A Reversible Data Hiding Technique Using Histogram Modification and Smvq for Very Large Payloads

R. LEELAVATHI, M. N. GIRI PRASAD

Abstract*: *Reversible data hiding (RDH), unlike traditional data hiding techniques, focuses on recovering the cover image after the secret data is extracted from it at the receiver. The prime essence of the concept is that the cover image should be recovered without any damage or with imperceptible error in the pixel values. In this paper, we propose an RDH technique based on simple histogram modification and Side Match Vector Quantization (SMVQ). The proposed histogram modification technique attains high PSNR values with impressive payload capacities. The SMVQ technique further increases the payload capacity by four folds. The damage in the retrieved message is imperceptible and the recovered cover media is either exactly same or has a very high PSNR value. The algorithm has been well tested and compared with various other techniques.

Keywords*: *Reversible Data hiding, Histogram modification, Side Match Vector Quantization, payload, secret information, cover image.

I. INTRODUCTION

Digital watermarks are secret codes added to a digital audio, video, image [1] or document file to provide authentication. The embedding of the digital watermarks is carried out by a process known as data hiding [2]. Over the years, researchers have proposed numerous techniques of data hiding. One way of broadly classifying the techniques is into reversible and irreversible data hiding techniques. Irreversible data hiding is commonly known as steganography [3], where only the secret message is extracted. In contrast, reversible data hiding [4] techniques recover both the embedded secret message and the cover data. Thus the application scope for these techniques is increasing in the current era of digital expansion, where huge data is transmitted every second. There are some highly useful applications of digital watermarking. The first one is user authentication and proving the ownership. This would help prevent pirated distribution of the media content without the consent of the owner. In case of false ownership claims of the file, the digital extraction process can identify the true owner. It has been proved that the owners could also control distribution of their file by inserting code that prevents copying but this is not a hundred percent effective.

Revised Manuscript Received on April 18, 2019.

R. Leelavathi, Assistant Professor, Vasavi College of Engineering, Hyderabad, India
M. N. Giri Prasad, Professor, JNTU College of Engineering, Anantapuram, A. P India

The second way that digital watermarking is used is for data security and the goal here is to certify and authenticate a file. The third use of digital watermarking [5] is forgery and tampering detection and the goal here is to detect any modifications made to a file.

The reversible data hiding techniques were researched extensively in the early 90's and 2000's. The algorithms used can be categorized into difference expansion schemes, prediction based schemes and histogram modification schemes. In the difference expansion based methods, the difference value between two pixels is used to embed the secret information. The difference value is multiplied by 2 or its LSB is replaced [6]. The location map of the min and max pixels, i.e. 0 and 255 is needed to prevent overflow and underflow. In the prediction based techniques, instead of the difference data embedding is done based in the error in prediction value. This method increases the payload by 2 folds in comparison to the previous method. The third is the histogram shifting scheme where the peaks in the histogram are modified to hide the data. Here the payload is directly proportional to the size of the first two peaks in the histogram.

II. LITERATURE REVIEW

In this effort [7], researchers popularized the reversible data hiding technique for natural portraits adopting the block-level prediction-error inflation. This technique confidential information is nested as a 2x2 portrait blocks by utilizing the picture element excess within each block. The reversible data hiding technique with the ciphered portraits utilizing the adaptive block-level prediction-error growth (ABPEE-RDHEI). High nesting rate can be achieved through this method. The amusing visual feature of the decided decoded portraits can be gained.

The writers projected a unique reversible data hiding technique [8] for portrait variation enhancement. The projected technique has been applied for two sets of portraits and distinguished with previous techniques. The proposed technique is capable of recovering the primary portrait completely without any errors. The results obtained in the practical condition say that difference can be reduced and RDH can be achieved. The mathematical analysis performed on the resultant images prove that the algorithm has recovered the cover data comletely.

This exertion explains about [9] the performance of the high capacity RDH strategy positioned on right-left shift. On can reduce the distortion of the considered portrait by using the technique proposed in this paper, for doing so the researchers improved the traditional technique proposed by the Wang et al. The multiple summit bins are shifted towards the zero bins, this shifting happens towards the right and left paths, mutually. By examining and determining the required circumstances for the conjecturing error, one can easily determine the picture elements experienced the overflow/underflow during the process of embedding and deriving. The overflow/underflow issues can be resolved by using a simple mathematical technique of addition and subtraction. The complexity of time can be reduced with this exertion.

The writers projected [10] a reversible data hiding strategy in an encrypted discipline which will strongly broadcast the media data through cloud planning and demonstrated its rightful ownership. This proposal works on the basis of Chinese Remainder Theorem (CRT) based partitioning scheme that administers the data present in the media into various encrypted shares and inserts the private information into the encrypted allotments through a secret key. These allotments can be secure to the incidentally scattered cloud information centers.

This work proposed [11] anRDH process for vector quantization (VQ) - condensed portraits. The protected [12], huge capacity and reversible information hiding pattern for electronic health care utilization are been proposed through this method. The developers retrieved the conventional inauguration technique for the purpose of hiding the portrait generation by pixel to Block (PTB) alteration approach. This process has been determined mathematically productive compared to conventional interpolation methods. The proposal has been figure out for affective imperceptibility and destroys determining strength by exposing it to different portrait transforming and mathematical errors. High-quality efficient portraits are achieved through this process.

The developers implemented [13] a versatile reversible data hiding method for encoded portraits. This model uses the additive modulo 256 for the purpose of encoding and mean property is applied for achieving the upper limit installing in the portrait encoded. This scheme generates superior values of PSNR, correlation coefficient, and SSIM for the precisely decoded portrait as distinguished to current methods. Computational efficiency is high in this method.

The developers projected [14] a technique through which quantized coefficient value of the inserted confidential information will become zero. The projected technique inserts the information by altering the steady all-zero coefficient continuance in individual inserted blocks. Initially, the embedding technique chooses various positions from the continuance path with respect to the portion of the confidential information. Later, the projected technique changes the zero coefficients in these selected portions by non-zero elements with respect to the leftover portion of the confidential information. Through this approach, one can get the maximum ratio among the increased file size and payload. As compared to before models embedding capacity is high.

This method uses [15] median edge discovery predictor for getting the efficient quantization values. The authors also used a substitute prediction approach for increasing the prediction accuracy by restricting the level of prediction values. The bit-rate is reduced through the centralized error diversion approach.

The writers popularized [16] a unique and easy reversible data hiding (RDH) strategy for installing a huge amount of data in JPEJ portraits. In this process, non-zero coefficient values are converted into a subtitle way for installing the binary message torrent, through which the installing capacity gets increased.

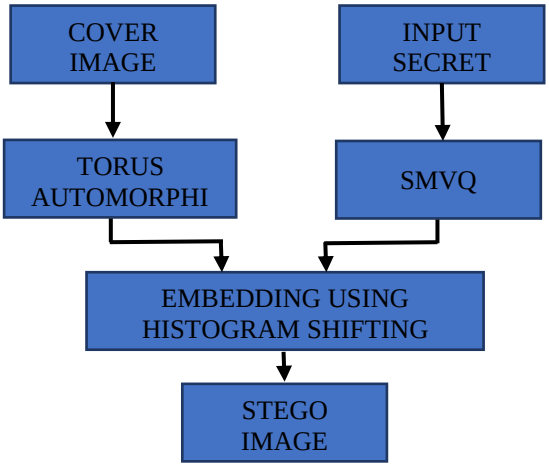


Figure 1: Proposed work flow

The novel algorithm proposed [17] in this paper is as follows. First the input secret image is compressed using SMVQ. Then the using bit decomposition the binary form of the message is obtained. Then torus automorphism selects the pixel locations to be embedded. The binary secret data is embedded using histogram modification, thus producing the stego image. The detailed explanation is given in the following sections.

III. TORUS AUTOMORPHISM

Torus automorphism is an algorithm used for encrypting data for secure transmission. It is similar to image shuffling algorithms like Arnold Transform. In this paper, we use the algorithm on the secret image to be hidden. The image pixels are transferred to different locations based on the following equation:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} mod\ N$$

Here,

(x,y) are the original coordinates of the pixels in the image

(X, Y) are the transformed coordinates

N is the size of the image

A is a 2x2 matrix used for the transformation of the data.

The matrix A has a determinant equal to 1 and has an eigen vector λ which can be described as:

$$\lambda = \frac{k + \sqrt{k^2 - 4}}{2}$$

k is an integer greater than 2.

For example, consider a 4x4 matrix to be transformed as shown below

$$\begin{bmatrix} 11 & 12 & 13 & 14 \\ 21 & 22 & 23 & 24 \\ 31 & 32 & 33 & 34 \\ 41 & 42 & 43 & 44 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$$

Then the transformed coordinates after applying the equationis as follows

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} mod\ 4$$

$$\begin{bmatrix} 44 & 13 & 12 & 21 \\ 22 & 41 & 14 & 13 \\ 14 & 23 & 42 & 11 \\ 12 & 11 & 24 & 43 \end{bmatrix}$$

When applied on an image, the resultant image becomes encrypted.

IV. SOLID MATCH VECTOR QUANTIZATION

Algorithm 1: Code block construction

1. Divide the input image into blocks of size 4x4.
2. Convert the 4x4 blocks into vectors of size 1x16.
3. Select the top 256 vectors with maximum Euclidean distance and arrange them into a matrix of size 256x16.

Algorithm 2: Block Vector Quantization

Inputs:

a. Code book described in algorithm 1.
b. 4x4 image block
c. A mask representing the pixel values of the block to be processed.

Output:

a. Sorted row indices of the code block.

Steps:

1. Extract the pixel values form the 4x4 image block based in the indices present in the mask.
2. Subtract each row of the code block with the output obtained in step 1.
3. Sort the code block's row indices based on the obtained absolute differences.

Algorithm 3: SMVQ

1. Read the image
2. Initialize the parameters – height and width of the image, block length, number of blocks.
3. Divide the image into equally sized blocks. For instance, a 256x256 image will be divided into 64x64 blocks (denoted by D), given the block size is 4x4. Let the block image be named as BImage.

a) Input Image

b) Block Image (*BImage*)

Figure 2: Block division

4. Load the code block presented in Algorithm 1.
5. Create a row vector names "Mask" with elements from 1 to 16.
6. Apply Block Vector Quantization algorithm as mentioned in **Algorithm 2** for the first row and column of the *BImage*. The inputs to the algorithm are:
   a. Code block generated in Algorithm 1.
   b. Each blocks from the 1st row and column of *BImage*.
   c. Mask generated in step 5.

The resultant compressed image after step 6 is shown in the figure 3.

Figure 3: the resultant image after step 6.

7. For the remaining blocks in the *BImage*, which start with an index (2, 2) to (64, 64), extract the upper and left blocks from the *BImage*.
8. Create a new mask with pixels in the bottom row of the upper block and the right most row of the left block, as depicted in the figure 4.

Upper Block

			Ub1	Ub5	Ub9	Ub13
			Ub2	Ub6	Ub10	Ub14
			Ub3	Ub7	Ub11	Ub15
			Ub4	Ub8	Ub12	Ub16
Lb1	Lb5	Lb9	Lb13	Bock being processed		
Lb2	Lb6	Lb10	Lb14			
Lb3	Lb7	Lb11	Lb15			
Lb4	Lb8	Lb12	Lb16			

Left Block

Figure 4: Left and upper block representation

9. Construct a 4x4 pixel block as shown in the figure 5. Let the name of the block be "*blockval*".

Lb13	Ub8	Ub12	Ub16
Lb14	0	0	0
Lb15	0	0	0
Lb16	0	0	0

Figure 5: 4x4 pixel block named "*blockval*"

10. Create a row vector '*MaskVQ'* with the elements 1, 2, 3, 4, 5, 9 and 13. This mask is used to extract the pixels in the first row and the first column of the block as shown in the figure 6.
11.

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

Figure 6: Mask description for a 4x4 pixel block.

12. Apply Block Vector Quantization algorithm as mentioned in **Algorithm 2** with the inputs to the algorithm as:
    a. Code block generated in Algorithm 1.
    b. *blockval* from step 9.
    c. *MaskVQ* from step 10.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

D(1,1)	D(1,2)	D(1,3)	.	.	D(1,63)	D(1,64)
D(2,1)						
D(3,1)						
D(4,1)						
.						
.						
D(63,1)						
D(64,1)						

13. Based on the extracted indices in step 11, compare the current 4x4 block with the rows of the code book and find the best match.
14. Assign the corresponding row value to the compressed image. The resultant compressed image is depicted in the figure 7.

Figure 7: compressed Image output of SMVQ algorithm

Algorithm 4: **De-SMVQ**

1. Reconstruct the first row and column directly from the index values retrieved form the compressed image. Gather the pixels from the row index of the code book and replace them in the output image.

2. For the remaining blocks, repeat the steps 8-11 from algorithm 4. Based on the extracted indices, the corresponding rows from the codebook are retrieved and replaced in the reconstructed image. The output is depicted in the figure 8.

Figure 8: Reconstructed Image.

V. DATA HIDING

The process of hiding the data into cover image by modifying the histogram of the image is histogram based reversible data hiding. This process first nullifies the boundary overflow and underflow conditions. Overflow is the phenomena where the pixel value goes above 255 and the underflow is its counterpart where the value ties to go below 0. Then it identifies the maximum value in the histogram and shifts the values based on the data to be hidden. Histogram based reversible data hiding method was introduced by Ni et al. in [2], where message is embedded within the histogram.

The proposed novel technique is a modified version of traditional data hiding schemes. In our proposed method, the histogram is modified according to the data bit that is to be embedded. The torus automorphism decides the pixel bit to be embedded. The process is divided into two stages:

Pre-processing stage

The image pixel values are modified according to their position in the image. The details of the modification process are listed below:

- The even positioned pixel should have even value.
- The odd positioned pixel should have odd value.

A simple demonstration is depicted in the figure 9 with a 2x2 pixel matrix.

36	137
89	205

a) Pixel Matrix

1	3
2	4

b) Pixel position values

37	137
90	206

c) Processed Image

Figure 9: Image pre-processing demonstration.

The pixels with the positions 1, 2 and 4 are incremented by 1 as the positions and values does not match the even odd combination. The pixel value is position 3 aligns with the algorithm thus remains unchanged.

The resultant images are visually intact and the change is imperceptible. The PSNR values and the output images are displayed in figures 10 to 17 and table 1 to 4.

Once the pre-processing stage is complete, the data bits to be hidden are embedded into the image according the algorithm described.

Embedding stage

The embedding stage is as follows.

Step 1: If the data bit to be embedded is 1, the pixel value in the processed image is incremented by 1.

Step 2: If the data bit to be embedded is 0, the pixel value remains intact.

The above mentioned steps 1 and 2 are repeated for every data bit that is to be hidden. This process distributes the payload uniformly throughout the image, increasing the embedding capacity compared to other conventional histogram based technique.

Data Extraction

The extraction process is the reverse of the embedding process.

Step 1: Scan every pixel in the image.

Step 2: If the pixel position is odd, and the pixel value is even, the data bit embedded is 1.

Step 3: If the pixel position is even and the pixel value is odd, the data bit embedded is 1.

Step 4: If the pixel position and the pixel value both are either even or odd respectively, then the data bit to be extracted is 0.

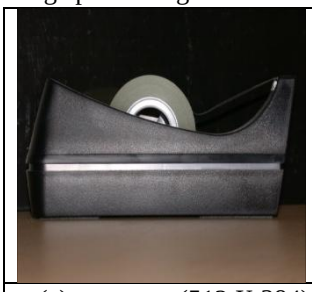
Retrieving the Cover Image:

During the data extraction process, the cover image can be retrieved back by changing the pixel values back to the pre-processed image. Based on the recovered data and the pixel positions, one can obtain the pre-processed image.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

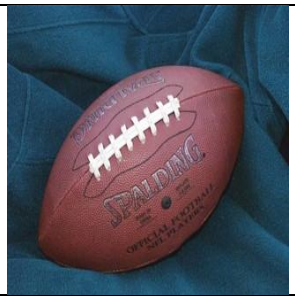
VI. EXPERIMENTAL ANALYSIS

The following images have been taken from MATLAB image processing tool box.



(a). tape.png (512 X 384)

(b). Peppers.png (512 X 384)



(c). football.jpg (320 X 256)

(d). fabric.png (640 X 480)

(f). greens.jpg (500 X 300)

(g). gantrycrane.png (400 X 264)

(h). concordaerial.png (3060 X 2036)

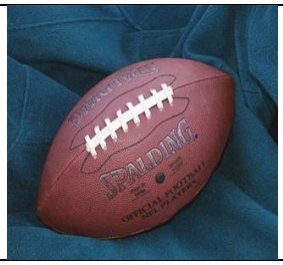
Figure 10. The figures (a) – (h) are the input cover images used to test the proposed algorithm.

Table 1. Payload in Procedure 1.

S. No	Image name	Payload (bits)
1.	tape.png	5,89,824
2.	Peppers	5,89,824
3.	Football	2,45,760
4.	Fabric	9,21,600
5.	Greens	4,50,000
6.	Gantrycrane	3,16,800
7.	concordaerial	1,86,90,480

(a). tape

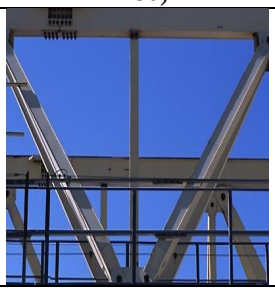
(b). Peppers



(c). football.jpg (320 X 256)

(d). fabric.png (640 X 480)

(f). greens.jpg (500 X 300)



(g). gantrycrane.png (400 X 264)

(h). concordaerial.png (3060 X 2036)

Figure 11: the above figures (a) – (h) are the output of pre processing stage. The PSNR values of the images are tabulated below.

Table 2. Pre-processed Images

S. No	Image name	PSNR
1.	tape.png	51.1832
2.	Peppers	51.2765
3.	Football	51.1943
4.	Fabric	51.2342
5.	Greens	51.1730
6.	gantrycrane	51.1896
7.	concordaerial	51.1802

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

SMVQ based compression – the original image "pears.png" was resized to 256 x 256 x 3 and given as an input to SMVQ. The resultant image was of size 64 x 64 x 3. The no of bits to be embedded was 98304, which can be accommodated by an image of size 128 x 256 x 3.

Figure 12. Input secret image for data hiding

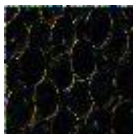
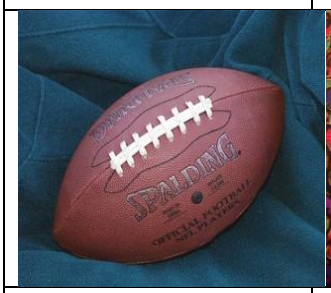


Figure 13. SMVQ output

The output of SMVQ is embedded into the cover images shown in figure 13. The resulting images are shown below.

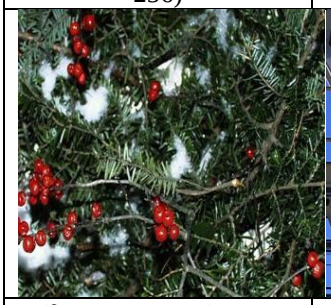
(a). tape

(b). Peppers



(c). football.jpg (320 X 256)

(d). fabric.png (640 X 480)



(f). greens.jpg (500 X 300)

(g). gantrycrane.png (400 X 264)

(h). concordaerial.png (3060 X 2036)

Figure 14. The figures (a)-(h) represent the output stego image with secret data hiding.

Table 3. PSNR values of proposed system stego images

S. No	Image name	PSNR
1.	tape.png	57.5025
2.	Peppers	57.5025
3.	Football	53.7003
4.	Fabric	59.4409
5.	Greens	56.3564
6.	gantrycrane	54.8476
7.	concordaerial	72.5114

The extracted secret image and the recovered cover images are shown below.

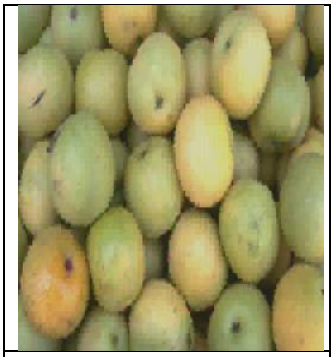


Figure 15. Extracted secret image (256x256)

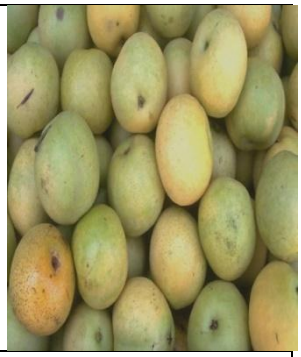
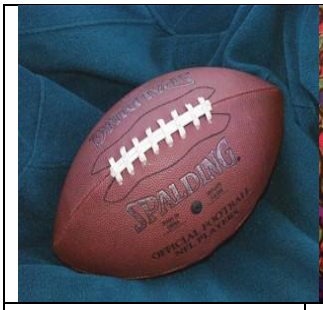


Figure 16. Extracted secret image (1024x1024)

(a). tape

(b). Peppers



(c). football.jpg (320 X 256)

(d). fabric.png (640 X 480)

(f). greens.jpg (500 X 300)

(g). gantrycrane.png (400 X 264)

(h). concordaerial.png (3060 X 2036)

Figure 17. Recovered cover images

Table 4. PSNR values of recovered cover images

S. No	Image name	PSNR (w.r.t original Image)	PSNR (w.r.t processed Image)
1.	tape.png	51.1832	inf
2.	Peppers	51.2765	inf
3.	Football	51.1943	inf
4.	Fabric	51.2342	inf
5.	greens	51.1730	inf
6.	gantrycrane	51.1896	inf
7.	concordaerial	51.1802	inf

Image	Jo and Kim's Scheme		Chang and Wu's scheme		Chin and Wei's method	
	Payload	PSNR	Payload	PSNR	Payload	PSNR
Lena	14930	28.19	13487	29.25	16129	30.78
Peppers	14992	28.74	13984	29.07	16129	30.18
Baboon	12462	21.54	8794	22.43	16129	22.66

S. No.	Image	Proposed Method	
		Payload	PSNR
1	Lena	786432	51.2356
2	Peppers	786432	51.2236
3	Baboon	786432	51.1258

VII. CONCLUSION

In this research, we proposed an RDH technique based on simple histogram modification and SMVQ for larger payloads. The proposed histogram modification technique attains high PSNR values with impressive payload capacities. The SMVQ technique further increases the payload capacity by four folds. The damage in the extracted secret message is imperceptible and the recovered cover image is either exactly same or has a very high PSNR value.

REFERENCES

1. G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data: A state-of-the-art overview," IEEE Signal Processing Mag., vol. 17, no. 5, pp. 20–46, Sept. 2000.
2. M. Wu, H. Yu, and B. Liu, "Data hiding in image and video: Part IIdesigns and applications," IEEE Trans. Image Process., vol. 12, no. 6, pp. 696–705, Jun. 2003.
3. K. Bailey, K. Curran, "An Evaluation of Image Based Steganography Methods", Multimedia Tools & Applications, Vol. 30, No. 1, pages 55-88, July 2006.
4. X. Li, W. Zhang, X. Gui, and B. Yang, "Efficient reversible data hiding based on multiple histograms modification," IEEE Trans. Inf. Forensics Security, vol. 10, no. 9, pp. 2016–2027, Sep. 2015
5. Kumar, Sanjay, and Ambar Dutta. "Performance analysis of spatial domain digital watermarking techniques." Information Communication and Embedded Systems (ICICES), 2016 International Conference on. IEEE, 2016
6. Bhatt S, Ray A, Ghosh A, Ray A (2015) Image steganography and visible watermarking using lsb extraction technique. In: 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO). IEEE, pp 1–6
7. Shuang Yi, Yicong Zhou and Zhongyun Hua, "Reversible Data Hiding in Encrypted Images using Adaptive Block-Level Prediction-Error Expansion", Journal of Signal Processing: Image Communication, pp. 1-26, 2018.
8. Wu H T, Tang S, Huang J and Shi Y Q 2018 A novel reversible data hiding method with image contrast enhancement Signal Processing: Image Communication 62 64-73
9. W. Wang, J. Ye, T. Wang, W. Wang, "A high capacity reversible data hiding scheme based on right-left shift", Signal Processing, vol. 150, pp. 102-115, 2018.
10. P. Singh, B. Raman, "Reversible data hiding for rightful ownership assertion of images in encrypted domain over cloud", AEU-International Journal of Electronics and Communications, vol. 76, pp. 18-35, 2017.

11. Rahmani P, Dastghailbyfard G (2018) An efficient histogram-based index mapping mechanism for reversible data hiding in VQ-compressed images. Inf Sci 435:224–239
12. S.A. Parah, J.A. Sheikh, F. Ahad, G.M. BhatHiding clinical information in medical images: a new high capacity and reversible dat hiding technique J. Biomed. Inf., 66 (2017), pp. 214-230
13. S. Agrawal and . M. Kumar, "Mean value based reversible data hiding in encrypted images," Optik - International Journal for Light and Electron Optics, vol. 130, p. 922–934, 2017.
14. Chang JC., Lee YH., Wu HL. (2018) Compression-Efficient Reversible Data Hiding in Zero Quantized Coefficients of JPEG Images. In: Pan JS., Tsai PW., Watada J., Jain L. (eds) Advances in Intelligent Information Hiding and Multimedia Signal Processing. IIH-MSP 2017. Smart Innovation, Systems and Technologies, vol 81. Springer, Cham.
15. Hong W, Ma YB, Wu HC et al (2017) An efficient reversible data hiding method for AMBTC compressed images. Mult Tools Appl 76(4):5441–5460
16. Y. Liu, C.-C. Chang, Reversible data hiding for JPEG images employing all quantized non-zero AC coefficients, Displays 51 (2018) 51–56.
17. Z. Pan and L. Wang, "Novel reversible data hiding scheme for Two-stage VQ compressed images based on search-order coding," Journal of Visual Communication and Image Representation, vol. 50, pp. 186–198, 2018.