International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8
***Article Received:** 25 May 2023 **Revised:** 12 June 2023 **Accepted:** 30 July 2023*

Reversible Data Hiding Using Histogram Shifting for Digital Images

Namitha R Shetty[1*], Yogish Naik G R2, Vidyasagar K B[3]
[1]Department of PG Studies and Research in Computer Science Kuvempu University Shimoga, India
namithashetty19999@gmail.com
[2]Department of PG Studies and Research in Computer Science Kuvempu University Shimoga, India
ynaik.ku@gmail.com
[3]Department of PG Studies and Research in Computer Science Kuvempu University Shimoga India
vkb2231@gmail.com

Abstract
Reversible Data Hiding (RDH) includes embedding information into cover media in a way that takes into consideration the recovery of the inserted information and in addition the original media. In our daily lives, JPEG is the most popular digital image format. Reversible data concealing technology has been frequently applied to JPEG images in recent years for specialized uses including management of file and for authentication of image. Moreover, because JPEG images have lesser information redundancy than uncompressed images, RDH in JPEG is far more difficult than in uncompressed images and any modifications made to the compressed domain could cause the host image to become even more distorted. In this study, we give some fundamental ideas to choose quantized DCT coefficients for RDH taking into account the statistical properties of DCT coefficients and theory behind the JPEG encoder. One of the most widely used methods for developing reversible data concealment technologies is histogram shifting. A new approach for JPEG images, histogram shifting-based RDH technique is put forth, zero coefficients are not extended; only coefficient values of +1 and -1 are used to convey message bits. In this research, we generate DCT coefficients for data hiding in an adaptive way by using a block selection strategy based on the number of zero coefficients in each 8x8 block. According to the experimental results, the recommended approach achieves a higher peak signal-to-noise ratio and has a smaller file size increase when compared to state-of-the-art approaches.

Keywords: *Reversible Data Hiding, Histogram Shifting, Discrete Cosine Transform, Variable-Length Code, Least Significant Bit, Prediction Error.*

1. INTRODUCTION

The evolution of the internet and computer science has revolutionized(transformed) how information is created, accessed and shared. Presenting chances as well as difficulties, especially with regard to defending intellectual property rights in digital media. A technique known as "reversible data hiding" allows for the precise restoration of the original content while permitting the retrieval of concealed data from digital content such as images without causing any permanent alteration [1]. To maintain reversibility, the ability to precisely rebuild the original host image following the extraction of hidden data is the major objective of RDH. This is achieved by designing extraction algorithms that retrieve the concealed information without loss from the modified image. Through lossless embedding methods like prediction error expansion or histogram modification, data is hidden within the host signal while maintaining its integrity. Prediction error expansion involves modifying the prediction errors obtained during compression or encoding processes, thereby creating space to embed additional data without altering the original content. Histogram modification [2] techniques subtly modify the pixel values distribution in images to encode extra information. Bit-plane manipulation operates on the binary representation of the signal, altering less significant bits to embed data

without significantly affecting the higher-order bits. This process ensures that the original content remains unchanged upon extraction of the concealed data. Reversible data hiding finds applications in steganography for covert communication and data authentication. It balances trade-offs between embedding capacity and signal distortion, making it valuable for secure data transmission where information concealment and preservation of original content are crucial. Continual advancements in reversible data hiding methods aim to improve capacity, security, and applicability across diverse digital media.

The JPEG standard is among the popular and ancient digital format of images utilized in everyday life. JPEG image compression is used by the majority of modern media broadcasting companies and digital devices to save information as form of graphics. The realistic and useful study field for image privacy, image authentication, and image archive management is reversible data concealment in the compression of JPEG domain.

The majority of RDH algorithms are designed for uncompressed images, and recent years have seen notable advancements in the visual quality and embedding capability. JPEG images cannot be applied directly with them [3]. First off, because a JPEG image is in a compression image format, it has a lot less redundancy than an uncompressed image. As compared to the spatial domain, the DCT domain adjustment may generate noticeably greater contortion due to the de-quantization process in second. Third, we must take into account both the visual quality potential increase in file size during the process of embedding when evaluating RDH in a JPEG image. RDH in a JPEG is therefore far more challenging than an uncompressed image.

RDH in JPEG images has drawn more attention lately. These days, though researchers are focusing on concealment of data in compressed images of JPEG. The discrete cosine transform (DCT), one of the fundamental components of JPEG compression serves as the foundation for JPEG image compression [4, 5]. The most crucial feature of DCT for JPEG compression is quantizing the DCT coefficients utilizing visually weighted values of quantization table. Huffman code mapping [6, 7] is employed to incorporate the payload is added to bit-stream of JPEG. By applying a map from the unutilized codes to utilized codes in order to employ the variable-length code (VLC) that was not being used for AC coefficients. The VLC-based data concealing approach for lossless images of JPEG was enhanced by Hu et al. [8]. They describe a lossless data hiding strategy that embeds data straight into the JPEG picture bit stream and is based on underutilised variable-length codes found in the Huffman table. Their approach [8] is a better version of this type of approach [6]. A obstruct, non-compressed reversible data concealment technique was introduced by Chang et al. [9] in DCT based images to conceal payload. Two consecutive zero coefficients of the mid frequency elements from every blocks are utilised for embedding.

The structure of the paper is as follows: A discussion of RDH in images is covered in Section 2. In Section 3, the suggested RDH method for JPEG images is displayed. The next part contains the findings of the experiment. The paper is concluded in part 5.

2. RELATED WORK

Ben He et al. [10] proposes a RDH strategy based on two-dimensional histogram shifting to enhance the annotated JPEG images' file size and visual quality. In the proposed histogram shifting method, only the pairs of coefficients having two non-zero DCT coefficients which are quantized are changed for data embedding. In particular, the remainders allow for the embedding of data, and Coefficient pairings that possess a minimum of one quantized discrete cosine transform coefficient with values of -1 or + 1 are shifted. The efficiency of annotated JPEG images is enhanced by reducing the amount of invalid shifting pixels by the implementation of suggested reversible data hiding strategy.

Huang et al. [11] proposed some of fundamental understandings of the process of choosing quantized discrete cosine transform (DCT) coefficients for RDH, based on the statistical properties of DCT coefficients and the mathematics underpinning the JPEG encoder. A histogram shifting RDH strategy for JPEG pictures is put forth, wherein message bits are carried by extending only coefficients with values of +1 and -1, while zero coefficients remain unchanged. Furthermore, an adaptive method for selecting DCT coefficients for data concealment is presented, that is determined by how many zero coefficients are there in each 8x8 block. Results from experiments show that suggested strategy improves better embedding capability.

Wedaj et al. [12] proposes a new coefficient selection approach for data concealing in images of JPEG. This scheme utilizes histogram shifting (HS) approach to embed data. Based on the quantity of zero AC coefficients, at first block ordering is utilised to embed data in blocks, resulting in less contortion. After the AC coefficients valued +1 and −1 are utilised for embedding, the last non-zero AC coefficients are transferred left or right by their sign. The suggested approach has a excessive PSNR and a inconsiderable file size, according to the testing results.

Hang Cheng et al. [13] paper used probabilistic and homomorphic techniques to offer a combination of data concealing, no data loss RDH algorithms for ciphertext graphics. The cover pixels are encountered directly by encryption and decryption and it avoids pixel division

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8
***Article Received:** 25 May 2023 **Revised:** 12 June 2023 **Accepted:** 30 July 2023*

and by this it reduces the computational complexity. Because of the probabilistic property, an encrypted image based on direct decryption can still yield the original plaintext image in the lossless scheme even when the data in encrypted image have altered for embedding. In the meanwhile, the encrypted domain allows for the extraction of embedded data. Data embedding and the reversible scheme can be utilized all at once in an encrypted image. Utilizing the merged method, a recipient can retrieve the original plaintext image and before decryption a portion of data can be extracted and another portion of the embedded data after decryption.

Zhang X et al. [14] proposed a method using multiple histogram shifting for RDH that uses rate and distortion optimization. The optimization of rate and distortion is the formulation used to address the HS-based multiple embedding problem. To expedite the faster computation two keys are derived to reduce the solution space. A genetic algorithm, also known as an evolutionary enhancing algorithm, is utilized to find the almost maximum zero and peak bins. In comparison to other methods, this scheme offers better quality of image and capacity in terms of embedding.

VijayKumar et al. [15] proposed an enhanced method based on histogram shifting for RDH utilizing digital images that retrieve the original image and accurately extract the concealed data. Two of the greatest peak values are chosen for the image histogram when data is hidden. Repeating this embedding process multiple times results in a larger embedding capacity. To achieve a good visual perception a controlled contrast enhancement is used. Numerous attacks on the host image, including rotation, scaling, shearing, and impulse noise, are taken into consideration to confirm the suggested method robustness. The experimental results show this method outperforms better in terms of PSNR, SSIM, and embedding rate.

Chen et al. [16] proposed a method to embed secret data that uses a magic matrix for dual-image data hiding technique. This method uses a cross magic matrix and histogram shifting modification for reversible dual image embedding. Despite the fact that the capacity is the same as what was actually achieved, our results demonstrate a high capacity and good PSNR. Dual image hiding using reversibility for two-phase method is proposed. Phase 1 uses a histogram-modification-shifting technique to embed data on pixel differences, and Phase 2 hides secret digits on the plane with differences in its four quadrants using a cross magic matrix. This technique can maintain the embedding capacity at a high level while also greatly enhancing the image quality as determined by the PSNR. The findings show a high capacity and good PSNR, although the capacity is the same as what was actually achieved.

Aziz F et al. [17] proposed an innovative method for reversibly hiding data is presented, capitalizing on the existence of patterns and textures within a picture. Rather than calculating the adjacent pixels difference, the suggested algorithm looks for patterns in picture that could have a high embedding capacity. The framework enhances the visual quality of the marked image while embedding over twice as much data for texture images as the current difference-based method can. This framework, however, is easily adaptable to any other reversible data hiding methods that take advantage of an image's smooth region to boost their embedding capacity. Additionally, in certain circumstances, rearranging an image's rows can also boost the embedding capacity. They have used both column-wise and row-wise transformations to embed the data.

Pankaj Garg et al. [18] proposed a RDH scheme that takes into account the variations in other pixels in two color channels while also utilizing independent RGB color channels has been proposed and put into practice. Data is embedded in color images using two ways: first, separately into each color channel and interconnection between every two color channels in second, particularly in the variations among the pixels in every one of these channels. By ensuring accurate picture recovery and providing the image's authenticity and integrity of data through a variety of methods, the suggested scheme successfully accomplishes its objective. Additionally, more channels can be used by embedding in the difference of all possible pairs of color channels, as opposed to simply the three pairs used in the recommended technique.

3. PRELIMINARIES

In this section compression standard of JPEG Images and frameworks for JPEG images using RDH is discussed.

3.1. Compression Standard of JPEG Image

The first worldwide standard for digital picture compression was created by the Joint Photographic Experts Group (JPEG) [4, 5]. The old JPEG compression standard has a high compression ratio with little visual quality degradation. Fig. 1 represents the steps required for compression of JPEG images. The

DCT, quantizer, and entropy encoder are the three primary components of the JPEG encoder. Two-dimensional DCT is used to convert the original data of the image to the frequency domain from the spatial domain for non-overlapping 8x8 blocks. After that, the quantizer receives the obtained DCT coefficients and quantizes them utilizing the preset quantization table. A zigzag scanning order is produced by pre-compressing coefficients of DCT which are quantized utilizing differential pulse code modulation (DPCM) for DC coefficients and run length encoding (RLE) for AC coefficients. At last, the final compressed bitstream is obtained by Huffman coding the symbol string. Once the header is previewed, the final JPEG file is produced.

3.2. Framework for JPEG Images RDH

In contrast to the pixel-based approach, JPEG RDH approach embeds in DCT coefficients which is quantized. RDH in DC coefficients of JPEG consists of following steps. In step 1, a non compressed approach that Fridrich et al. [19] proposed. It preserves the embedding area to image's expendable component by compressing. This approach has not garnered as much attention as it could because of the small message capacity. Wang et al. [20]-improved quantization table modification technique that was first put forth by Fridrich et al. [19] in step 2. In order to make space for data concealment, their methods involve processing the coefficients of DCT earlier and changing the quantized table.

Despite the fact that the experimental results of [20] produce a high PSNR, the file size increases dramatically. Huffman table modification is handled in step 3. This method maps a utilized VLC to an unutilized VLC in order to achieve data embedding. Although this method was enhanced by Qian and Zhang [6], these methods have relatively small payload sizes. The coefficients of DCT which are quantized are modified in step 4. The quantized DCT coefficient histogram was shifted by Xuan et al. [21] using an ideal searching technique. This ideal approach made it possible for the technology to work well. To make data embedding unnoticeable and the visual quality of the marked image better, only beneath and intermediate frequency coefficients of DCT are chosen to embed the data when a particular number of data is embedded. A reversible data concealing strategy on images of JPEG was proposed by Li et al. [22] and was based on three minor adjustments to the quantization table and the lower DCT value selection method. High capacity reversible data concealing for JPEG images was covered by Lin et al. [23]. They achieved a high embedding capacity by utilising varying block sizes. Additionally, no data loss data hiding in the JPEG DCT coefficients' least significant bit was suggested by Celik et al. [24]. High distortion DCT values are augmented into the payload as auxiliary data. A method of histogram shifting on AC coefficients was presented by Huang et al. [25]. Based on the statistical characteristics of the number of zero AC coefficients in the blocks, the method employs ordering of blocks. The approach is very new and greatly improves upon earlier research.

4. PROPOSED SCHEME

The proposed JPEG reversible data concealment technique is explained in this section. The embedding of AC coefficients and block selection strategy will be explained. Then describes the procedure of choosing the locations of AC coefficients for minimising the distortion, which is the primary contribution of this work. Next, describes in brief how the payload extraction and retrieval of the DCT coefficients were accomplished. Fig. 1 represents the steps required for compression of JPEG images.

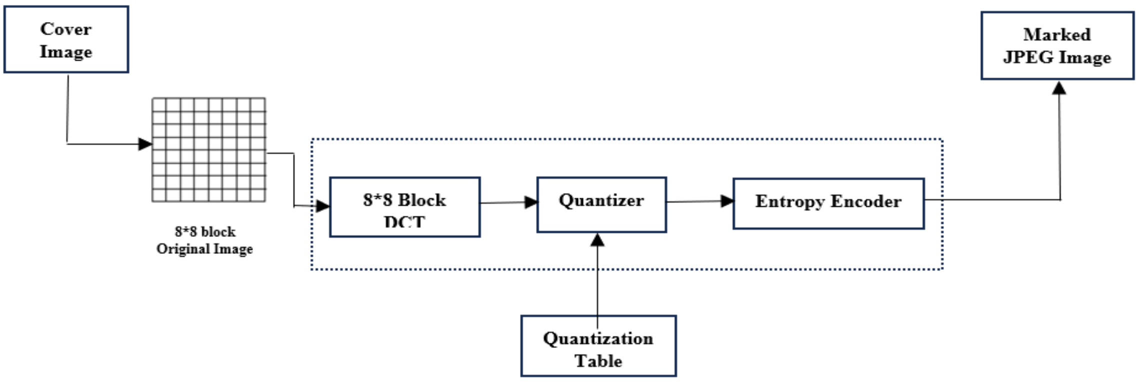


Fig. 1. Structure of JPEG Image

4.1. Data Embedding and Block Selection

The first coefficient in a quantized coefficient block is called the coefficient of DCT, and the last sixty-three are known the AC coefficients. The top corner left values in Fig. 2. are the DC coefficients, and they are grey level of the block in an average. The other 63 coefficients in the above figure are called AC Coefficients (alternating-current terms). So the payload is embedded only in AC coefficients. The following process, which also provides the watermarked coefficient of AC C, is utilised to embed the payload bit B ∈ 0, 1 for every coefficient of AC C.

$$\bar{C} = \begin{cases} C + B, \ if \ C = 1 \\ C - B, \ if \ C = -1 \\ C + 1, \quad if \ C > 1 \\ C - 1, \quad if \ C < -1 \\ C, \quad else \end{cases} \tag{1}$$

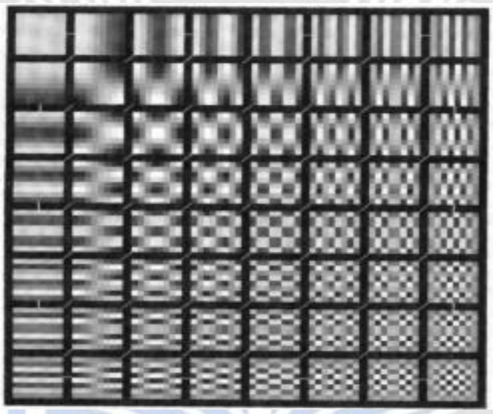


Fig. 2. Visualization of 64 cosine frequencies of a DCT.

If and only if C is either +1 or −1, then bit B is embedded. C is moved by +1 or -1, accordingly, depending on whether it is greater than or less than +1. Observe that, contrary to what was stated at the beginning of this section, the zero coefficients are not changed. In Histogram shifting AC coefficients with +1 or -1 for embeddable coefficients and the same is 0 for unchangeable coefficients and AC coefficients that are less than −1 or more than +1 are shiftable coefficients.

The ordering of blocks based on the quantity of zero AC coefficients was first proposed by Huang et al. [25]. According to the experimental findings, blocks having maximum zero AC coefficients are probably also high in AC coefficients values -1 or +1. [25] derived embedding solely in +1 and −1, and arranged the blocks in an embedding order such that the blocks with the greatest quantity of zero coefficients of AC are embedded at the top. This approach decreases distortion. Modifying a zero AC coefficient results in an increase in file size because each time a zero coefficient of AC is changed to non-zero, an additional sign must be coded. Consequently, this approach reduces distortion and file size compared to current techniques that incorporate zero AC coefficients. Similar block ordering circumstances are employed in the proposed scheme prior to the AC coefficients selection stage. A block having zero coefficients will present at top with the greatest embedding capability, whereas a block with fewer zero coefficients will present at bottom with the smallest embedding priority.

4.2. Selection of Coefficients

It has been established that the block ordering technique put out by Huang et al. [25] works well to reduce distortion, but they are not considering the additional two points. The distribution of embeddable, moveable, and unvarying coefficients varies for every position of the AC coefficients across all blocks at first. Some AC positions have a higher proportion of embeddable than shiftable coefficients. The entire amount of moveable and embeddable coefficients of every block at each point is displayed in Fig. 3. The statistics retrieved from the images with QF value of 70, are shown in Fig. 3a. and 3b. For instance, Fig. 3a. demonstrates that there are more embeddable AC coefficients at places like 24 than moveable coefficients of AC. Position 5 has more shiftable AC coefficients than embeddable coefficients. For embedding the 24th position is preferable. Second, changing the coefficient of AC that are closest to the coefficient of DC may create lesser contortion than altering the final one due to the quantization table's non-uniform structure. This is because modifying cost for each place is not uniform. For these reasons, we propose a procedure for choosing the embedding areas of AC coefficients.

The embedding capability as well as distortion are taken into account when selecting the positions. The quantity of embeddable AC coefficients is utilized to calculate the embedding capacity. The quantization table, PSNR function, and number of shiftable AC coefficients are used to model the distortion.

While the PSNR function is the most widely accepted way to measure distortion, this is greatest model for assessing distorted perception. The penalty for modification is the square of the deviance i.e, f(x) = x2.

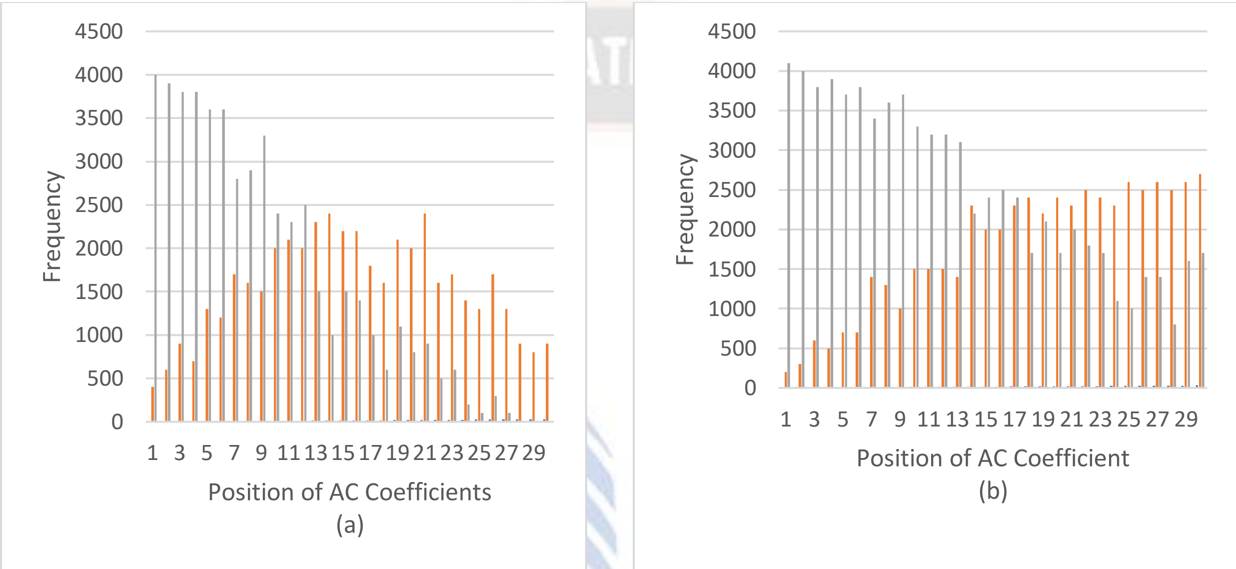


Fig. 3. The number of embeddable and shiftable at each AC coefficient location at QF=70

Compression ratio can be found in the quantization table as a important information. For every DCT coefficient block, the quantized table entry specifies compression level. This is as a result of the division of every DCT coefficient by the matching quantization entry. Consequently, at the pixel reconstruction step, the square of quantization entry of the quantized coefficient determines how the quantized coefficient is modified. An AC coefficient histogram can be simply modified using the histogram shifting approach. But in order to calculate RDH in JPEG images, along with the visual quality and embedding capability, we also need to consider how much storage the final marked image will have. Based on the research above, if fewer zero coefficients have changed during the process of embedding, the size of the image file can go up a lot.

The suggested approach must detect positions by taking into account the quantization table entry and the amount of embeddable and moveable coefficients of AC in order to determine which positions cause the least amount of distortion. The suggested approach determines an embedding efficiency Xi metric for every position i between 1, 2,..., 63:

$$Xi = \sum_{n=1}^{4096} \frac{E_{(i,n)}}{\left(Y_{(i,n)} + \frac{E_{(i,n)}}{2}\right) * Z_i^{2}} \qquad (2)$$

where E(i, n) ∈ {0, 1} denotes the embeddability (1) or not of the coefficients of AC at location i in the nth block (0). In parallel, Y(i, n) denotes the shiftability (1) or not (0) of the coefficient of AC at location i in the nth block. The entry in the quantization table at location i is Zi.

The nominator of the equation, when employing coefficients of AC in location I for embedding, reflects the entire embedding capability, and the denominator, corresponds to predicted distortion. Addition of all moveable coefficients of AC, part of the embedding capability, and the squared quantization term of the penalty. To examine its impact on the de-quantization step, the quantization table item has been squared. After data embedding, an image may be severely deformed even though there are many embeddable coefficients

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8
***Article Received:** 25 May 2023 **Revised:** 12 June 2023 **Accepted:** 30 July 2023*

---

and few shiftable coefficients at a particular point if the entry of quantized table is huge. To clearly display the variation between locations with respect to the value of X, this approach doubled the results from the quantized table.

Consequently, out of two sites with comparable distortion (denominator result from equation (2) with no doubling Zi), the site with the value of lesser Zi will be chosen for embedding. As a result, the highest embedding efficiency Xi value will belong to the best embedding site, while the lowest value will belong to the worst. When using Equation (2), a picture with a size of 512 × 512 is taken into account, resulting in 4096 number blocks.

After calculating the embedding efficiency for each of the 63 places, the numbers can be arranged from maximum to least. Subsequently, we must ensure that sufficient locations are chosen to enable the payload to be inserted. In order to do this, take the minimum number of places from the highest position of the listing to ensure each selected position's embedding capacity is equal to or greater than the payload. After the places are selected, each block can then be embedded in the sequence that was previously determined via block ordering. The authors advise to guarantee that the decoder and encoder utilise a similar sequence for the locations, embedding should be done from the smallest to the topmost location. For example, if locations (5, 1, 6, 9) are selected from the embedding, which then every single block should start with an embed in location 1, followed by 5 further, then next 6 and at last 9, and for removal, start at position 1 and work your way down to position 5, next 6, and at last 9.

The locations of the embedded coefficients of AC and payload length needs to be embedded in the coefficients of DC for suggested technique to be fully reversible. The greatest payload length that can be transferred is log2(W × H), whereby W and H stand for the width and height of the image. For images with a dimension of 512 × 512, this is 18 bits. The positions of the coefficients of AC that are utilized for embedding can be expressed through a binary matrix of size 63 bits; that is, if locations (5, 1, 6, 9) are selected from embedding. The bits located at vector indexes 1, 5, 6, and 9 will then have the following values: 1 and 0, respectively. As a result, the first 81 LSBs of the DC values can contain the 81 bits of side information required for image of 512 × 512. They are attached as part of the payload prior to embedding in order to allow the LSB of the initial DC values to be perfectly recovered.

4.3. Extraction and Image Recovery

While using RDH concealing approach, it is necessary for extracting the payload and recover the initial image precisely. Recovery and extraction are carried out concurrently. Obtaining the quantized DCT coefficients comes first. To determine the capability of payload and the locations of the coefficients of AC that were utilized for process of embedding, the 81 LSBs of the DC coefficients are read. Next, the blocks' embedding order is determined by block ordering. Since the coefficients of AC with zero values does not alter, the embedding's same ordering is attained. Equation (3) recovers the original AC coefficients based on the area of the employed coefficients of AC. Equation (4) is utilized to retrieve the payload and the initial LSBs of the 81 coefficients of DC. Once the primary LSBs of the DC coefficients are retrieved, an exact restoration of the actual JPEG image is achieved.

$$C = \begin{cases} \bar{C} - 1, & if\ \bar{C} > 1 \\ \bar{C} + 1, & if\ \bar{C} < -1 \\ \bar{C}, & else \end{cases} \tag{3}$$

$$B = \begin{cases} 0, & if\ |\bar{C}| = 1 \\ 1, & if\ |\bar{C}| = 2 \end{cases} \tag{4}$$

4.4. Encoder and Decoder

The encoder and decoder overview is detailed in this section. The encoder has following steps: 1. Blocks are ordered based on quantity of AC coefficients of value zero. 2. For every 63 potential locations, determine the entire quantity of moveable, embeddable, and fixed AC coefficients. 3. Determine which places will result in the least distortion when embedding the payload by calculating the efficiency of embedding of each position. 4. Utilising embedding efficiency Ri, sort the positions 5. The payload is embedded 6. Using the side information, reconstruct the initial DC coefficients' LSBs.

The steps for Decoder consists of: 1. To determine the capability of payload and the locations of the coefficients of AC utilised for process of embedding, read LSBs of first 81 DC coefficients. 2. Blocks are sorted the quantity of AC coefficients of value zero. 3. Payload is extracted and get the initial coefficients of AC back. 4. Restore LSBs with original value of DCT coefficient.

5. RESULTS

RDH method is usually evaluated using two performance evaluation metrics which are visual quality and file size preservation. This section compares suggested approach to three cutting-edge approaches: the approaches by Wedaj et al. [12], and Huang et al. [25].

To better highlight the performance, we offer scenarios for 2 different approaches. Ours, the two other, and the one above and compare them to one another in the coherent condition for both mentioned above.

For the purpose of studies, six 512 * 512 colour images (Lena, Boat, slash, Peppers, Baboon and F-16) are obtained from USC-SIPI database. Prior to being transformed into JPEG files with values of QF = 70 and QF = 80, these colour photos are first divided into six grayscale images.

The following describes related analysis and comparisons.

5.1. Visual quality

This section evaluates annotated pictures with specified embedding payloads for quality utilizing PSNR. Fig. 4. illustrates link between PSNRs and embedded payloads on the six designated JPEG images. The vertical axis shows PSNR, whereas embedding capability is shown by the axis that is horizontal.

As illustrated in Fig. 4. the PSNRs for the three RDH techniques fall as the embedding payload increases. The suggested approach PSNRs are higher than the other two techniques, even if the PSNRs of the other methods drop under the identical embedding payload circumstance.

It can be concluded as:

1. Compared to other approaches with the similar embedding capability, the suggested approach has greater PSNR.
2. The PSNRs of the suggested technique are close to those in Ben et al. [10]. as the embedding capacity increases. On the other hand, this article's solution employs a greater number of blocks but modifies the coefficients within every block less.

In the first and second tables, Table I and Table II we present the PSNRs of six samples that were captured by the three approaches (2500 bits and 4096 bits) as a means to clarify the results. Our approaches PSNRs are greater compared to that of the other schemes, as the first and second tables. Table I and Table II demonstrate, despite our system having less embedded bits. Selection of block is similar to doing not selecting at all when more and more secret data are implanted, hence the quantity of secret embedded bits rises, the PSNRs of the approach described in this paper approach those of the other approaches. However, as compared to the other two methods, the coefficient pairing method utilized here has a lower embedding capability. The outcomes of this experiment demonstrate the effectiveness of the suggested strategy, which takes advantage of the 2D RDH: first, On the 2D JPEG domain, we use the pairwise method to choose comparatively smooth areas for block selecting and process of embedding. And then, we eliminate all coefficients of AC that contain 0. The reason for this is coding an additional symbol is required anytime a coefficient of AC of value zero is changed to a not a zero, increasing the file size.

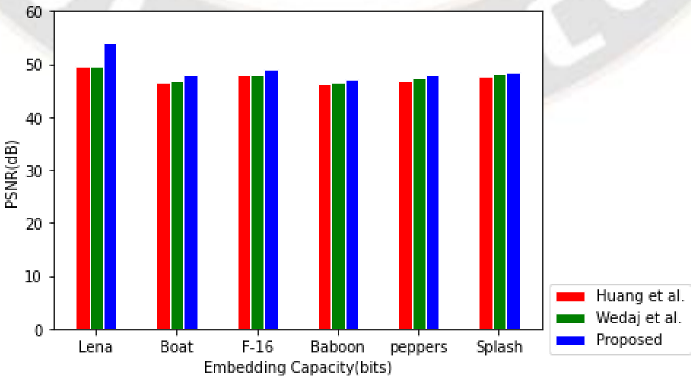


Fig. 4. PSNR and embedding capacity for six test images

Additionally, in Huang et al.[25] PSNRs are higher in some photographs than in the other three methods, such as Lena and Aircraft F-16 in Fig. 4. This could be because of the textural features. Notably, unlike Ben et al.[10], our suggested approach by 2D HS approach and only the coefficients of AC -1 and +1 are chosen for embedding. The rise in PSNRs in certain photos is therefore not of statistical significance.

Table I. Comparison of PSNR (dB) when the embedded bits are 2500 bits.

Image	Methods		
	Huang et al.[25]	Wedaj et al.[12]	Proposed
Lena	49.58	49.55	53.98
Boat	46.64	46.98	47.97
F-16	47.85	48.06	48.96
Baboon	46.19	46.47	47.12
Peppers	46.93	47.34	47.91
Splash	47.74	48.29	48.63

PSNR: peak signal-to-noise ratio.

Table II. Comparison of PSNR (dB) when the embedded bits are 4096 bits.

Image	Methods		
	Huang et al. [25]	Wedaj et al. [12]	Proposed
Lena	49.58	49.55	53.98
Boat	46.64	46.98	47.97
F-16	47.85	48.06	48.96
Baboon	46.19	46.47	47.12
Peppers	46.93	47.34	47.91
Splash	47.74	48.29	48.63

PSNR: peak signal-to-noise ratio.

5.2. File size increment

The increase in file size is one of the significant comparative indices for RDH. Well known RDH schemes for JPEG images aim to minimise storage size increases while maintaining embedding capacity. When QF = 70 and QF = 80, Table III shows each test image's increase in size compared to the initial image with various embedding capabilities. Table III shows that all of the file sizes grow in each of the three approaches.

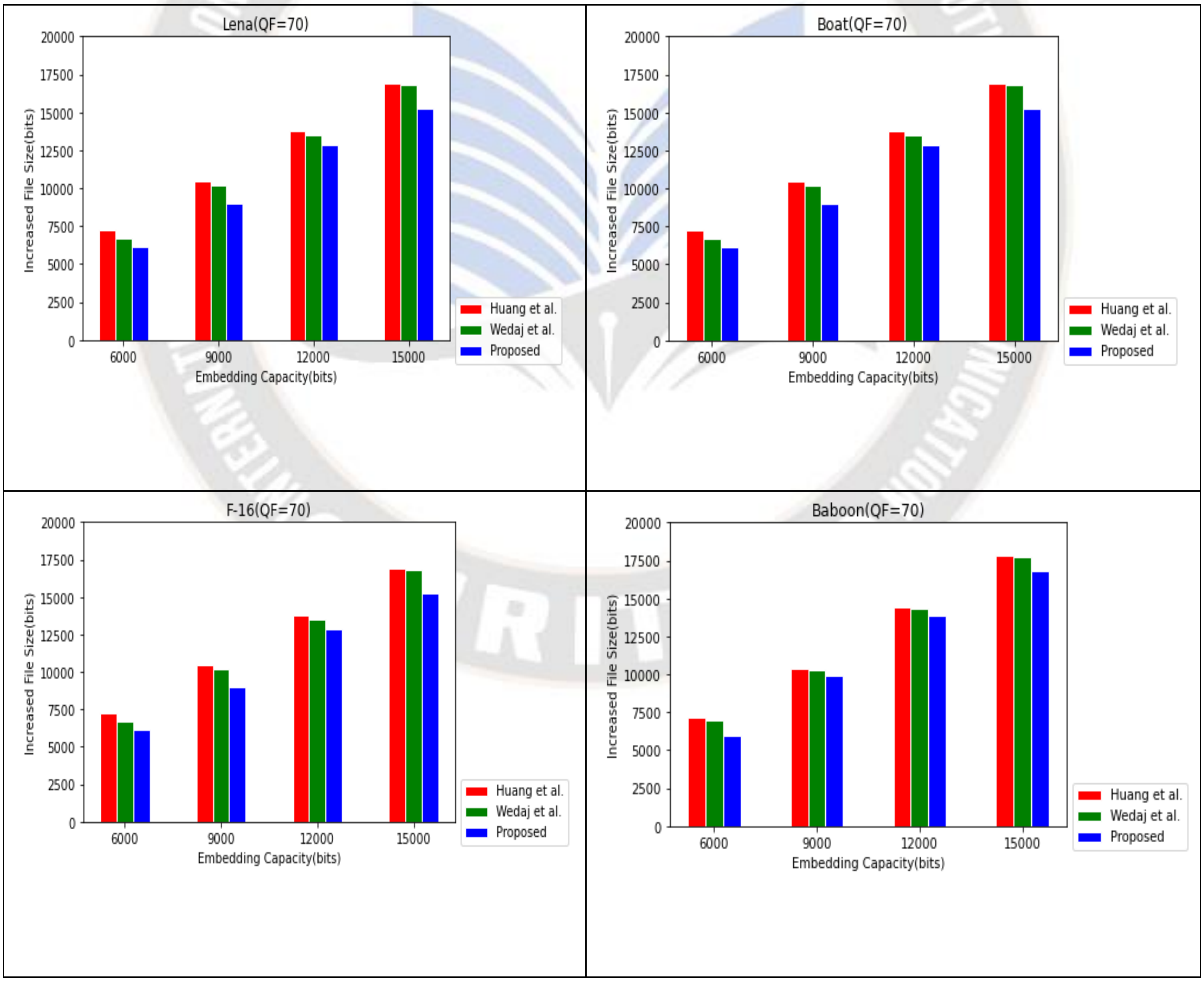
It implies that adding a secret message will inevitably cause the marked image's file size to grow in comparison to the original. Nevertheless, even if this article allows for the increasing, decreasing, or preservation of a coefficient pair's two values (-1 and +1) following each embedding procedure, the final file size does not rise noticeably.

The recommended methodology for different embedding capabilities and the median size of the file expansion for each of the two different reference methodologies are shown in Fig. 5. The embedded capacity (measured in bits) is represented by the horizontal axis, while the file increment value is represented by the vertical axis. Fig. 5. illustrates a comparison between the file size increment and the findings of Huang et al. [25] and Wedaj et al. [12] for identical embedded capacity conditions. It demonstrates how the technique in this paper has file size expansion is more efficient and less. Comparing this benefit to that in Ben et al. [10], though, is less clear. Similarly, comparison between Ben et al. [10] and the methodologies in this article within similar embedding capability is either near or significantly less. Data are incorporated in coefficient pairs, which explains the situation. At least one AC coefficient must be changed during each embedding process; most of the time, both AC coefficients will be altered.

Table III. The file size increment (bits) of each test image compared with the original image at different embedding capacities when QF = 70 and QF = 80.

Image	Method	Embedding capacity (bits) with QF = 70				Embedding capacity (bits) with QF = 80			
		6000	9000	12,000	15,000	6000	9000	12,000	15,000
Lena	Huang et al.[25]	7029	10,725	14,012	17,567	7088	11,952	14,691	18,715
	Wedaj et al.[12]	6880	10,576	13,957	17,296	7021	11,084	14,264	18,201
	Proposed	6388	10,308	13,054	16,276	6302	10,457	13,632	17,603
Boat	Huang et al.[25]	8146	12,024	15,523	18,816	8342	12,846	15,989	19,405
	Wedaj et al.[12]	7256	11,225	15,039	18,742	7782	11,754	15,106	19,225

	Proposed	6638	9964	14,902	17,276	7504	10,579	13,745	18,752
F-16	Huang et al.[25]	7248	10,503	13,776	16,914	7583	11,052	14,251	17,782
	Wedaj et al.[12]	6992	10,208	13,468	16,843	7310	10,768	13,894	17,549
	Proposed	6181	9003	12,885	15,239	6561	10,245	13,699	16,851
Baboon	Huang et al.[25]	7201	10,405	14,392	17,802	7691	11,092	15,068	18,861
	Wedaj et al.[12]	6944	10,296	14,352	17,709	7529	11,248	14,892	18,267
	Proposed	6002	9875	13,904	16,841	6730	9951	13,180	16,979
Peppers	Huang et al.[25]	8831	11,729	14,640	17,568	9024	13,054	15,743	18,628
	Wedaj et al.[12]	8643	11,586	14,421	17,423	9037	12,473	15,449	18,120
	Proposed	8468	11,044	14,329	17,063	8512	10,980	14,849	17,650
Splash	Huang et al.[25]	8906	12,579	15,996	19,106	9658	13,568	16,031	19,827
	Wedaj et al.[12]	8849	12,303	15,743	18,934	9315	13,157	15,990	19,821
	Proposed	8820	12,254	15,520	18,511	9013	12,681	15,501	18,920
Average	Huang et al.[25]	7894	11,328	14,723	17,962	8231	12,261	15,296	18,870
	Wedaj et al.[12]	7594	11,032	14,497	17,825	7999	11,747	14,933	18,531
	Proposed	7081	10,411	14,102	16,860	7432	10,818	14,102	17,798



International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 8
***Article Received:** 25 May 2023 **Revised:** 12 June 2023 **Accepted:** 30 July 2023*

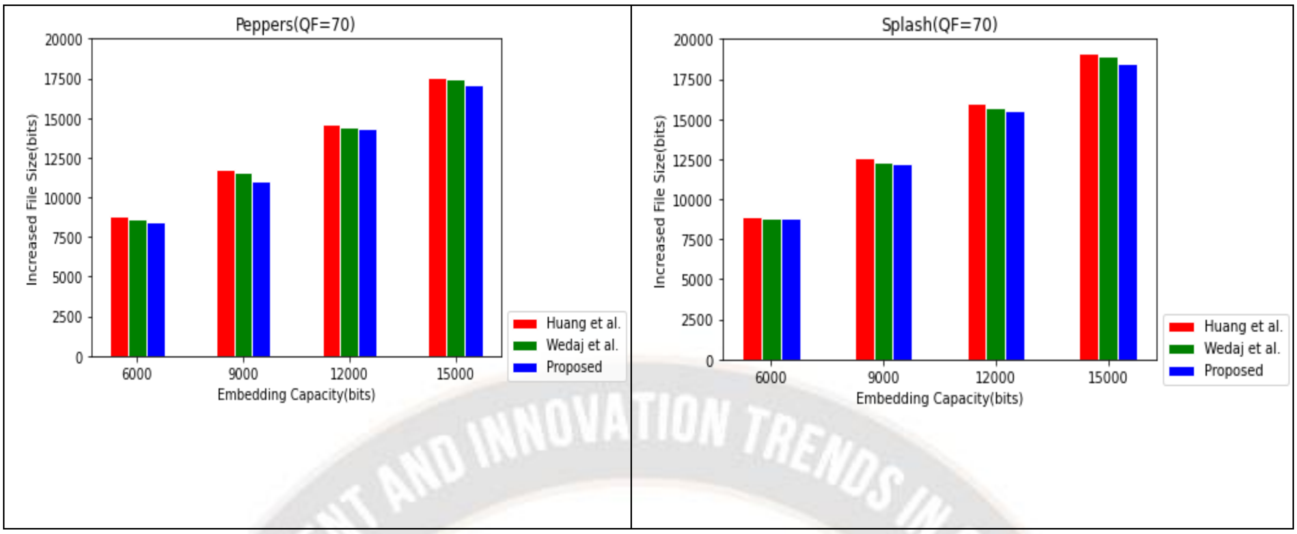
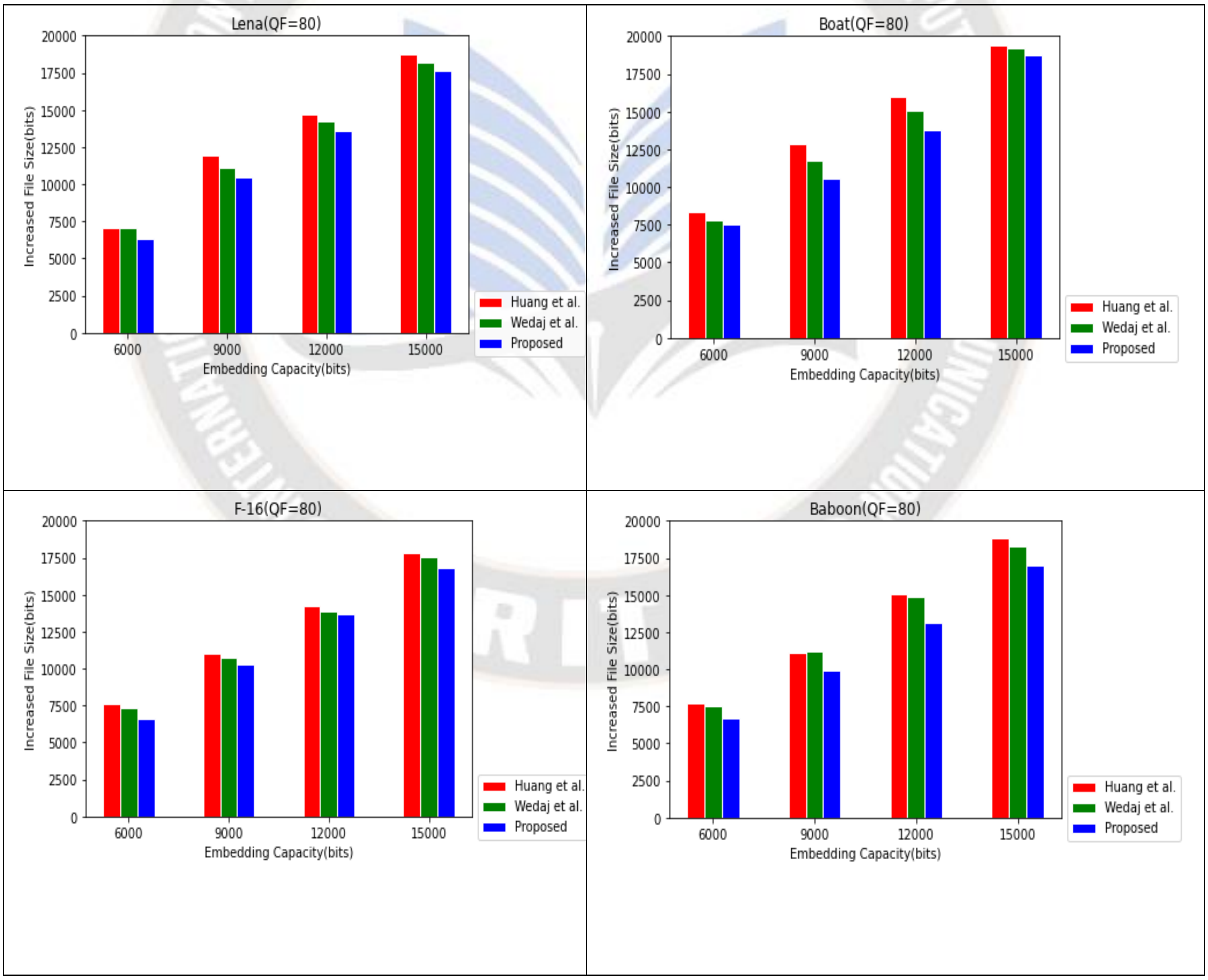
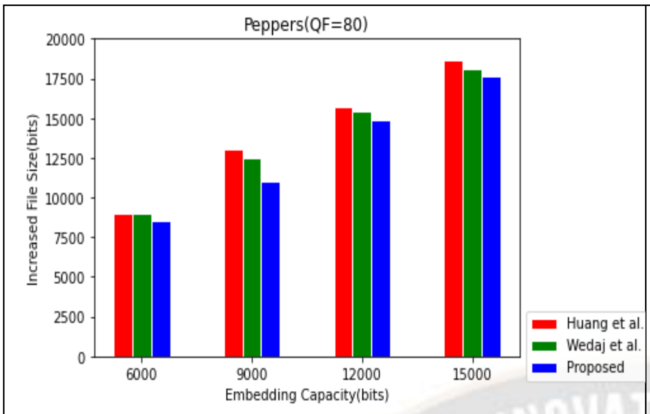


Fig. 5. The three state-of-the-art methods' related file sizes are compared to the method proposed in this article under the condition QF=70





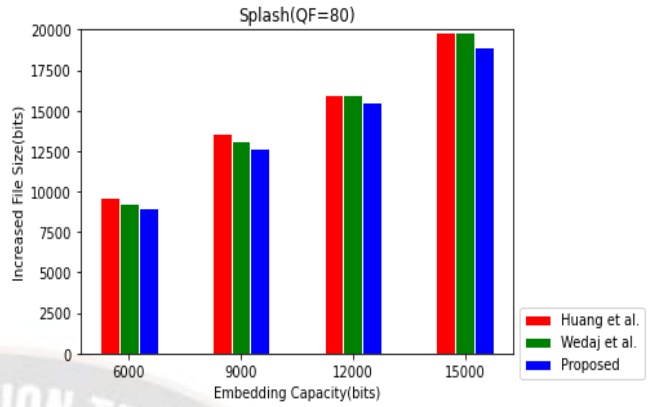


Fig. 6. The three state-of-the-art methods' related file sizes are compared to the method proposed in this article under the condition QF=80

When QF = 80, Fig. 6. compares the file size increment with various payloads. Furthermore, it can be observed by merging the data in Fig. 5. and 6. that, even under varying QF settings, the growth pattern of the file size increment remains constant.

6. Conclusion

In this work we present a high embedding rate RDH approach based on block-wise histogram shifting. The method generates marked information by determining if a block is appropriate for data hiding and by utilizing run-length encoding with DCT, Quantization and Zigzag procedure the information is compressed. In order to clear way for information extraction and recovery of image, the information is embedded using the histogram highest and zero value by shifting the values. Through experimental study of the suggested approach, increased embedding efficiency is attained despite sacrificing the stego image's visual quality. To assess the visual quality of the stego pictures, a reference image quality parameter is applied, such as the PSNR or the structurally comparable index. Nearly all of the stego images had SSIM values of one and PSNR values of more than fifty dB. Furthermore, non-reference picture evaluation like images natural assessor and the blind image spatial accuracy analyzer are utilized to gauge the visual accuracy of stego image, and the outcomes demonstrate that there is little difference in the stego image quality compared to the original image's quality. Instead of focusing on the nonoverlapping blocks, Histogram shifting can be utilized in further studies on the segmentation-created areas histogram.

REFERENCES

[1] Zhicheng Ni, Yun-Qing Shi, N. Ansari and Wei Su, Reversible data hiding. in IEEE Transactions on Circuits and Systems for Video Technology, vol. 16, no. 3, pp. 354-362, 2006.
[2] Kamil Khudhair, S.; Sahu M. K. R., R. Sahu, A.K., Secure Reversible Data Hiding Using Block-Wise Histogram Shifting. Electronics 12, 1222, 2023.
[3] Xiao, Mengyao & Li, Xiaolong & Ma, Bin & Zhang, Xinpeng & Zhao, Yao. Efficient Reversible Data Hiding for JPEG Images With Multiple Histograms Modification. IEEE Transactions on Circuits and Systems for Video Technology. PP. 1-1, 2020.
[4] W. B. Pennebaker, and J. L. Mitchell, JPEG: Still Image Data Compression Standard, Springer, 1993.
[5] G.K. Wallace, The JPEG still picture compression standard. IEEE Trans. Consumer Electron. 38(1), 30–44, 1992.
[6] Z. Qian, X. Zhang, Lossless data hiding in JPEG bit-stream. J. Syst. Software, 85, 309–313,2012.
[7] B.G. Mobasseri, R.J. Berger, M.P. Marcinak, Y.J. Naik Raikar, Data embedding in JPEG bit-stream by code mapping. IEEE Trans. Image Process. 19(4), 958–966 ,2010
[8] Y. Hu, K. Wang, Z.-M. Lu, An improved VLC-based lossless data hiding scheme for JPEG images. J. Syst. Software. 86, 2166–2173,2013.

[9] C.-C. Chang, C.-C. Lin, C.-S. Tseng, W.-L. Tai, Reversible hiding in DCT-based compressed images. Inf. Sci. 177(13), 2768–2786, 2007.

[10] Ben He , Yi Chen , Yonghui Zhou , Yongrong Wang and Yanli Chen, A novel two-dimensional reversible data hiding scheme based on high-efficiency histogram shifting for JPEG images , International Journal of Distributed Sensor Networks, Vol. 18(3), 2022.

[11] Huang, Fangjun & Qu, Xiaochao & Kim, Hyoung & Huang, Jiwu, Reversible Data Hiding in JPEG Images. IEEE Transactions on Circuits and Systems for Video Technology. 26. 1-1, 2015.

[12] Wedaj, F.T., Kim, S., Kim, H.J, Improved reversible data hiding in JPEG images based on new coefficient selection strategy. J Image Video Proc., 63, 2017.

[13] Xinpeng Zhang, Jing Long, Zichi Wang, and Hang Cheng, Lossless and Reversible Data Hiding in Encrypted Images with Public Key Cryptography, IEEE Transactions on Circuits and Systems for Video Technology, Volume 26Issue, pp 1622 - 1631, 2016.

[14] Wang J, Ni J, Zhang X, Shi YQ, Rate and Distortion Optimization for Reversible Data Hiding Using Multiple Histogram Shifting. IEEE Trans Cybern,47(2):315-326, 2017.

[15] VR VijayKumar, V Suresh Babu, Histogram Shifting Based Reversible Data Hiding With Controlled Contrast Enhancement, Third International Conference on Biosignals, Images and Instrumentation (ICBSII), pp-1-6, 2017.

[16] C. -F. Lee, S. -T. Chen and J. -J. Shen, Reversible Dual-image Data Embedding on Pixel Differences Using Histogram Modification Shifting and Cross Magic Matrix, International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), pp. 113-116,2015.

[17] Aziz F, Ahmad T, Malik AH, Uddin MI, Ahmad S, Sharaf M, Reversible data hiding techniques with high message embedding capacity in images. PLoS ONE 15(5): e0231602, 2020.

[18] Pankaj Garg , Singara Singh Kasana, Geeta Kasana, Block-Based Reversible Data Hiding Using Histogram Shifting and Modulus Operator for Digital Images, Journal of Circuits, Systems and Computers, Vol. 26, No. 06, 1750103, 2017.

[19] J. Fridrich and M. Goljan, Lossless data embedding for all image formats, in SPIE Proceedings of Photonics West, Electronic Imaging, Security and Watermarking of Multimedia Contents, vol. 4675, pp. 572–583, 2002.

[20] K. Wang, Z.-M. Lu, Y.-J. Hu, A high capacity lossless data hiding scheme for JPEG images. J. Syst. Software. 86, 1965–1975, 2013.

[21] G. Xuan, Y. Q. Shi, Z. Ni, P. Chai, X. Cui and X. Tong, Reversible data hiding for JPEG images based on histogram pairs, In Proceedings of the International Conference on Image Analysis and Recognition, Montreal, Canada, pp. 715–727, 2007.

[22] Li, Q., Wu, Y., Bao, F., A Reversible Data Hiding Scheme for JPEG Images. In: Qiu, G., Lam, K.M., Kiya, H., Xue, XY., Kuo, CC.J., Lew, M.S. (eds) Advances in Multimedia Information Processing - PCM 2010. PCM 2010. Lecture Notes in Computer Science, vol 6297. Springer, Berlin, Heidelberg, 2010.

[23] Lin, Yih-Kai., High capacity reversible data hiding scheme based upon discrete cosine transformation. Journal of Systems and Software. 85. 2395–2404, 2014.

[24] M.U. Celik, G. Sharma, A.M. Tekalp, E. Saber, Lossless generalized-LSB data embedding. IEEE Trans. Image Process. 14(2), 253–266, 2005.

[25] F. Huang, X. Qu, H.J. Kim, Reversible data hiding in JPEG image. IEEE Trans. Circuits Syst. Video Technol. 26(9), 1610–1621, 2016.