Multimedia Tools and Applications (2021) 80:12641–12670
https://doi.org/10.1007/s11042-020-10217-9

Improved data hiding method for securing color images

Mostafa M. Abdel-Aziz[1] · Khalid M. Hosny[1] · Nabil A. Lashin[1]

Received: 25 April 2020 / Revised: 23 October 2020 / Accepted: 9 December 2020 /
Published online: 12 January 2021

Abstract
Recently, data hiding techniques have become very popular in several vital applications, especially in telemedicine. The reason for this is their ability to give good results such as high embedding capacity while preserving visual image quality as much as possible after extracting the hidden secret message. In earlier studies, many researchers have achieved the goal of reversible data hiding (RDH) algorithm. All these methods have achieved excellent results on standard and natural images. However, in the case of medical images, especially color medical images, we face the problem of how to preserve the visual quality of image contents while achieving the goals of RDH in avoiding the loss of patient data or the distortion of the diagnosing image. In this paper, we proposed a secure data hiding method using a hyper chaotic map and left-most embedding strategy. The proposed methods are hybrid, where it is applied in the DCT frequency domain and encrypted domain together as presented here. This gives a higher embedding rate and higher visual image quality than existing methods without any loss or distortion of both hidden secrets data and reconstructed image. The novelty of this paper is to embed the desired secret data in each quantized block of DCT using (8-bit LMSB) strategy for embedding process. We tested our algorithm on both color medical images and standard color images of different sizes and different formats. We evaluated the performance of our algorithm on the basis of the quality metrics MSE, PSNR, BER, SSIM, Correlation, Symbol Error Rate, additional quality evaluation metrics, execution time, and different types of geometric and signal attacks. All of these parameters are demonstrated and represented in this proposed work in detail.

Keywords Data hiding · DCT · Left-most significant bit · JPEG · YCbCr · PSNR · MSE · Hyper chaotic · Secret data

✉ Khalid M. Hosny
k_hosny@yahoo.com

[1] Department of Information Technology, Faculty of Computers and Informatics, Zagazig University, Zagazig 44519, Egypt

1 Introduction

Color image data hiding is the process of hiding a set of secret data into a cover image imperceptibly such that it does not perceptually distort. Especially in medical images, any distortion leads to misdiagnosis. The medical image, and the hidden patient data, can be accurately recovered at the receiving end without errors. Color images are widely used in many research areas, such as governmental issues, authentication, and other vital issues. Therefore, we tested our scheme on both medical and standard color images.

In past years, several researchers have proposed many useful reversible data hiding (RDH) algorithms according to embedding domain on both spatial domain and frequency domain. [15] Authors Mentioned to these techniques, from this we note that the spatial domain is easy to implement and has low computation cost. However, spatial domain-based techniques still have low capacity and are not robust enough to image compression and other image attacks. On the other hand, transform domain-based methods can embed more bits and have better robustness against different attacks such as noise, JPEG compression, and Gaussian low-pass filter. The study focuses on this community [17]. There are many transform domain-based data hiding techniques such as discrete cosine transforms (DCT), singular value decomposition (SVD), discrete Fourier transforms (DFT), and discrete wavelet transforms (DWT). Since the JPEG and MPEG compression standards are still based on DCT transform, it is more popular to embed or hide data in the DCT domain [17] because DCT is more resistance to several attacks, provide a higher compression ratio and at a time avoid jamming artifacts.

DCT can not only focusing on transform the main information of image into the smallest low-frequency component, but also it can cause the image blocking effect being smallest, which can realize a good compromise between the information centralizing and the computing complication.

Nowadays, chaotic maps strategies in data hiding algorithms have become widely used because it achieves excellent results such as reversibility, high embedding capacity, confidentiality, data integrity, and other evaluation metrics. Chaotic maps strategies can achieve primary security purposes and, at the same time, preserve image quality as good as possible compared with different techniques. A. Awad Attaboy et al. [8] proposed data hiding inside JPEG images using a novel technique DCT-M3 to achieve high resistance to steganalysis; they introduced two ideas.

The first one is to compress the secret message as much as possible using traditional compression techniques to minimize the changes in the cover image. The other idea is to use a new hiding technique DCT-M3 which employs the modulus three as a base factor for data hiding farther than the traditional Least Significant Bits (LSB) technique which uses the modulus two as a base factor. This algorithm gives higher PSNR compared to the traditional LSB algorithm on both color and gray images.

Neha Batra and Pooja Kaushik [9] proposed the implementation of 16×16 quantization table stenography on JPEG color images in the DCT frequency domain. They have achieved a higher embedding capacity compared to other methods that use an 8×8 standard quantization table. The implementation of 16×16 increases the DC-AC coefficients of low- and mid-frequency parts to 136 AC coefficients in the quantization table, which leads to an increase in the embedding capacity. LSB embedding is used in this algorithm to embed secret data into the least significant bits of the AC coefficients. The authors used three basic metrics to evaluate their algorithm: MSE, PSNR, and capacity. Still, they don't mention bit error rate or the execution time of algorithm according to a large (16×16) proposed quantization table, which has complexity time.

Zhenjun Tang et al. [31] expressed a hot topic widely used in recent days. He uses only PNG format for gray images, and proposed RDH algorithm to hide encrypted text in an encrypted alpha channel. The data hider side encrypts the input image using XOR operation with encryption key and then extracts the alpha channel to embed data inside it. Next, combine the encrypted image with the alpha encrypted channel to get a stego encrypted image with PNG format, and then perform the inverse operation to get back the original recovered image and extract the hidden data. The algorithm shows that the reconstructed image is identical to the original image. The author achieves good security results and excellent image quality PSNR up to 95 dB, but if the author uses any lossless compression techniques during the encryption of the alpha channel, he will get more superior image quality.

Osama F. Abdel Wahab et al. [3] have applied multiple methods to hide the secret file in a color image in the DCT algorithm by dividing the cover image into 8 × 8 blocks and applying DCT for each block, then determining LSB of each DCT coefficient and replacing it with LSB of the secret file. Finally, get the stego image. The proposed algorithm shows that embedding in DCT coefficient using LSB achieves a high image quality, low distortion, and high-security level compared with using traditional LSB based algorithm separately. It is known that working on a hybrid domain consists of the DCT frequency domain and the encrypted domain gives high quality enhancement results to the recovered image. MELAD J. SAEED [26] tries to hide text inside a color image using the RGB color model. He does not determine which color level (channel) to choose which may cause image distortion during the data hiding process due to essential components in the used image. He uses principles of a chaotic map as an image index to increase image security and then convert the image to spatial domain to get the stego image.

Yih-Kai Lin [20] enhances the DCT coefficients by trying to hide data in the high-frequency coefficients of DCT. He faced a problem to form back to the correct modified coefficients after embedding. To overcome this problem the author uses integer mapping to implement DCT transformation to get the recovered image from modified coefficients without error with a high capacity; the author adds an excellent genitive to his research.

Alan Anwer Abdulla et al. [5] proposed anew steganography scheme based on pixel intensity value decomposition. Authors decompose pixel intensity values into 16 virtual bit-planes for the embedding process which have property that the number of all bit-planes does not exceed the maximum pixel intensity value. Although embedding in lowers bit-planes have advantage in terms of preserve image quality as possible compared with embedding in higher bit-planes; the scheme has a good result compared with others similar methods; the only limitation of this method is embedding in higher bit-planes.

On order to evaluate any success data hiding criteria there are many general factors that must be addressed Alan Abdulla[4] introduced those requirements as: perceptual quality difference between the stego image and cover image; the payload capacity(the amount of data to be embedding in cover image); the ability to protect recover and detect the hidden secret data by an unauthorized party; the robustness of stego image against different types of attacks; increasing the payload capacity of the cover image at a time preserve image quality ass possible during changing the pixels values to the size of secret data.

Abdulla et al. [6] authors scope their offers to enhance the modern data hiding schemes by increasing the similarity between secret data and cover image using bit-planes mapping technique to achieve a higher embedding ratio in both secret data and cover image LSB plane, this technique makes each cover pixel useable for secret data embedding. The benefits of this technique are that we get an optimized stego image with minimum distortion and high embedding capacity and still robustness to common attacks.

In the field of (YCbCr) color space image encryption Ahmed A. Abd El-Latif et.al [1] proposed A new approach to chaotic image encryption based on quantum chaotic system using integer wavelet transform by scrambling only (Y) color space component then apply two diffusion modules by mixing the features pixel using a quantum chaotic map, finally generating a chaotic key stream image, this scheme achieve a good security and performance.

In the field of electronic media security Vijay Kumar Sharma et.al [28] proposed a high secure DWT steganography scheme based on cryptography. Authors encrypt the input plain text message to produce a secret image to be concealed inside a cover image using Daubechies DWT to finally get a stego image. Authors success to recover the original image and extract the secret image in a good manner with fitting visual quality.

Saleem et al. [27] proposed a reversible data hiding scheme to hide secret image inside color cover image using reserve room before encryption by using LWT on color image and encrypt the secret image using a hyper chaotic system. This proposed scheme achieves a high embedding capacity and high visual quality compared with other similar schemes.

In this research we introduce a high image quality securing using improved data hiding method with minimum errors based on chaotic maps in the frequency domain by using discrete cosine transform (DCT) within encrypted domain on color images. This proposed method will be discussed in detail.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents the proposed methods for data hiding embedding and extraction process. The experimental results and discussion are presented and analyzed in Section 4, Robustness to attacks are represented in Section 5 followed by the conclusion in Section 6.

2 Related work

A lot of data hiding approaches for telemedicine applications and other fields have been proposed in the past decade. There are various techniques used to achieve the reversibility of the host images; some of those methods are based on compression techniques such as JPEG compression and encryption methods, using chaotic maps in both spatial domain and the frequency domain. Interesting effort on data hiding approaches for medical images is discussed.

In Parah et al. [29], the authors try to achieve the goal of image reversibility by transforming a pixel in the medical cover image into 2×2 blocks (equivalent to up-sampling) to hide a patient record and logo in the image. They also try to increase the security of EPR by using the chaotic encryption system, and they consider different attacks on watermarked images, including various image processing and geometric attacks such as noise addition, various filtering process, JPEG compression, and tamper detection. They evaluate watermarked image quality according to PSNR, CAPACITY, NCC, and SSIM score. They show PSNR up to 46.3684 for both standard images and medical images with high capacity. But the watermarked image is still fragile to previous geometric attacks.

Wang et al. [35] have proposed a scheme to improve the embedding capacity for medical images using the logistic map to scramble a cover image before embedding the data, and also to preserve the seed points (ROI) in the image. It is excluded manually before the embedding operation and an adaptive embedding strategy is employed using traditional four least significant bits (4 LSB) of the cover image to hide secret information inside the cover image.

Arunkumar et al. [7] proposed a robust image stenography scheme to hide a secret image in the cover image. The proposed scheme combines Redundant Integer Wavelet Transform

(RIWT), Discrete Cosine Transforms (DCT), Singular Value Decomposition (SVD), and chaotic logistic map. The proposed scheme first scrambling the secret image and performing DCT to divide the secret image into 4 × 4 blocks, followed by applying SVD for each block. On the other side, the cover image was divided into 8 × 8 blocks and RIWT was applied for each block. Afterward, apply DCT on (LL-sub band) and SVD for each block, and finally, by doing the inverse process, get the stego image. The scheme compared to similar methods, according to imperceptibility, robustness, and resistance to geometric attacks. The results showed better image quality relative to similar methods. Authors performed experiments on gray images only.

Kordov and Stoyanov [19] proposed a novel least significant bit (LSB) stenography algorithm based on a Hitzl-Zele chaotic map to hide text in color images. They embedded three bits of the input sequence into the LSB of three-channel RGB and score a good visual image results compared with others similar methods but not mention to different geometric attacks analysis. Puteaux et al. [24] proposed a reversible data hiding method using two approaches based on correction error prediction and linear chaotic map to encrypt pixels of cover image by using a reserved room before encryption and vacating room after encryption principles. The authors used MSB embedding in the encrypted domain to avoid LSB attacks, and they achieved superior image quality with a high capacity on selected gray image only and does not use color images.

Zaghbani et al. [36], the authors proposed a scheme that applies DCT on the color space (YCbCr) of the host image, encrypts the secret data using a logistic map to generate encrypted confidential data, and then embeds it in the modified AC coefficients of a chosen (Cr) color space. The proposed scheme reconstructs the cover image by reverse operations. However, because the Cr channel contains important information about the image which is affected during the data hiding process, this scheme causes image distortion.

Thabit and Khoo [32] proposed a lossless data hiding scheme for color medical images in the frequency domain. This scheme uses the slant-let transform (SLT) matrix to embed patient record data in a color image by modifying the difference between the mean values of the SLT coefficients in the high-frequency sub-bands on the three channels of the RGB color image to increase the embedding capacity through the three channels. They tested their watermarked image robustness against only two types of attacks JPEG compression, additive Gaussian noise (AGN), but more type of attacks is needed to best robustness evaluation.

Lin [21] proposed a data hiding scheme based on DCT coefficient modification, and the author tried to solve the problem of hiding data in high-frequency coefficients, which causes errors in the image and leads to drawbacks in obtaining the correct modified coefficients. To overcome this problem, the author used integer mapping and adjusted LSB embedding in the proposed scheme. Also embedding capacity need to be improved to gain a higher embedding ratio.

Tuncer and Sonmez [34] proposed a new block-based data hiding method to hide secret data inside a cover image, using the principle of the most significant bit (MSB) embedding strategy to determine the block type of the confidential data (msb = 1 for odd block). The other pixels of secret data determine the index of an abnormal pixel, proposed method tested in standard gray images chosen from a dataset.

Elkamchouchi et al. [13] proposed a data hiding scheme to hide an 8-bit grayscale image inside 24-bit true-color image in the chosen spatial domain to easy implantation, authors, using both 1D and 2D chaotic map for time-consuming and robustness against attacks. They selected the red channel from RGB color model of the cover image to embed the secret image in one or more LSBs on the selected color channel. Authors mentioned only to basic image quality

metrics PSNR and MSE, for more perceptual quality other assessment are needed such as SSIM, correlation degree, etc.

Ke et al. [16] proposed a high capacity and prediction error correction on medical images based on the most significant bit with an image encryption algorithm. The authors use the local correlation between the pixels and their adjacent areas in the image to adjacent two pixels closely and predict the desired pixel, finally get an encrypted image without errors which lead to reconstructing the original image with minimal errors.

Thakur et al. [33] proposed a multi-level security watermarking approach in the transform domain using DWT, DCT, and SVD. They embed the patient report in the host image and apply chaos-based encryption on the watermarked image to increase robustness to many attacks without any distortion in the recovered image. The authors tested many different attacks on both watermarked and recovered watermark images. The proposed approach has a good result compared with other similar methods.

Haque et al. [14] proposed a data hiding scheme based on JPEG compression, DCT, chaotic logistic map and SHA-256 hash function. Any type of data can be hidden inside the JPEG compressed cover image. Encryption of secret data by chaotic logistic map enhances the security level of confidential data and generates SHA-hashing to assure data integrity. The authors evaluated and analyzed the results based on different quality factors and various attacks.

Karabatak and Yigit [23] proposed a new method to minimize the corruption which occurs in cover images reasoned distortion during the data hiding process when using traditional LSB, and improved the hiding data conservation by creating an appropriate masking algorithm (OMVG) to encrypting data. The authors used three RGB channels in the cover image to increase the amount of data to be concealed in the cover image with no dependence on the size of a cover file. The proposed OMVG method gives a high PSNR value compared with traditional LSB method.

Rhouma and Belghith [25] proposed two different attacks on a hyper-chaos image. The first one shuffles the image rows and columns to disarrange the correlation among pixels by iterating the logistic map. The second attack generates a keystream to mix the generated key with the shuffled image using a hyper-chaos system. The authors study these attacks in detail to evaluate a recent data hiding algorithm because of the extensive use of these chaotic methods.

Khosravi and Yazdi [18] enhance the watermarking techniques of the DICOM images using a hybrid technique based on histogram error computing and image interpolation with adaptive weights, this method provides ahigh performance compared with others similar methods.

The motivation of this paper is focus on its promising contribution concerned to spread out a meaningful image quality to increase the correlation similarity between the original cover image and reconstructed image and increasing capacity. The robustness of this proposed method will be tested against several types of attacks such as: cropping attack, rotation attack, scaling attack, shift array circular attack, horizontal shear attack, JPEG QF = 100/80/60, and reconstruction filter (using bi-linear/bi-cubic interpolation after up-scaling a 25% and down-sampled image 25%).

3 Proposed method

The main aim of the proposed method is to maximize the embedding capacity, integrity, and security, and preserve the visual quality of an image without any distortion if possible. The proposed method targets both color medical images and standard images in different formats

like JPEG, PNG, BMP, TIFF, and other image formats with different image sizes (512 × 512 and 256 × 256).

The input color image is scrambled using a hyper chaotic map and reshaped to a sequence to get an encrypted image that converted from the RGB color model to JPEG-YCbCr color space. We select the luminance channel (Y) from the color space for the embedding process because it is a grayscale version of the original color image. However, it is the ideal space for data hiding, and it is easy to reconstruct back the original image again without problems. We divide the luminance channel (Y) into 8 × 8 non-overlapping blocks and apply DCT on each block, then quantize each DCT block using the recommended standard JPEG quantization in Table 1.

We proposed embedding in the left-most significant bit (left-MSB) to avoid frequent attacks on a right-most significant bit (traditional LSB). Most researchers have widely use AC and DC coefficients in the data hiding process. It has become more prone to attacks and affects image quality because these coefficients contain essential visual parts of the image and high energy components, so we use the left-most embedding process instead. Moreover, to add more security and integrity to the secret data we compressed it using Huffman coding. The complete data hiding procedures are presented in Fig. 1a and the data extraction and image recovery process is shown in Fig. 1b**.**

3.1 Proposed algorithm

3.1.1 Procedure for data hiding process

The proposed model introduced with the help of block diagram in Fig. 1a, the embedding steps are:

Step 1: Given an input color image.
Step 2: Scrambling the input color image using a hyper chaotic system for more security and improvement of the threshold-based correlation to get an encrypted version of the image.
Step 3: Divide the encrypted image into the RGB color model.
Step 4: Convert the RGB color model to the YCbCr color space.
Step 5: Extract the first channel (Y) and the second two channels (Cb and Cr).
Step 6: Select the first channel (Y) for the data hiding process and perform the following steps.
Step 7: Divide the chosen luminance channel (Y) into 8 × 8 non-overlapping blocks.
Step 8: Apply and compute DCT for each block.

Table 1 Recommended standard JPEG quantization Table [17]

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

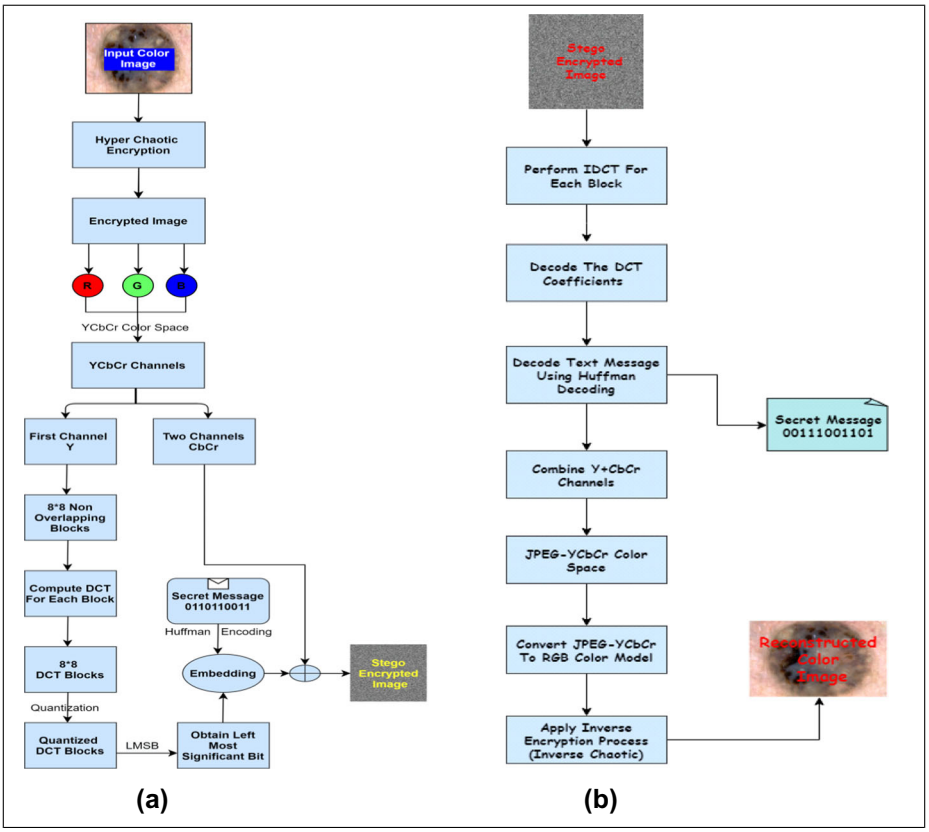

Fig. 1 Block diagram **a** Data hiding embedding and **b** Data Extraction and recovery processes

Step 9: Quantize each DCT block using the recommended standard JPEG quantization in Table 1.

Step 10: Embedding compressed text data in left-most significant bit using Huffman encoding to avoid attacks in traditional right-most LSB.

Step 11: Combine all YCbCr Components again.

Step 12: Finally, we get an encrypted stego image containing secret hidden data inside of it.

3.1.2 Procedure for image recovery and data extraction process

The image recovery and extraction process is the inverse of the embedding process. The proposed model for retrieval and extraction is presented with the help of block diagram in Fig. 1b. Detailed steps are shown below:

Step 1: Given an input encrypted stego image that contains data inside it.

Step 2: Perform an inverse process (IDCT) on each DCT block.

Step 3: Decode the DCT-Coefficients.

Step 4: Apply the inverse process of left-most significant embedding process (decoding message).

Step 5: Extract and decode the hidden text message from the DCT-Coefficients using Huffman decoding.
Step 6: Rearrange and combine all channels again (stego encrypted channel Y + CbCr channels).
Step 7: As a result, we get JPEG-YCbCr color space.
Step 8: Convert JPEG-YCbCr color space to RGB color mode.
Step 9: Regenerate the color image by applying the inverse scrambling process.
Step 10: Finally, we reconstruct the original color image.
Step 11: In the following, we will discuss these procedures in detail:

3.2 Image encryption

To increase the security of input color image we use a hyperchaotic system which was introduced by Rhouma and Belghith [25]. We first shuffle the image's rows and columns to disarrange the correlation among pixels followed by a logistic map. A random key stream is generated to mix the logistic map with the pixels that result from the shuffled image using a hyper chaos system. Next, we created a keystream which neither depends on the plaintext or on the cipher text. Any changes in the key stream will change the whole encryption process which leads to the false decryption process. In this way the cryptosystem is totally secure and more robust to more popular attacks. To break this cryptosystem requires previous knowledge of plaintext and cipher text. In this way, our stego encrypted image is more secure. We tested several common attacks on our stego image to evaluate its robustness.

Also, we have used the following color models to preserve image quality and increase capacity.

3.3 RGB color model

As we know, RGB stands for red, green, and blue. The RGB color model shown in Fig. 2 is additive: red, green, and blue components are added together in varying proportions to produce an extensive range of colors**.** The RGB model has been wildly successful and is frequently used in sensor and image-processing applications.

3.4 JPEG-YCbCr color space

In YCbCr color space, the luminance component (Y) represents the intensity of the image. It is the ideal space for data hiding process; the chrominance components (Cb and Cr) specify the blueness and redness of the image respectively. JPEG-YCbCr color space is a rescaling of YCbCr which are used in the JPEG image format to achieve maximum robustness against several attacks with Y, Cb, and Cr [17]. The transformation from the RGB color model passage to JPEG-YCbCr color space is ensured [36] in formula (1) and transformed back to RGB color model ensured in formula (2):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299000 & 0.587000 & 0.114000 \\ -0.168736 & -0.331264 & 0.500002 \\ 0.500000 & -0.418688 & -0.081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \quad (1)$$

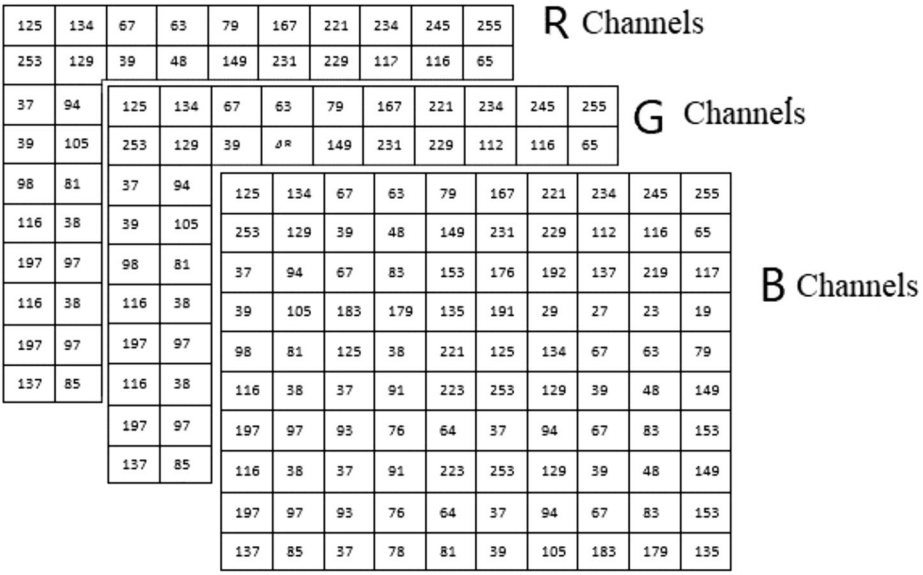

Fig. 2 Image file colors channel RGB [23]

And transformed to the RGB color model again:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.40210 \\ 1.0 & -0.34414 & -0.71414 \\ 1.0 & 1.77180 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ Cb-128 \\ Cr-128 \end{bmatrix} \tag{2}$$

3.5 Discrete cosine transform (DCT)

After changing the color space, we select the luminance component(Y) and divide into 8 × 8 non-overlapping blocks converted from unsigned integers to signed integers and input to the DCT as shown in the following mathematical definition:

$$F(u,v) = \frac{1}{4}\,\mathrm{C}(u)\mathrm{C}(v) + \sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) * cos\frac{(2x+1)u\pi}{16} cos\frac{(2y+1)v\pi}{16} \tag{3}$$

Each 8 × 8 block of the source image samples is effectively a 64-point discrete signal which is a function of the two spatial dimensions x and y. The DCT takes such a signal as its input and decomposes it into 64 unique two-dimensional spatial frequencies which comprise the input signal's spectrum. The output of the DCT is the set of 64 basis-signal amplitudes (DCT coefficients) whose values are regarded as the relative amount of the 2D spatial frequencies contained in the 64-point input block. The DCT based encoding process is shown in Fig. 3.

3.6 Quantization

Quantization is the process of dividing each DCT coefficient by its corresponding quantized coefficients, followed by rounding to the nearest integer, as shown in Eq. (4):

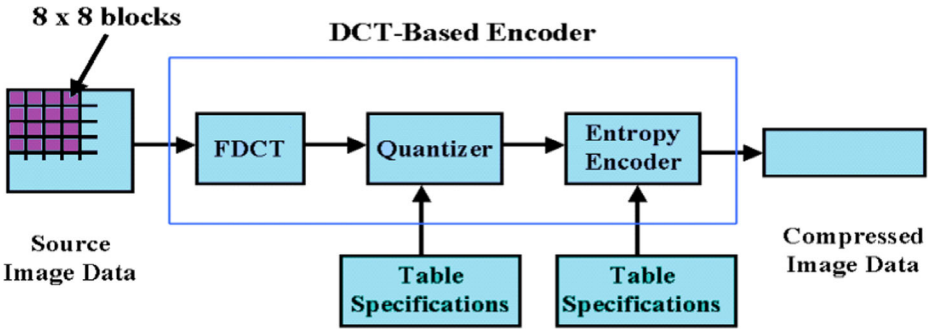

Fig. 3 Based DCT encoding process

$$F^{Q}(u,v) = Integer\ Round\left[\frac{F(u,v)}{Q(u,v)}\right] \tag{4}$$

The output of the quantization process is normalized by the quantized coefficients. Each step size of quantization ideally should be chosen as the perceptual threshold to compress the image as much as possible without visible artifacts; it is also a function of the source image characteristics which display characteristics and viewing distance. The purpose of quantization is to discard information that is not visually significant. To achieve further compression, each of the 64-DCT coefficients is uniformly quantized in conjunction with a 64-element of recommended Standard Quantization Table as shown in Table 1 below:

3.7 Left-most significant bit (left-MST) embedding and extraction process

In literature reviews, most researchers have two basic aspects in the data hiding process (MSB and LSB) as shown in Fig. 4. The Most Significant Bit (**MSB**) is the bit position in a binary number format having the most significant values; the MSB is sometimes referred to as left-MSB.

The Least Significant Bit (**LSB**) is the bit position in a binary integer giving the units value to determine whether the number is odd or even; the LSB is sometimes referred to as right-MSB.

As we know, embedding in left-MSB increases the embedding capacity (close to 1bpp) and preserves the image visual quality during the data hiding process liken original image with PSNR up to (≈ **96.29** dB) as shown in the experimental part. It is introducing a reversible data hiding method in both the frequency domain and encrypted domain [16].

Compared with the traditional LSB embedding algorithm, which uses 4-bit substitution, LSB does not achieve a high embedding capacity and does not introduce a reversible data hiding method, especially when working with the encrypted domain. On the other hand, the left-MSB uses 8-bit, which increases the embedding capacity as much as possible.

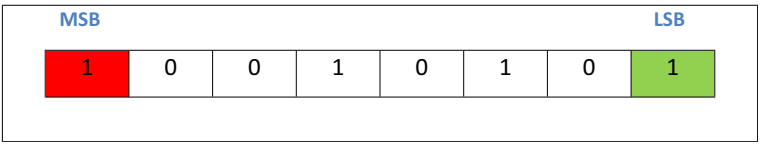

Fig. 4 Most significant bit and least significant bit significances

We are embedding a secret message in the left-MSB to avoid loss of confidential data and errors during the data hiding process. In the proposed data hiding method, we use this recommended embedding in the left-MSB, increase the embedding capacity, and increase the image quality on both 512 × 512 and 256 × 256 images. The result of these is shown in the experimental and discussion section in details.

In the data hiding stage, at the sender side, we embed the secret text message in the quantized version of the input color image after applying steps on it, as shown in Fig. 1a. We first select the desired Luminance encrypted channel(Y), divide it into 8 × 8 non-overlapping blocks, and compute DCT for each block to get 8 × 8 DCT blocks. We then quantize each block using quantization from Table 1. The result is a quantized version of 8 × 8 blocks represented in Fig. 7a in the experimental results. Next, we obtain 8-bit left-MSB to embed the desired secret text message.

To ensure integrity, we also compress the secret text message using Huffman coding to a proper message extraction without any data loss or errors. The input text secret message is converted from ASCII mode into binary to perform the embedding process in selected 8-bit left-MSB. We explain this embedding strategy with an illustrative example in the experimental section.

On the other side, the receiver takes the stego encrypted image, which contains the secret message and applies an inverse operation (IDCT) for each block, then decodes the DCT coefficients and the message using Huffman decoding to finally get the correct secret text message without error; and finally, reconstruct the image again.

4 Experimental results and discussion

We conducted several experiments to evaluate the effectiveness of our proposed method. In these experiments we tested our scheme on five popular standard color images of sizes 512 × 512 and 256 × 256 from dataset [30], and nineteen color medical images with size 256 × 256 from medical datasets [10–12, 22] shown in Fig. 5.

The data hiding simulation process was encoded in MATLAB 2015a (V 8.5.0) and run on PC Intel core i5 with 6GB of ram under Windows 7 64-bit operating system.

The desired embedding part is represented in graphical simulation in Fig. 7 for more clarification. We embedded a chosen secret text message in the quantized version of the DCT of encrypted luminance channel(Y) shown in Fig. 7a using 8-bit left-MSB embedding strategy.

Many images are adopted to demonstrate the feasibility of our proposed method. For example, a chosen image **seq.10** showing the experimental result in short is presented in Fig. 6 below.

Example of a chosen secret text message of size 20 bytes to be concealed:

Chosen Input Text Message: “This is a patient report”

ASCII mode conversion:

116 104 105 115 32 105 115 32 97 32 112 97 116 105 101 110 116 32 114 101 112 111 114 116

Conversion from ASCII to BINARY:

01110100 01101000 01101001 01110011 00100000 01101001 01110011 00100000 01100001 00100000 01110000 01100001 01110100 01101001 01100101 01101110 01110100 00100000 01110010 01100101 01110000 01101111 01110010 01110100

Fig. 5 24 Tested images selected from various data sets

By applying Huffman encoding, we get a stego encrypted image shown in Fig. 7b, containing the secret text message. To reconstruct the original image, apply the inverse operation steps as shown in Fig. 1b and represented in Section 3.1.b in detail. Finally, we get the reconstructed image as shown in Fig. 7c, which is more visual and closely resembles the original cover image.

As a result, the proposed method achieves a good result of PSNR (up to **96.29** dB), distortion less, MSE, BER, SSIM, correlation, and execution time; all of these results are discussed below in Tables 2 and 3 in detail. Also, histogram analysis is shown in Figs. 8 and 9

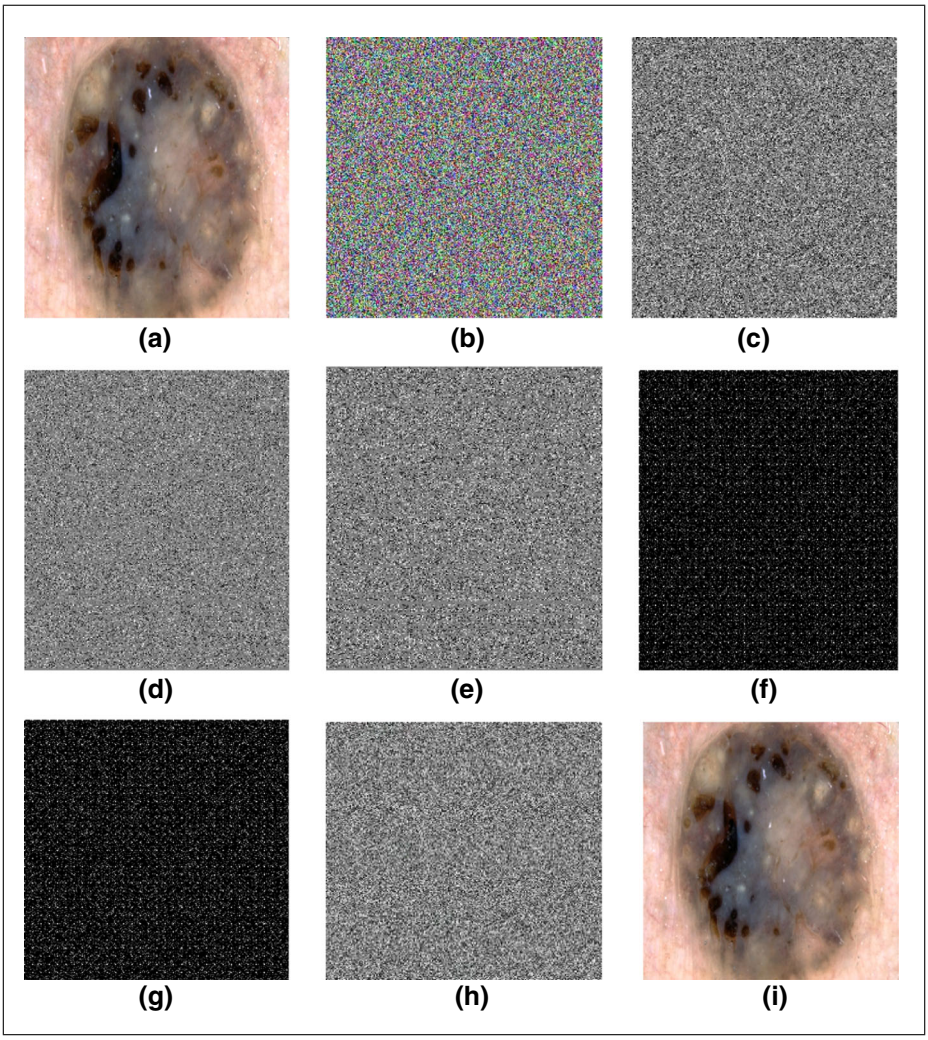

Fig. 6 Simulation results overview of a chosen test image seq.10. **a** Original Input Color Image, **b** Encrypted Image, **c** Encrypted Luminance Channel Y **d** Encrypted Channel Cb, **e** Encrypted Channel Cr, **f** DCT Of Encrypted Channel Y, **g** Quantized Version Of DCT, **h** Stego Encrypted Image, **i** Reconstructed Image

of the original color image and reconstructed color image for the chosen text size embedded. A crypto analysis of different attacks on stego encrypted images is mentioned here.

We evaluate our proposed method using symbol error rate to determine secret message errors. In our experiments we use the PAM model, which refers to pulse-amplitude modulation, where the message information is encoded in the amplitude of a series of signal pulses. The amplitudes of theoretical carrier pulses are varied according to the sample value of the message signal. The results of different text sizes of 20 bytes and 40 bytes presented in Fig. 10 show that our tested symbols are all positive and above the theoretical line, which means that no symbol errors were detected.

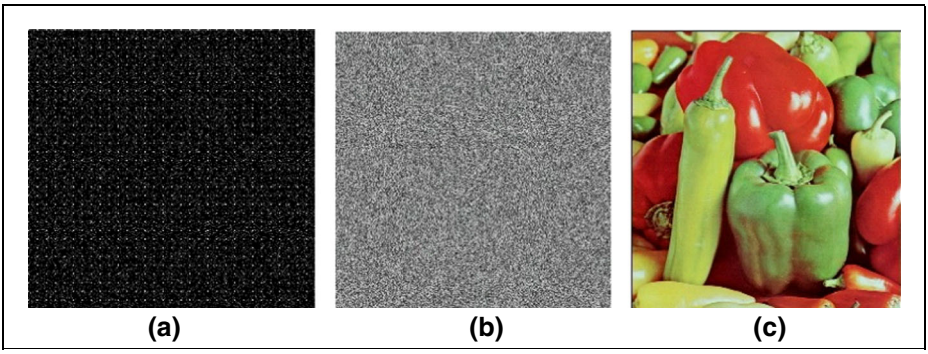

Fig. 7 **a** The desired Quantized DCT coefficients Selected to embedding process, **b** Encrypted Stego Image Within Hidden Message and **c** Reconstructed Image

4.1 Perceptual quality analysis

4.1.1 PSNR

In the experimental study, PSNR values are used to compare the difference between the input image and the reconstructed image. The higher the PSNR value, the less difference between the two images; also, MSE is calculated using Eq. (5) and the PSNR value is calculated using

Table 2 Five tested standard color images analysis

Image	Image Sequence	Image Size	Text Size (bytes)	Execution Time (Seconds)	Capacity (No. of bits)	MSE	SSIM	Correlation	PSNR (db) Reconstructed Image
	Seq.1	512*512	20	657.01099 seconds	655,360	0.000187	0.958900	0.9234624	85.41
		256*256	20	55.234475 seconds	163,840	0.0000611	0.998778	0.9638643	90.27
	Seq.2	512*512	20	660.830874 seconds	655,360	0.000973	0.7114989	0.90398730	78.25
		256*256	20	52.358645 seconds	163,840	0.000183	0.978737	0.97341193	80.73
	Seq.3	512*512	20	654.595490 seconds	655,360	0.001945	0.630531	0.85068726	75.24
		256*256	20	53.011533 seconds	163,840	0.005127	0.9756928	0.95968028	71.03
		512*	20	655.530051	655,	0.000065	0.9764682	0.96489234	90.01
	Seq.4	512		seconds	360				
		256*256	20	53.455193 seconds	163,840	0.005402	0.7604519	0.9261742	70.80
	Seq.5	512*512	20	661.368170 seconds	655,360	0.002918	0.6246547	0.9565434	73.47
		256*256	20	56.499630 seconds	163,840	0.002991	0.9180265	0.9657985	73.37

Table 3 19 Tested medical color images analysis

Image	Image Sequence	Image Size	Text Size (bytes)	Execution Time (Seconds)	Capacity (No. of bits)	MSE	SSIM	Correlation	PSNR (db) Reconstructed Image
	Seq.6	256 * 256	20	51.192	163,840	0.000061	0.99189985	0.99820519	90.27
	Seq.7	256 *256	20	51.402	163,840	0.000015	0.99624799	0.99786682	96.29
	Seq.8	256 *256	20	54.194	163,840	0.000015	0.97214491	0.95275466	96.29
	Seq.9	256 *256	20	54.801	163,840	0.000137	0.98703905	0.99554679	86.75
	Seq.10	256 *256	20	53.446	163,840	0.000015	0.97280964	0.99626531	96.29
	Seq.11	256 *256	20	53.902	163,840	0.000275	0.98057342	0.94967011	83.74
	Seq.12	256 *256	20	52.716	163,840	0.000443	0.99135644	0.9946498033	81.67
	Seq.13	256 *256	20	52.575	163,840	0.000092	0.98764830	0.9973984226	88.51
	Seq.14	256 *256	20	54.495	163,840	0.000061	0.99424642	0.99272652	90.27
	Seq.15	256 *256	20	54.160	163,840	0.000504	0.88017385	0.95715887	81.11
	Seq.16	256 *256	20	52.804	163,840	0.000580	0.98560757	0.99595045	80.49
	Seq.17	256 *256	20	57.595	163,840	0.000015	0.99613273	0.99448320	96.29
	Seq.18	256 *256	20	55.022	163,840	0.000015	0.99695875	0.99566112	96.29
	Seq.19	256 *256	20	55.393	163,840	0.000626	0.99686559	0.98372562	80.16
	Seq.20	256 *256	20	52.7793	163,840	0.000137	0.99834184	0.99594899	86.75
	Seq.21	256*256	20	54.5654	163,840	0.000384	0.09876324	0.99765432	83.45
	Seq.22	256*256	20	52.653	163,840	0.000075	0.99424782	0.99272321	90.65
	Seq.23	256*256	20	58.983	163,840	0.000814	0.9856421	0.992562245	81.34
	Seq.24	256*256	20	26.654	163,840	0.000076	0.9954319	0.999786315	88.91

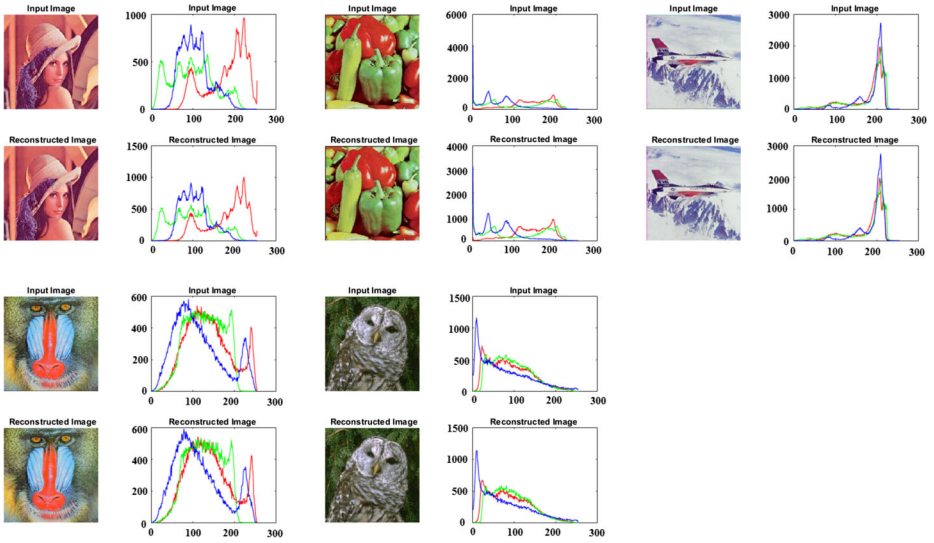

Fig. 8 List of 5 tested standard color images and its corresponding histogram before and after data hiding process for both the original input image and reconstructed image

the formula [23] written in Eq. (6).

$$Mse = \frac{1}{m+n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j)-K(i,j)]^2 \tag{5}$$

$$PSNR = 10*\log_{10}\left(\frac{MAX_I{}^2}{MSE}\right) \tag{6}$$

Since we deal with color images [23], the PSNR value calculated separately for each color channel R, G, and B using Eq. (7), then converted to a general form [18] called (MPSNR) which represents all color channels and **M** is refer to RGB image and equal **3** as shown in Eq. (8), respectively.

$$\begin{aligned} PSNR\,r &= 20*\log_{10}\left(\frac{255}{MSE\ \ r}\right) \\ PSNRg &= 20*log_{10}\left(\frac{255}{MSEg}\right) \\ PSNR\,b &= 20*\log_{10}\left(\frac{255}{MSE\ \ b}\right) \end{aligned} \tag{7}$$

$$MPSNR = 20\log\frac{\mathbf{2}^{d}-\mathbf{1}}{\sqrt{\frac{\mathbf{1}}{\boldsymbol{N1}\times\boldsymbol{N2}\times\boldsymbol{M}}\sum_{i=1}^{N1}\sum_{j=1}^{N2}\sum_{k=1}^{M}\left(\boldsymbol{f}_{ik}-\boldsymbol{f}'_{ik}\right)^{2}}} \tag{8}$$

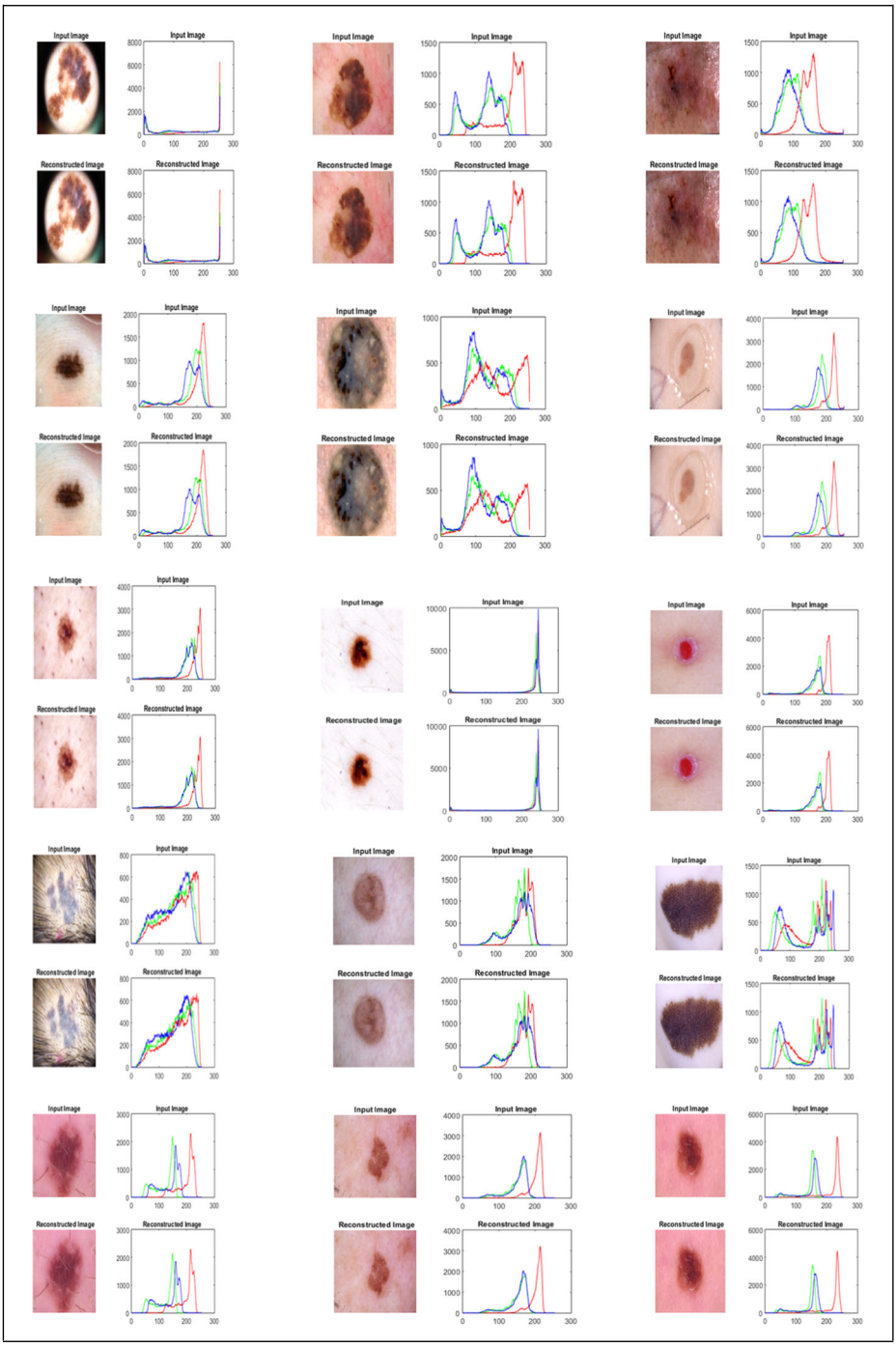

Fig. 9 List of 15 tested color medical images and its corresponding histogram before and after data hiding process for both Original input image and reconstructed image

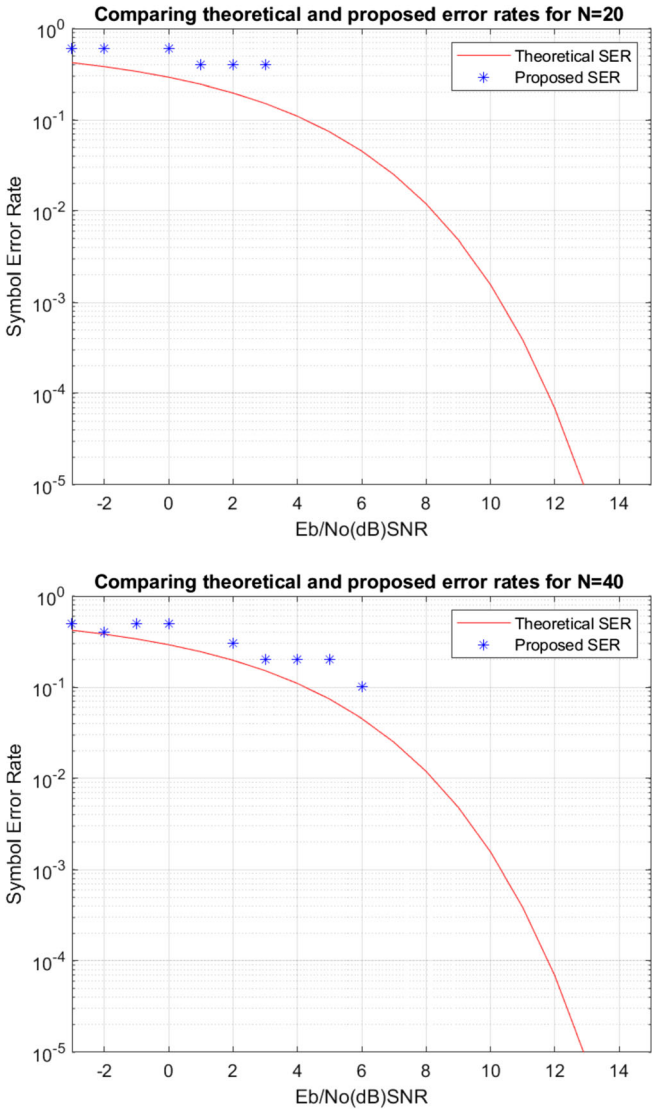

Fig. 10 Symbol error rate At $N = 20$, $N = 40$ symbols

Generally, our method scores a high PSNR compared with other similar methods according to the values shown in Tables 2 and 3, respectively.

Also, in In Table 6, we provided computed values for MSE and PSNR for the proposed method, where MSE and PSNR are calculated for selected images (256*256) with 100 chars (800 bits), 200 chars (1600 bits), 300 chars (2400 bits), 400 chars (3200 bits), and 500 chars (4000 bits) embedded, are presented. From the obtained results, Table 6, it is clear that the PSNR values up to 90 dB, which indicate that our method has still, score a good image quality as shown in simulation results Fig. 16**.**

4.1.2 Correlation degree

Also, we measure the effectiveness of the proposed method by measuring the correlation degree between the original image and the reconstructed image. The proposed method scores value up to (0.9734119) in a chosen image **seq.2** and the results of other tested images are shown in Tables 2 and 3, which mean that the reconstructed image more closely resembles the original cover image.

4.1.3 Structure similarity index measurement (SSIM)

SSIM is used for measuring the similarity degree between two images. The SSIM index is a full reference metric; in other words, the measurement or prediction of image quality based on an initial uncompressed or distortion-free version of the image as the reference (original image). SSIM is designed to improve on traditional methods (https://en.wikipedia.org/wiki/Structural_similarity), such as PSNR and MSE. The experimental results are shown in Tables 2 and 3.

4.1.4 Additional evaluation metrics

Additional quality evaluation metrics on both input image and reconstructed image are also mentioned here, and the result shows that the reconstructed image is very close to its original image content and is slightly affected through the data hiding embedding and extraction processes. Our proposed method succeeds at preserving image quality as much as possible.

- Mean value: gives the contribution of individual pixel intensity for the entire image.
- Standard Deviation: a measure that is used to quantify the amount of variation or dispersion of whole images**.**
- Entropy: a measure of image information content, interpreted as the average of the information source and defines the intensity level in which individual pixels can adapt (adaptive pixel).
- Gradient**:** measures the directional changes in the intensity of colors in images.

The experimental results are shown below in Table 4 for five-color standard tested models and a chosen five-color medical image from the dataset shown in Table 5.

5 Robustness to attacks

Different types of attacks on our encrypted stego image were evaluated, analyzed, and briefly discussed. First, the robustness to a chosen plain text attack of the selected images **seq.2** and **seq.16**. Graphical simulation results are presented and shown in Figs. 11 and 12 on encrypted stego images of different sizes, and slightly changing in the key leads to misleading or reconstructing the correct original image.

Second, we evaluated the robustness of our stego image against several geometric attacks such as cropping attack, rotation attack, scaling attack, shift array circular attack, horizontal shear attack, JPEG QF = 100/80/60, and reconstruction filter (using bi-linear/bi-cubic interpolation after up-scaling a 25% down-sampled image).

Table 4 Evaluation experimental results of 5 selected standard color images from dataset

Selected Images Evaluation Metrics	Image Seq. 1		Image Seq. 2		Image Seq. 3
	Original	Reconstructed	Original	Reconstructed	Original
Quality Score prediction	25.5934127192746	25.0272257603391	27.2193001350471	27.934579757608	23.358659885494
Mean value	128.169896443685	128.182464599609	107.423100789388	107.157424926758	181.933837890625
Standard Deviation	59.0032237117	58.7152853162064	64.6929101239839	64.4476201893119	43.1234039153546
Entropy	7.76544012558603	7.759394173735	7.70596568964168	7.71712017875735	6.68088872739393
Gradient	9.69267441829126	8.35487492026713	9.01275121779617	8.1568040735724	8.9056753564405
Total Average	128.169896443685	128.182464599609	107.423100789388	107.157424926758	181.933837890625

Selected Images Evaluation Metrics	Image Seq. 3	Image Seq. 4		Image Seq. 5	
	Reconstructed	Original	Reconstructed	Original	Reconstructed
Quality Score prediction	23.3216012911421	36.8613876050798	36.002505152919	36.6551580188642	36.5222390122441
Mean value	181.01252746582	126.495676676432	126.795542399089	84.6970062255859	84.932622273763
Standard Deviation	42.8437484407721	52.438657164251	51.9454789342599	52.427396822741	52.202923490064
Entropy	6.68550923280271	7.67762400603646	7.66304531432815	7.61725581893351	7.61405349683124
Gradient	8.19105789313497	15.3995377206843	13.2236300494763	13.3853020894573	12.4108194520845
Total Average	181.01252746582	126.495676676432	126.795542399089	84.6970062255859	84.932622273763

Table 5 Evaluation experimental results of 5 selected medical color images from dataset

Selected Images / Evaluation Metrics	Image Seq. 6		Image Seq. 7		Image Seq. 8
	Original	Reconstructed	Original	Reconstructed	Original
Quality Score prediction	22.9739360660786	22.9528989770018	24.9528989775632	24.9228989745934	10.111698275168
Mean value	130.801427205404	130.310602823893	148.939809163411	148.857467651367	108.519922892253
Standard Deviation	90.8894704602146	90.7980246985715	55.5352999887165	55.6482846584222	39.3425359228211
Entropy	7.4416041838935	7.44925223405891	7.57629918545474	7.57582876217363	7.29081787461978
Gradient	4.09319101579205	3.84870518984019	2.36757324613109	2.24475128043239	4.68832230982661
Total Average	130.801427205404	130.310602823893	148.939809163411	148.857467651367	108.519922892253

Selected Images / Evaluation Metrics	Image Seq. 8	Image Seq. 9		Image Seq. 10	
	Reconstructed	Original	Reconstructed	Original	Reconstructed
Quality Score prediction	10.8871590717768	59.9228989751367	59.867756355318	10.4207147797794	10.071044241747
Mean value	108.327926635742	183.461217244466	183.403213500977	135.039204915365	134.563217163086
Standard Deviation	39.0667772917981	47.1868580556835	47.2436372868689	55.9575779886655	55.7827021497838
Entropy	7.28167909441898	7.05588427904106	7.05289033560234	7.72391652525048	7.72146976851661
Gradient	4.23539882614632	4.46547385217565	4.10660805780952	4.30628888997567	4.10895482831685
Total Average	108.327926635742	183.461217244466	183.403213500977	135.039204915365	134.563217163086

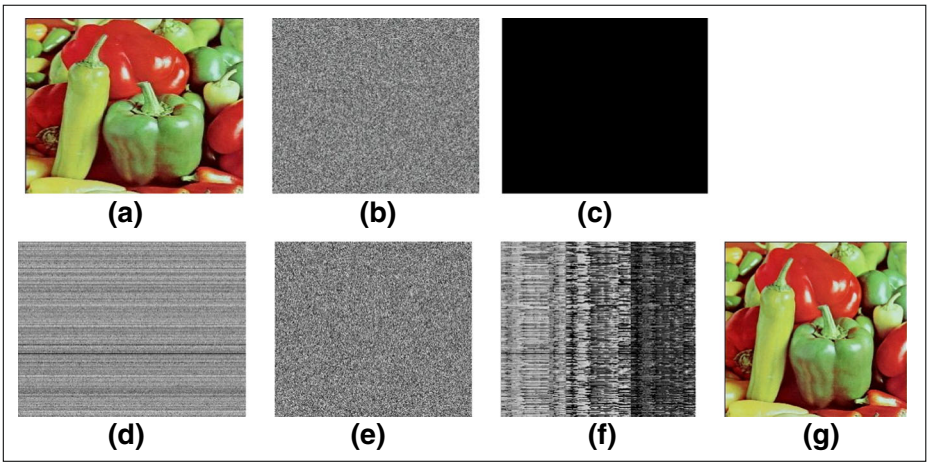

Fig. 11 **a** Original color image, **b** stego encrypted Image; **c** chosen plain zeros of (M*N), **d** shuffle image row; **e** shuffle image column; **f** shuffled image; **g** reconstructed image with correct key

We tested those attacks on selected stego encryption of image **seq.6.** The results are shown in Fig. 13 as a graphical representation, proving that our stego encrypted image has more robustness against these attacks (Table 6).

Finally, we evaluate and compare our proposed method with other similar existing methods according to major essential factors such as PSNR (see Table 7) and MSE (see Table 8) and denote the results as shown in the tables below. These values represented in graphical simulation in Figs. 14 and 15.

A qualitative comparison of these compared methods is listed in Table 9.

Also, a quantities analysis according to specific features like PSNR, SSIM and capacity has been provided in comparison Table 10 between our proposed method and method in [2]. From

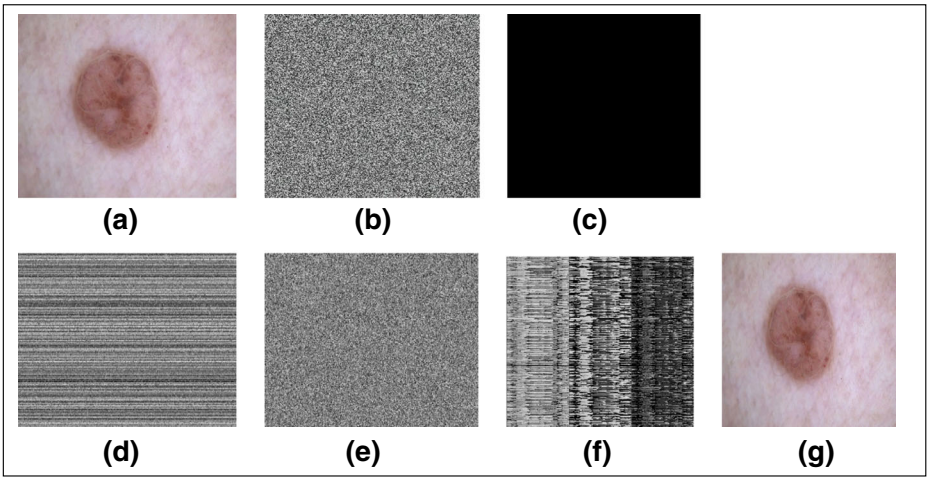

Fig. 12 **a** original color image; **b** stego encrypted Image; **c** chosen plain zeros of (M*N); **d** shuffle image row; **e** shuffle image column; **f** shuffled image; **g** reconstructed image with correct key

this comparison we note that our method exceeds this method according to these enumerated features (Fig. 16).

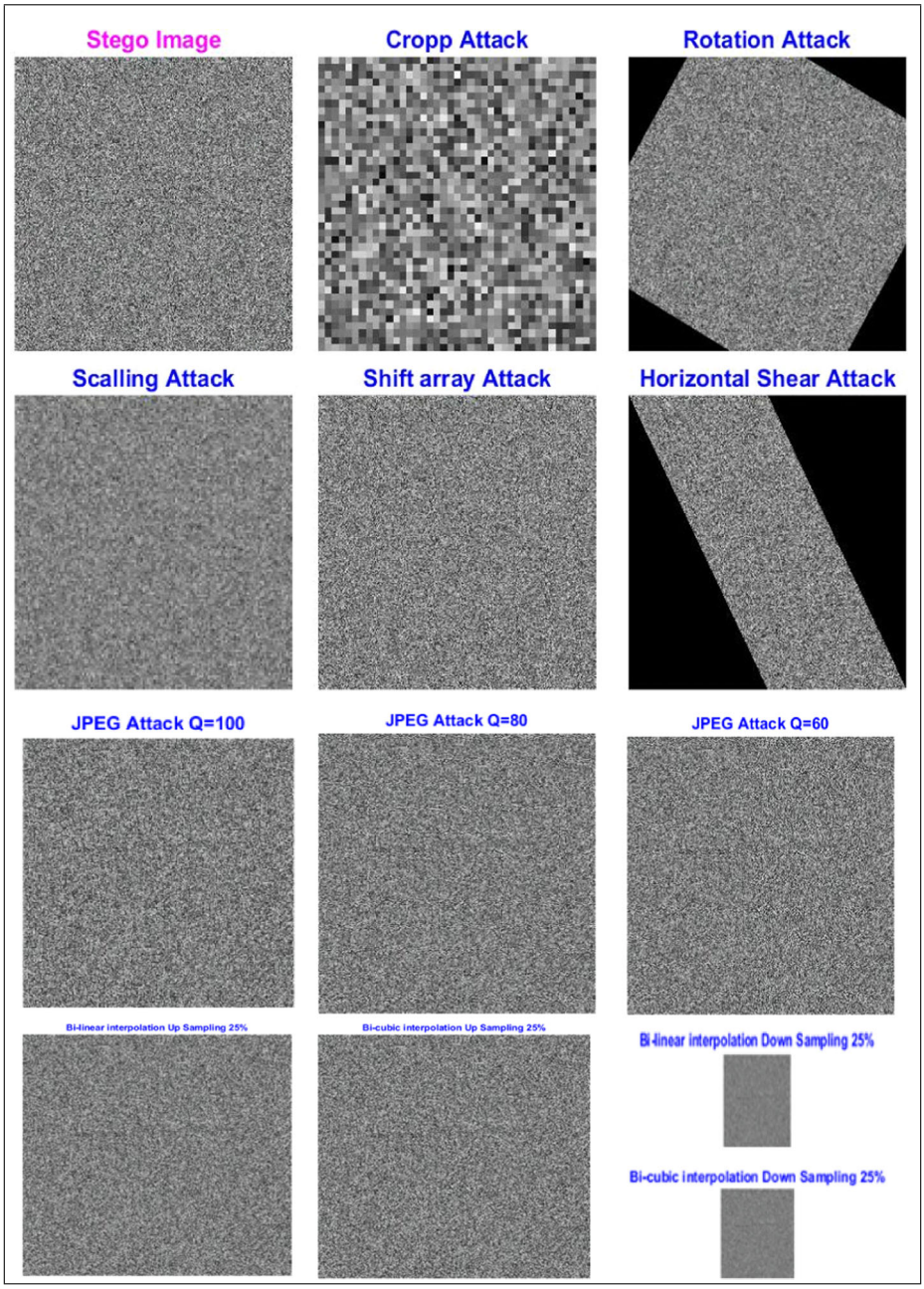

Fig. 13 Different geometric attacks analysis on stego encrypted image of a chosen image Seq.6

Table 6 PSNR and MSE values for a selected images(256*256)with different text size embedding

Images	100 Chars		200 Chars		300 Chars		400 Chars		500 Chars		2050 Chars	
	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE
Seq.1	90.55	0.000041	89.34	0.000063	87.25	0.000083	85.043	0.000097	84.056	0.0000102	67.95	0.010323
Seq.2	80.63	0.001943	79.45	0.0015632	77.34	0.001063	75.0680	0.002048	74.054	0.002159	56.32	0.104571
Seq.3	71.06	0.00513	70.44	0.007249	68.022	0.08937	66.004	0.010458	65.033	0.01096	47.78	0.4174601
Seq.4	70.40	0.005202	68.97	0.010386	67.34	0.011453	65.56	0.023574	64.28	0.054031	46.65	0.434789
Seq.5	73.15	0.002665	72.43	0.002274	70.65	0.006267	68.34	0.013042	66.53	0.023271	49.72	0.175491

Table 7 PSNR comparison between the proposed method and other similar methods

Image name		[17] Mehdi Khalili	[9] Neha Batra et al.	[3] Abdel Wahab et al.	[32] Rasha Thabit et al.	Proposed Method
Seq.1	512*512	60.47	64.1282	–	56.29	85.41
(Lena)	256*256	–	58.2122	–	–	90.27
Seq.2	512*512	60.43	62.2132		55.71	78.25
(Peppers)	256*256	–	56.9715	–	–	80.73
Seq.3	512*512	–	63.0185		55.40	75.24
(Airplane Jet)	256*256	–	55.6473	55.56	–	71.03
Seq.4	512*512	60.49	64.3199	–	52.33	90.01
(Baboon)	256*256	–	58.3766	51.22	–	70.80

6 Conclusion

In this work, we implement a proposed data hiding method to hide a secret message into a cover image. The cover image is scrambled using a chaotic hybrid map to increase robustness against several attacks. The cover image is then transformed from RGB model to YCbCr color model to increase capacity. We chose the luminance channel(Y) to hide data inside it because it is the ideal space for the data hiding process. It is a gray version of the original color image, which helps us reconstruct the cover image again without any problems. We chose this, rather than the (Cb and Cr), which represents important component parts that are more prone to distortion, side-channel attack, and other types of attacks.

The text secret message was compressed using Huffman coding to ensure data integrity during the data hiding process. Also, we worked on a hybrid domain (frequency domain and encrypted domain) to increase stego image robustness. We made this decision to avoid working on a spatial domain. Although easy to implement, a spatial domain changes gray levels of some pixels in the image which are considered important parts of the image and causes distortion. We tested several attacks on the stego image to evaluate its robustness, and the results show that our stego image is more robust against both geometric and signal attacks. The selected encrypted channel(Y) is divided into 8×8 non-overlapping blocks and we applied DCT on each block. We quantized each DCT block using the recommended standard quantization table to preserve the correlation among the image pixels during the data hiding process. This made it possible to easily reconstruct the cover image with minimum distortion.

Table 8 MSE Comparison with other similar methods

Image name		[17] Mehdi Khalili	[9] Neha Batra et al.	[3] Abdel Wahab et al.	[32] Rasha Thabit et al.	Proposed Method
Seq.1	512*512	–	0.0251	–	–	0.000187
(Lena)	256*256	–	0.0981	–	–	0.0000611
Seq.2	512*512	–	0.0250	–	–	0.000973
(Peppers)	256*256	–	0.0931	–	–	0.000183
Seq.3	512*512	–	0.0324	–	–	0.001945
(Airplane Jet)	256*256	–	0.1771	0.895	–	0.005127
Seq.4	512*512	–	0.0240	–	–	0.000065
(Baboon)	256*256	–	0.0944	0.575	–	0.005402

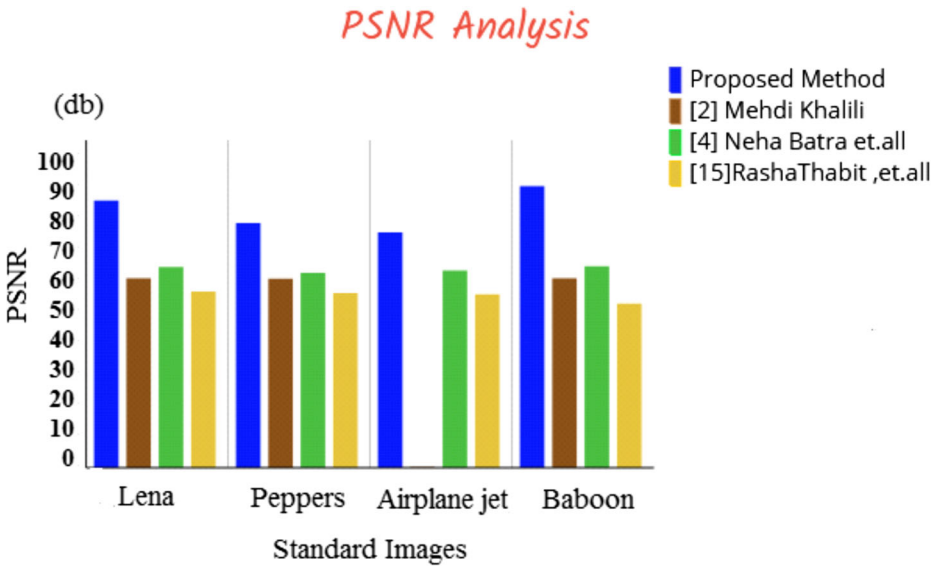

Fig. 14 PSNR comparison analysis on 4 standard images between proposed method, [9, 17, 32]

As we know, MSB embedding strategy gives a good result compared to other similar methods, so we embedded the desired secret data on 8-bits of the quantized version of the DCT coefficients using left-MSB strategy, which increases image quality and capacity rather than traditional LSB strategy with only 4-bits which minimize capacity.

The experimental results findings indicate that the proposed method extracts the hidden secret message correctly without any data lose or visual distortion in the

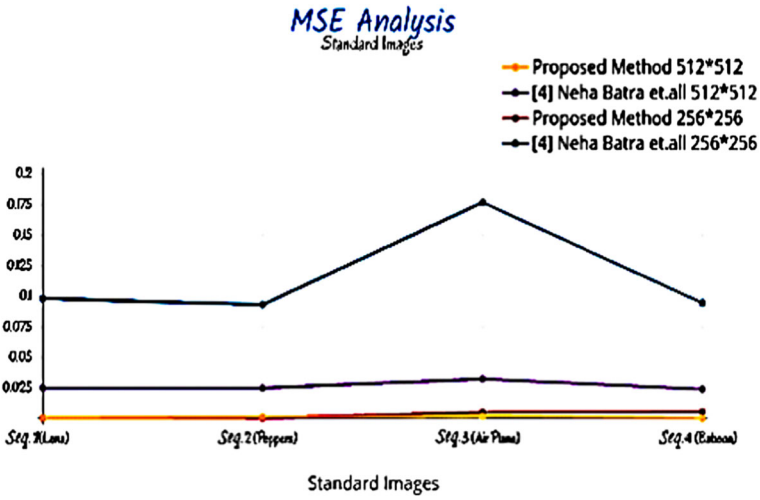

Fig. 15 MSE comparison analysis of proposed method and [9] Neha Batra et al. on selected 4 standard images

Table 9 Qualitative comparison of all compared techniques

Features	[17] Mehdi Khalili	[9] Neha Batra et al.	[3] Abdel Wahab et al.	[32] Rasha Thabit et al.
Working domain	Hybrid domain Frequency DCT domain + Encrypted domain	Frequency DCT domain	Frequency DCT domain	Frequency Slant let transform (SLT) domain
Performance	Good	Good	Good	Good
Robustness	High	Not Mentioned	Medium	High
Capacity	Medium	High	Medium	Medium
PSNR	High	High	High	Medium
MSE	Low	Low	Low	Low
Complexity	Not Mentioned	Not Mentioned	Not Mentioned	Low

Table 10 Quantitative comparison between proposed method and methods in [2, 19]

Features	Image size	Maximum embedding	PSNR dB	SSIM	Capacity
Proposed	256 × 256	16,400 bits	Up to 67.95	0.97968028	2-bit/8-bit
Method [2]	256 × 256	16,384 bits	Up to 44.2314	0.9499	2-bit/8-bit
Method [19]	–	4000 bits	Up to 74.14	–	2-bit/8-bit

recovered image. We evaluated our proposed method according to the parameters PSNR, MSE, SSIM, capacity, correlation, and other settings to ensure image quality. Also, additional image quality metrics such as mean value, standard deviation, entropy, and gradient show that the reconstructed image is more closely identical to the original image. From these evaluation results, we deduce that our proposed method achieves competitive performance compared with other similar methods. The proposed

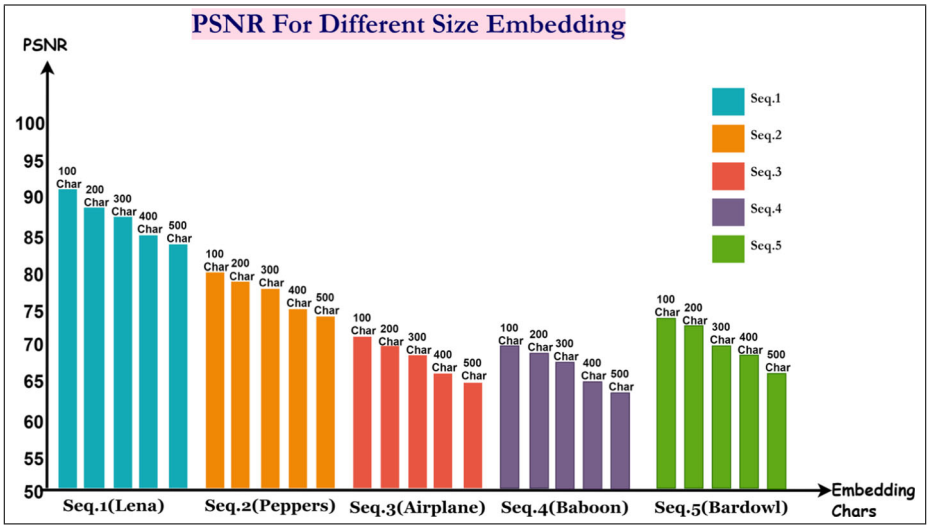

Fig. 16 PSNR analysis for different sizes chars embedding of sizes(100,200,300,400,500)

method can be safely used in telemedicine applications for doctors to diagnose diseases and other similar purposes.

In future work, we want to improve this method to allow any type of data to be concealed inside color images.

References

1. Abd El-Latif AA, Li L, Wang N, Han Q, Niu X (2013) A new approach to chaotic image encryption based on quantum chaotic system, exploiting color spaces. Signal Processing 93:2986–3000 Published by Elsevier BV
2. Abd-El-Atty B, Iliyasu AM, Alaskar H, Abd El-Latif AA (2020) A robust quasi-quantumwalks-based steganography protocol for secure transmission of images on cloud-based e-healthcare platforms. Sensors 20:3108. https://doi.org/10.3390/s20113108
3. Abdel-Wahab OF, Hussein AI, Hamed HFA, Kelash HM, Khalaf AAM, Ali HM (June 2019) Hiding data in images using steganography techniques with compression algorithms. Telkomnika 17(3):1168–1175. https://doi.org/10.12928/TELKOMNIKA.v17i3.12230
4. Alan Anwer Abdulla (October 2015) Exploiting similarities between secret and cover images for improved embedding efficiency and security in digital steganography, School of Science in the University of Buckingham
5. Abdulla AA, Sellahewa H, Jassim SA (2014) Steganography based on pixel intensity value decomposition, mobile multimedia/image processing, security, and applications. Proc of SPIE 9120:912005 · © 2014 SPIE. https://doi.org/10.1117/12.2050518
6. Alan Anwer Abdulla, Harin Sellahewa and Sabah A. Jassim (2019) Improving embedding efficiency for digital steganography by exploiting similarities between secret and cover images, Springer Science and Business Media, LLC, part of Springer Nature, Multimedia Tools and Applications , https://doi.org/10.1007/s11042-019-7166-7
7. Arunkumar S, Subramaniyaswamy V, Vijayakumar V, Chilamkurti N, Logesh R SVD-based robust Image Steganographic scheme using RIWT and DCT for the secure transmission of medical images, S0263. Measure 6413 2241(19):30186–30181. https://doi.org/10.1016/j.measurement.2019.02.069
8. Awad Attaboy A, Mursi Ahmed M, Alsammak AK (December 2018) Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3. Ain Shams Engineering Journal 9(4): 1965–1974. https://doi.org/10.1016/j.asej.2017.02.003
9. Batra N, Kaushik P (October 2012) Data hiding in color images using modified quantization table. International Journal of Advanced Research in Computer Science and Software Engineering 1(8)
10. Brain atlas available at: https://www.osirix-viewer.com
11. Color anatomical images available at: https://www.nlm.nih.gov
12. Dicom images available at: https://www.aycan.de/sample-dicom-images.html
13. Hassan Elkamchouchi, Wessam M. Salama and Yasmine Abouelseoud (2017) Data hiding in a digital cover image using chaotic maps and LSB technique. https://doi.org/10.1109/ICCES.2017.8275302.
14. Nur Imtiazul Haque, Kazi Md. Rokibul Alam, Tasfia Mashiat, and Yasuhiko Morimoto (2018) A technique to enrich the secrecy level of high capacity data hiding steganography technique in JPEG compressed image
15. Kadhim I, Premaratne P, Vial PJ, Halloran B (2018) Comprehensive survey of image steganography: techniques, evaluations, and trends in future research. Neuro computing, [m5G; November 16 3:21. https://doi.org/10.1016/j.neucom.2018.06.075
16. Ke G, Wang H, Zhou S, Zhang H (March 2019) Encryption of medical image with most significant bit and high capacity in piecewise linear chaos graphics. Measurement 135:385–391. https://doi.org/10.1016/j.measurement.2018.11.074
17. Khalili M (December 2015) DCT-Arnold chaotic based watermarking using JPEG-YCbCr. Optik 126(23): 4367–4371. https://doi.org/10.1016/j.ijleo.2015.08.042
18. Khosravi MR, Yazdi M (2018) A lossless data hiding scheme for medical images using a hybrid solution based on IBRW error histogram computation and quartered interpolation with greedy weights. Springer Neural Computing and Applications. https://doi.org/10.1007/s00521-018-3489-y(0123456789
19. Kordov K, Stoyanov B (2017) Least significant bit steganography using Hitzl-Zele chaotic map. International of electronics and telecommunications 63(4):417–422. https://doi.org/10.1515/eletel-2017-0061
20. Yih-Kai Lin (2014) A data hiding scheme based upon DCT coefficient modification Computer Standards & Interfaces 36: 855–862 Volume 36, Issue 5, September 2014, Pages 855–862, https://doi.org/10.1016/j.csi.2013.12.013

21. Lin Y-K (2014) A data hiding scheme based upon DCT coefficient modification. Computer Standards & Interfaces 36:855–862. https://doi.org/10.1016/j.csi.2013.12.013
22. Medical melanoma images (2020) available at: https://www.isicarchive.com/#!/topWithHeader/onlyHeaderTop/gallery
23. Murat karAbatak and Yildiray Yigit (2018) Developing the LSB method using mask in colored images.
24. Puteaux U, Puech W (2018) An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images. IEEE Transactions on Information Forensics and Security. https://doi.org/10.1109/TIFS.2018.2799381
25. Rhouma Rhouma and Safya Belghith (2008) Cryptanalysis of a new image encryption algorithm based on hyper-chaos.
26. Saeed MJ (2013) A new technique based on chaotic steganography encryption text in DCT domain for a color image. Journal of Engineering Science and Technology 8(5):508–520
27. Shiffa Saleem, Dominic Mathew, Thomas A (2017) Secure reversible data hiding in color images using LWT and hyper-chaotic encryption; In: International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT 2017)
28. Sharma VK, Mathur P, Srivastava DK (2019) Highly secure DWT steganography scheme for encrypted data hiding. In: Satapathy S, Joshi A (eds) Information and communication Technology for Intelligent Systems, smart innovation, systems and technologies, vol 106. Springer, Singapore. https://doi.org/10.1007/978-981-13-1742-2_66
29. Parah SA, Ahad F, Sheikh JA, Bhat GM (February 2017) Hiding clinical information in medical images: A new high capacity and reversible data hiding technique. Journal of Biomedical Informatics 66:214–230. https://doi.org/10.1016/j.jbi.2017.01.006
30. Standard dataset images available at http://decsai.ugr.es/cvg/dbimagenes/c512.php
31. Tang Z, Lua Q, Lao H, Yua C, Zhanga X (March 2018) Error-free reversible data hiding with high capacity in the encrypted image. Optik 157:750–760. https://doi.org/10.1016/j.ijleo.2017.11.154
32. Thabit R, EeKhoo B (March 2015) A new robust, lossless data hiding scheme and its application to color medical images. Digital Signal Processing 38:77–94. https://doi.org/10.1016/j.dsp.2014.12.005
33. Sriti Thakur, Amit Kumar Singh, Satya Prakash Ghrera, and Mohamed Elhoseny (2018) Multi-layer security of medical data through watermarking and chaotic encryption for telehealth applications, Springer Science +Business Media, LLC, part of Springer Nature, https://doi.org/10.1007/s11042-018-6263-3
34. Türker T, Yasin S (2017) Block-based data hiding method for images. European Journal of Technic 7(2)
35. Wang D, Chen D, Ma B, Xu L, Zhang J (2016) A high capacity spatial domain data hiding scheme for medical images. J Sign Process Syst, Springer Science +business Media New York. https://doi.org/10.1007/s11265-016-1169-7
36. Zaghbani S, Boujnah N, Bouhlel MS (June 2017) High capacity data hiding scheme for DCT image using the YCbCr color space and chaotic map. Journal of Image and Graphics 5(1). https://doi.org/10.18178/joig.5.1.10-15