i(s_i - F(s_i))\left(s_i - \overline{s_i}\right) \qquad (16)$$

To keep the variance of the estimate low, it was suggested that w_i be proportional to $1/(1 + \sigma_i^2)$, where σ_i^2 is the local variance of four neighboring pixels. Ker [11] further improved WS components to improve detection. An adaptive convolution filter F of

Table 6 PSNR on hiding grayscale and color data images in lena and baboon

cover image 512 × 512x3	PSNR							
	512x512 grayscale data image				256 × 512 color data image			
	pepper (a1)	jet (b1)	boat (c1)	house (d1)	pepper (e1)	jet (f1)	boat (g1)	house (h1)
lena	40.62	40.54	40.60	40.57	38.91	38.82	38.85	38.86
baboon	40.59	40.59	40.58	40.58	38.83	38.83	38.83	38.83
average	40.60	40.57	40.59	40.57	38.87	38.82	38.84	38.84

Fig. 9 stego images resulting from hiding grayscale images (Fig. 8 (a1 - d1)) in Lena and Baboon

the symmetrical form (17) given by (18) was proposed to enhance pixel prediction. Weights $w_i \propto 1/(5 + \sigma_i^2)$ was suggested over the one in [6] to reduce higher weights that over-emphasized the flat areas in image.

$$
\begin{matrix}
b & a & b \\
a & 0 & a \\
b & a & b
\end{matrix}
\tag{17}
$$

$$
\begin{matrix}
-1/4 & 1/2 & -1/4 \\
1/2 & 0 & 1/2 \\
-1/4 & 1/2 & -1/4
\end{matrix}
\tag{18}
$$

Also, specialized WS method to detect sequential embedding [11] was introduced that has been described ahead. For payload hidden in the first t samples, $z = 1/2$ is fixed and $z = 0$ for the rest. Using (12) and (14),

Fig. 10 stego images on hiding color images (Fig. 8 (e1 - h1)) in Lena and baboon

Table 7 Time complexity of different techniques

Techniques	Order of complexity	embedding time (sec)	extraction time (sec)
QVD [24]	O(n)	18	9
MFPVD [24]	O(n)	16	12
proposed technique	O(n)	21	0.13

$$E(t) = \sum_{i=1}^{t} \left(\frac{1}{2}(s_i + \overline{s_i}) - F(s_i) \right)^2 + \sum_{i=t+1}^{n} (s_i - F(s_i))^2 \qquad (19)$$

is minimized in expectation at t = q. However, its derivative has no closed-form and the function can have multiple minima. Thus, all values of t can be tried to locate the minimum. However, computing the sum (19) for each t = 0, …, n would need $O(n^2)$ operations. Therefore, noting that the linear recurrence

$$e_o = 0, e_t = e_{t-1} + \left(\frac{1}{2}\left(s_t + \overline{s_t}\right) - F(s_t) \right)^2 - \left(s_t - F\left(s_t\right) \right)^2$$

generates $e_t = E(t) - \sum_{i=1}^{n}(s_i - F(s_i))^2$; the minimum term e_t thus gives a minimum of E(t) and it takes linear time to generate and examine the sequence e_t. This linear time algorithm has been used in this paper to estimate the embedding rate r = q/n using two cover pixel predictors. As stated in [4, 11], WS steganalysis crucially depends on the accuracy of the pixel predictor. For the proposed method, dynamic pixel predictor that uses the Canny Edge Detection algorithm [2] and adaptive filter in expression (18) used in [11] have been used. Dynamic pixel predictor uses the Canny Edge Detection algorithm to separate the pixels as edge and non-edge pixels. For a target pixel that is an edge pixel, only those neighbors which are the edge pixels are used to predict c_i. For a non-edge target pixel, c_i is predicted using non-edge neighbors. The result showing the estimated rate using both the predictors over stego images with 100% embedding rate (r = 1) has been presented in Table 8. It shows that the adaptive filter has relatively failed to detect any message. For adaptive filter, estimated rate r' is negligible, and extensively lower for the dynamic pixel predictor relative to the true embedding rate r = 1. Fig. 11 compares Mean Absolute Error (MAE) in rate estimation for different embedding rates

Table 8 estimated embedding rate over the actual rate of 1.0 using two predictors

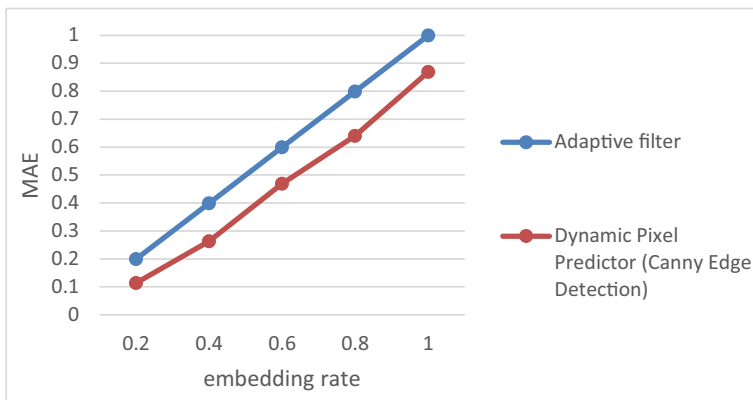| Images	adaptive filter	dynamic pixel predictor (Canny Edge Detection)
lena	0.0000	.0009
baboon	0.0045	.2079
pepper	0.0000	.0280
jet	0.0000	.0007
boat	0.0000	.3263
house	0.0008	.0126
barbara	0.0000	.1429
pot	0.0000	.3310

Fig. 11 Mean Absolute Error (MAE) over different embedding rates using two detection schemes of WS steganalysis

over a set of $N = 8$ stego images, where MAE $= 1/N$　|r'-r|. It shows that the dynamic pixel predictor gives better results than the adaptive filter in estimating r. However, for both the predictors, MAE is higher which increases with r. Thus, for the proposed technique, Weighted Stego-Image steganalysis has shown poor detection performance.

6 Conclusion

This paper has proposed a novel digit-based image steganography technique, unlike the common techniques which hide binary data. The technique has outperformed some existing spatial hiding techniques in terms of capacity and imperceptibility, achieving second-best capacity in the available literature. Indeed, the trade-off attained between payload and imperceptibility on 100% embedding rate (about 39 dB PSNR at 3,145,728 bits with 4bpp) has not been found in the available literature. Also, the method can hide any data (text, grayscale or a color image) without incurring falloff boundary problem. Moreover, the proposed linear time technique has secured data via encryption and against state-of-the-art steganalysis.

References

1. Anderson RJ, Petitcolas FAP (1998) On The Limits of Steganography. IEEE J Sel Areas Commun 16:474–481. https://doi.org/10.1109/49.668971
2. Anjum A, Islam S (2016) LSB Steganalysis Using Modified Weighted Stego-Image Method. In: 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, pp 630–635
3. Balakrishna C, NaveenChandra V, Pal R (2013) Image Steganography Using Generalized Base Single Digit Sum (GB-SDS). In: 2013 Annual IEEE India Conference (INDICON). pp 1–4

4. Bohme R (2010) Advanced Statistical Steganalysis
5. Faez K, Khodaei M (2012) New adaptive steganographic method using least-significant-bit substitution and pixel-value differencing. IET Image Process 6:677–686. https://doi.org/10.1049/iet-ipr.2011.0059
6. Fridrich J, Goljan M (2004) On estimation of secret message length in LSB steganography in spatial domain. In: Proc. SPIE 5306, Security, Steganography, and Watermarking of Multimedia Contents VI
7. Hussain M, Wahab AWA, Bin Idris YI et al (2018) Image steganography in spatial domain: A survey. Signal Process Image Commun 65:46–66. https://doi.org/10.1016/j.image.2018.03.012
8. Hussain M, Wahid A, Ho ATS, Javed N (2017) A data hiding scheme using parity-bit pixel value differencing and improved rightmost digit replacement. Signal Process Image Commun 50:44–57. https://doi.org/10.1016/j.image.2016.10.005
9. Hussain M, Wahid A, Javed N, Jung K (2016) Hybrid Data Hiding Scheme Using Right-Most Digit Replacement and Adaptive Least Significant Bit for Digital Images. Symmetry (Basel) 8. https://doi.org/10.3390/sym8060041
10. Jung KH, Ha KJ, Yoo KY (2008) Image Data Hiding Method Based on Multi-pixel Differencing and LSB Substitution Methods. In: 2008 International Conference on Convergence and Hybrid Information Technology. IEEE, pp 355–358
11. Ker AD, Bohme R (2008) Revisiting Weighted Stego-Image Steganalysis. In: Proc. SPIE 6819, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X
12. Laffont A, Maniriho P, Ramsi A, et al (2017) Enhanced Pixel Value Modification based on Modulus Function for RGB Image Steganography. In: 2017 11th International Conference on Information & Communication Technology and System (ICTS). IEEE, pp 61–66
13. Li S, Zhang X (2019) Toward Construction Based Data Hiding: From Secrets to Fingerprint Images. IEEE Trans Image Process 28:1482–1497. https://doi.org/10.1109/TIP.2018.2878290
14. Nagaraj V, Vijayalakshmi V, Zayaraz G (2013) Color Image Steganography based on Pixel Value Modification Method Using Modulus Function. In: IERI Procedia-2013 International Conference on Electronic Engineering and Computer Science. pp 17–24
15. Pradhan A, Raja Sekhar K, Swain G (2016) Digital Image Steganography based on Seven Way Pixel Value Differencing. Indian J Sci Technol 9. https://doi.org/10.17485/ijst/2016/v9i37/88557
16. Prasad S, Pal AK (2017) An RGB colour image steganography scheme using overlapping block-based pixel-value differencing. R Soc Open Sci 4. https://doi.org/10.1098/rsos.161066
17. Sch P, Korff S, Rainer B, Notation A (2012) Weighted Stego-Image Steganalysis for Naive Content-Adaptive Embedding. In: 2012 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, pp 193–198
18. Shen S, Huang L, Tian Q (2015) A novel data hiding for color images based on pixel value difference and modulus function. Multimed Tools Appl 74:707–728. https://doi.org/10.1007/s11042-014-2016-0
19. Sudeep P V., Navas KA (2009) A Novel Data hiding Method in Spatial Domain. In: 10th National Conference on Technological Trends (NCTT 2009)
20. Swain G (2014) Digital Image Steganography using Nine-Pixel Differencing and Modified LSB Substitution. Indian J Sci Technol 7:1444–1450
21. Swain G (2018) Adaptive and Non-adaptive PVD Steganography Using Overlapped Pixel Blocks. Arab J Sci Eng 43:7549–7562. https://doi.org/10.1007/s13369-018-3163-9
22. Swain G (2018) High capacity image steganography using modified LSB substitution and PVD against pixel difference histogram analysis. Secur Commun Networks. https://doi.org/10.1155/2018/1505896
23. Swain G (2018) Digital image steganography using eight directional PVD against RS analysis and PDH analysis. Adv Multimed. https://doi.org/10.1155/2018/4847098
24. Swain G (2019) Two new steganography techniques based on quotient value differencing with addition-subtraction logic and PVD with modulus function. Opt - Int J Light Electron Opt 180:807–823. https://doi.org/10.1016/j.ijleo.2018.11.015
25. Swain G (2019) Very High Capacity Image Steganography Technique Using Quotient Value Differencing and LSB Substitution. Arab J Sci Eng 44:2995–3004. https://doi.org/10.1007/s13369-018-3372-2
26. Tao J, Li S, Zhang X, Wang Z (2019) Towards Robust Image Steganography. IEEE Trans Circuits Syst Video Technol 29:594–600. https://doi.org/10.1109/TCSVT.2018.2881118
27. Thanikaiselvan V, Selvaraj S, Rengarajan A (2014) PVD Based Steganography on Scrambled RGB Cover Images with Pixel Indicator. J Artif Intell 7:54–68. https://doi.org/10.3923/jai.2014.54.68
28. Thien C, Lin J (2003) A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function. Pattern Recogn 36:2875–2881. https://doi.org/10.1016/S0031-3203(03)00221-8

Vinita Verma received the B.Sc. (Hons) degree and M.Sc. degree in Computer Science from University of Delhi (DU), India, in 2015 and 2017, respectively. She is currently a Ph.D. Scholar (Information Security) in the Department of Computer Science, DU. Her research interests include image processing, data hiding, and information security.

Sunil K. Muttoo has done B.Sc. Gp 'A' in 1974, M.Sc. (Mathematics) in 1976, M.Phil. Mathematics in 1978, Ph.D. (Coding Theory) in 1982 from DU and MTech. (Computer Science and Data Processing) in 1990 from IIT Kharagpur, India. He has more than ten years of teaching experience in DU colleges at Under Graduate Level. Since 1991 to date, he has more than twenty-five years of teaching experience in the Department of Computer Science, DU at Post Graduate and Doctorate Level. He was the Head of the Department of Computer Science from 1999 to 2002. He is a life member of the Cryptographic Society of India and works in the field of Coding Theory, Cryptography, Steganography, and Digital Watermarking.

V. B. Singh received the B.Sc. degree and M.C.A. degree from the College, UP, India, in 1992 and 1996, respectively. He received his Ph.D. degree in Software Reliability from DU in 2008. He has more than 17 years of teaching experience. He was a Lecturer and Sr. Lecturer from 1996 to 2007, Reader from 2007 to 2010 and has been an Associate Professor since 2010 in Delhi College of Arts & Commerce, DU. He is a reviewer for more than five academic journals and Editorial Board member for two computer science journals. He is a member of the Big Data Initiative, Department of Science and Technology, Govt. of India. His research interests include Mining Software Repositories, Empirical Software Engineering, and Software Reliability Engineering.