Reversible Data Hiding Algorithm for Secured E-Healthcare Applications

Anasua Choudhury, Harsh Vardhan Chaudhry, Arkyajyoti Saha, Thanikaiselvan V.
School of Electronics Engineering, Vellore Institute of Technology, Vellore

anasuachoudhury14@gmail.com,sahaarkajyoti2018@gmail.com, harsh168vardhan@gmail.com

Abstract

With an increase in accessibility of telediagnosis for patients, the emphasis has been brought on the secure communication of patient image or text-based reports to its various remote centres. This makes access to a secure means of patient data communication essential to the e-healthcare framework. The proposed algorithm employs an Integer Wavelet Transform (IWT) and Elliptic Curve Cryptography (ECC) based asymmetric key hill cipher approach and chaotic maps for encryption of the medical images and a histogram-shift based reversible data hiding for secure transmission of the medical images. The patient data is decomposed using IWT before encryption using the ECC key hill cipher algorithm for the first level of encryption and chaotic maps for the second level of encryption. The data transmission is then hidden using histogram-shift based approach before recovery at the end point. In order to measure the efficiency of the proposed algorithm we have employed and verified the values of the entropy, peak to signal noise ratio (PSNR), unified average changing intensity (UACI) against their critical values and the correlation between pixels along horizontal, vertical and diagonal directions is measured.

Keywords: Reversible Data Hiding, Integer Wavelet Transform (IWT), Elliptic Curve Cryptography (ECC), Hill Cipher, Histogram-shift based data Hiding.

1. Introduction

The integration of vast areas of traditional healthcare practices with telemedicine has led to a surge in digitalization of various operations in the healthcare domain. One such digitized operation is the storage and sharing of patient information such as personal details of patients and their recent medical treatments, diagnosis and medical history. An ideal telemedicine

practice in today's e-healthcare framework requires the sharing of patient information to its various remote centres for a complete and efficient healthcare service. This is where the role of secure data hiding techniques become crucial. Reversible data hiding is the hiding of information or data inside of a digital media- images, audios, videos- before transmission of data and also recovery of a distortion free media at the end point when the data is extracted [1]. Before data is hidden, however, the data is encrypted to achieve an added layer of security. Encryption is done in many ways with novel algorithms being introduced time and again. The encryption and decryption algorithm of an image is termed under image cryptography and is a widely researched topic [2, 3]. Data Hiding is widely done on encrypted images which means that the data is encoded such that it can only be accessed by authorised parties using predefined access keys [4]. Reversible data hiding in encrypted images (RDHEI) provides a second barrier against any malicious attacks that attempt to steal or manipulate data. Data Hiding as a separate field has widely been researched with new and efficient techniques surfacing increasingly that use both a mix of traditional data hiding techniques and novel approaches to the data hiding algorithm [5].

Hill Cipher Methods are very often used to build upon its efficiency in encryption algorithms as a cipher system [6-10]. Another very widely used cryptosystem is Pailier's Cryptosystem, a homomorphic cryptosystem alongside another widely used cryptosystem, RSA, which is a public key cryptography (PKC) method [8-13]. Data hiding coupled with encryption techniques provides a vast spectrum of security which can be claimed to be unbreakable. Histogram-shift based data hiding is one of the most commonly used data hiding techniques with an effective data hiding status. Data is hidden between zero and peak pairs for a set threshold on both the positive and negative side. More the negative value of the negative threshold and more the value on the positive threshold side, more is the data embedding capacity [11].

Reserving room before encryption (RRBE) schemes allow pre-processing before the encryption of image and facilitates high embedding capacity [14]. Vacating room after encryption (VRAE) is another popularly used reversible data hiding in encrypted images method but its disadvantage being that most VRAE based reversible data hiding of encrypted images (RDHEI) do not make desirable load which, again, leads to RDHEI methods such as adaptive difference recovery (ADR) to be explored and used to achieve desired payload [15]. Novel techniques based on pixel manipulation and charting are very common as well and widely used [16]. For example, the division of pixels into blocks before further processing and

encryption. This is extensively used in the data hiding method of adaptive block encoding [19]. Prediction Error expansion (PEE) is also widely used in reversible data hiding schemes and is a highly investigated topic that deals with the complexity of pixels and pixel shift [20]. Pixel value prediction (PVO) brings to the next efficient and popularly used reversible data hiding method that sorts the pixels within a block based on their values [21]. One of the simpler reversible data hiding methods is least significant bit (LSB) matching and substitution, popularly used to draw novel data hiding techniques due to its efficiency despite its simplicity [22].

2. Related Works

2.1 Chaotic Maps

Chaotic maps utilize the Chaotic hypothesis on deterministic frameworks whose way of behaving over the long run can be anticipated by hypothesis. The primary thought that is involved here in these map is that brief contrast at the

The start of the planning can bring about an enormous change in the eventual outcome as time increments. In these maps, the vulnerability increments dramatically with time. A few chaotic maps are ready till date and not many of them that are utilized in proposed calculation make sense beneath.

The Chebyshev polynomial map defined as with a degree n > 1 is a chaotic map with its invariant density function. for some positive Lyapunov exponent λ = In n > 0.

The Chebyshev Polynomials are as follows:

$$C_y(d): [-1,1] \rightarrow [-1,1] \quad (1)$$

$$C_y(d) = \cos(y \cos^{-1} d) \quad (2)$$

where y is the degree of polynomial Cy (d). The above polynomials fulfill the following conditions:

$$C_0(d) = 1 \quad (3)$$

$$C_1(d) = \mathrm{d} \quad (4)$$

$$C_y(d) = 2C_{y-1}(d) - C_{y-2}(d), \quad y \geq 2 \quad (5)$$

$$\mathrm{C_{y_1}}\left(\mathrm{C_{y_2}(d)}\right) = \mathrm{C_{y_2}}\left(\mathrm{C_{y_1}(d)}\right) = \mathrm{C_{y_1 y_2}(d)}, \quad \mathrm{y_1, y_2} = 0,1,2,\ldots \quad (6)$$

The map is as follows:

$$y(n+1) = \cos(p * a \cos(k(n)) \quad (7)$$

Where p=0.632, k(1) =0.632;

2.2 Elliptic Curve Cryptography

As the name recommends, ECC is an asymmetric cryptography method in light of uses of the mathematical construction of elliptic curves over finite fields. The algorithm works on the elliptic curve discrete logarithm problem (ECDLP). This cryptography strategy is more difficult to break since there is no known answer for the numerical problem given by the equation delivering the elliptical curve in a graph. Subsequently, just a single way stays for hackers: a brute-power-attack — or an experimentation approach, as such. This intricacy makes ECC safer contrasted with RSA.

$$y^2 = x^3 + ax + b \qquad (7)$$

As ECC — by structure — is safer contrasted with RSA in light of the fact that it offers ideal security with more limited key lengths. Subsequently, it requires a lesser load for network and computational power, which converts into a superior client experience. To give a few numbers, RSA can answer 450 solicitations each second with a 150-millisecond normal reaction time, while ECC takes just 75 milliseconds to answer a similar number of solicitations each second. The Elliptic Curve follows the following equation,

2.3 Trapdoor Function:

A trapdoor function is a function that is easy to enroll in one bearing, yet expected to be difficult to calculate the alternate way (considering to be its opposite) without uncommon information, called the "secret entryway". Hidden entryway capacities are for the most part used in cryptography.

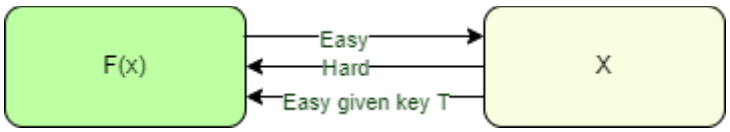


Fig. (1): Trapdoor Function F(x)

2.4. XOR Function:

The bitwise XOR activity is involved here as a piece of a more perplexing encryption algorithm. XOR is utilized as a stage after each significant encryption process in each cycle of the technique to increment the complexity of encryption. Two unique tumultuous arrangements are created. Every one of it is round moved with itself to produce a key. The two such keys created are XOR-ed with the picture lattice to bring about a resultant picture grid.

2.5. Reversible Data Hiding (RDH):

Reversible Data Hiding (RDH) calculations deal with concealing information inside images with the end goal that the first image can be completely recuperated upon the extraction of stowed away information. Concealing data in a picture in a way that doesn't influence the first cover picture pixels or cause a permanent distortion subsequent to removing that data is known as reversible data hiding. It distinguishes data inside a picture such that the original or the primary picture can be totally recovered upon extraction.

2.5.1 Histogram-shift Based Data Hiding:

The histogram-shifting strategy is a very notable technique among reversible information concealing methods. To arrive at its objective of information stowing away, this technique moves all pixel values between the pinnacle and zero focuses and leaves empty spaces for data hiding.

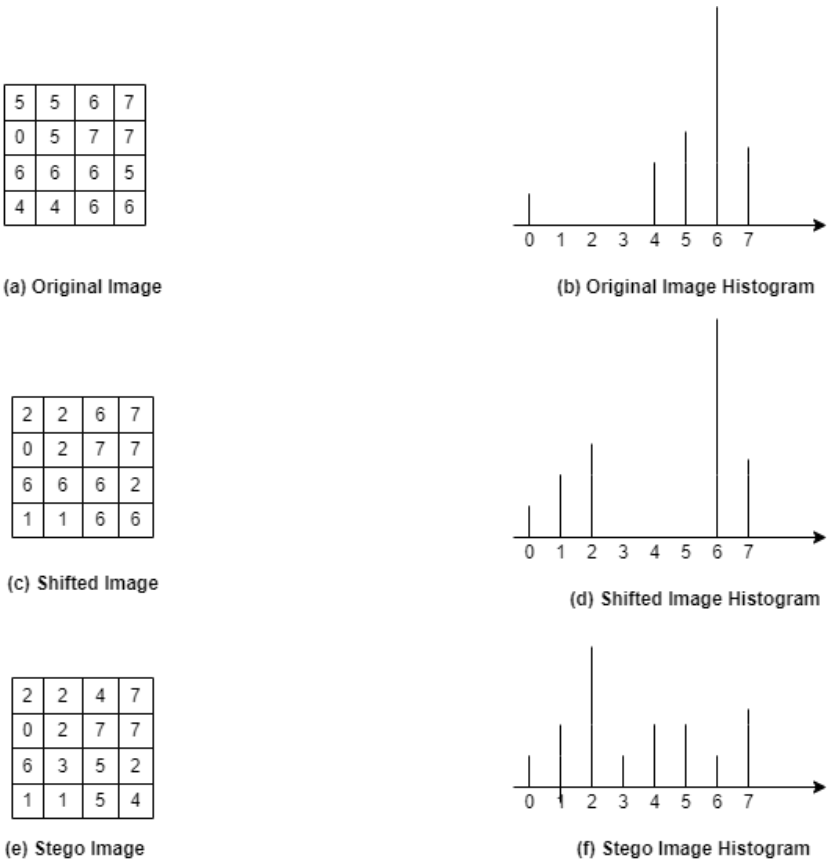


Fig. (3): Histogram-shift based Data Hiding

2.6 Integer Wavelet Transform: The IWT is an invertible integer-to-integer wavelet analysis method. IWT can be used in your desired applications to deliver integer coefficients for integer-encoded signals. Contrasted to the continuous wavelet transform (CWT) and the discrete wavelet transform (DWT), the IWT isn't just computationally quicker and more memory-proficient yet additionally more suitable in lossless data-compression applications. The IWT empowers you to reproduce an integer signal impeccably from the computed integer coefficients.

Integer haar wavelet transform is the simplest among the transforms in the wavelet family. It converts spatial domain coefficients into integer frequency coefficients. The nature of the integer coefficients obtained after decomposing; these transforms are reversible in nature.

$$HF = Codd - Ceven \quad (8)$$

$$F = Ceven + \lfloor HF/2 \rfloor \quad (9)$$

Where,

$Codd$ and $Ceven$ are odd and even column pixels respectively.

$$LH = LF - LF \quad (10)$$
$$LL = LF + LH \quad (11)$$
$$HL = HF - HF \quad (12)$$
$$HH = HF + HL \quad (13)$$

Fig. (2): Integer Wavelet Transform Output on original lena.jpg image

2.7 Multiple Key Hill Cipher:

The main concept behind this encryption technique is based on assigning each letter of the message by a numerical value, for example, a = 0, b= 1, c=3, and so on. This plaintext (message), is then divided into blocks consisting of the same size m depending on the key matrix size m × m. Multiple such keys can be generated using a generator point. The ciphertext is decrypted by using the inverse of the generated key matrices. For example, if the input block size is 2x1 ($M_{2\times 1}$), then key matrix should have a block size of 2x2 ($K_{2\times 2}$), the encryption process will produce the cipher text with the block size of 2x1 ($C_{2\times 1}$) as shown:

$$\text{If } \mathrm{M} = \begin{bmatrix} \mathrm{M1} \\ \mathrm{M2} \end{bmatrix} \text{ and } \mathrm{K} = \begin{bmatrix} \mathrm{K11} & \mathrm{K12} \\ \mathrm{K21} & \mathrm{K22} \end{bmatrix} \quad (14)$$

$$\text{Then } \mathrm{C} = \begin{bmatrix} C1 \\ \mathrm{C2} \end{bmatrix} = \begin{bmatrix} \mathrm{K11} & \mathrm{K12} \\ \mathrm{K21} & \mathrm{K22} \end{bmatrix} \begin{bmatrix} \mathrm{M1} \\ \mathrm{M2} \end{bmatrix} (\mathrm{mod}\ 31) = \begin{bmatrix} (\mathrm{M1}.\mathrm{K11} + \mathrm{K12}.\mathrm{M2}\ ) (\mathrm{mod}\ 31) \\ (\mathrm{M1}.\mathrm{K21} + \mathrm{K22}.\mathrm{M2}) (\mathrm{mod}\ 31) \end{bmatrix} \quad (15)$$

Hence the cipher text block is achieved, to decrypt the block one should produce the inverse of key matrix K^{-1}, ($K.K^{-1}=I$; I is the identity matrix), to decrypt the ciphered text equation below should be used:

$$\mathrm{M} = K^{-1}.C\ (mod\ 31) \quad (16)$$

3. Proposed Steganography and Cryptography Algorithm

The flowchart for the proposed algorithm is as shown in figure 1. The algorithm works in the following steps given below:

Step 1: Haar Integer Wavelet Transform is performed on the image. The high frequency bands are selected accordingly.

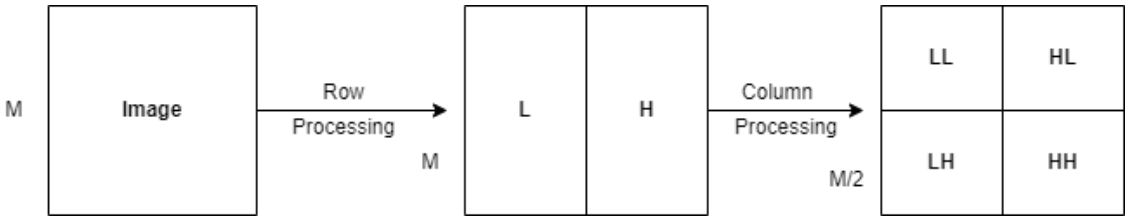


Fig. (4): Integer Wavelet transform decomposition

Step 2: User inputs the threshold value to be used for histogram shifting as shown in Fig. (2)

Step 3: The decomposed image undergoes histogram shifting and maximum possible bits calculated can be stored in the decomposed image.

Step 4: Then reverse Haar Transform is performed and overflow prevention is done in the stego image.

Step 5: The stego image is then moved towards the encryption process. Here two levels of encryption system are used to give more stability.

Step 6: Encryption level 1 is done using Elliptic Curve Cryptography with Multiple Key Hill Cipher. Where the two keys are asked from the user and the image is encrypted using the key matrix obtained. The image is used as the input for encryption level 2.

Step 7: Chebyshev map encryption is used for level 2. The random pattern values are used to perform the XOR operation with the pixels of the input image and the final encrypted image is obtained.

Step 8: Now to reverse the whole process, the decryption level 2 is done for a chaotic map algorithm.

Step 9: Similarly, decryption level 1 is done using the inverse of the key matrix obtained from the Elliptic Curve Cryptography encryption process.

Step 10: The ciphered image is changed back to stego image.

Step 11: We check for the overflow if the overflow is present compensation is done.

Step 12: The image now goes for decomposition through Integer Wavelet Transform and the high frequencies band is chosen.

Step 13: The data is extracted from the bands through reversible histogram shifting.

Step 14: The reverse Integer Wavelet transform is performed on the image to get the retrieved image.

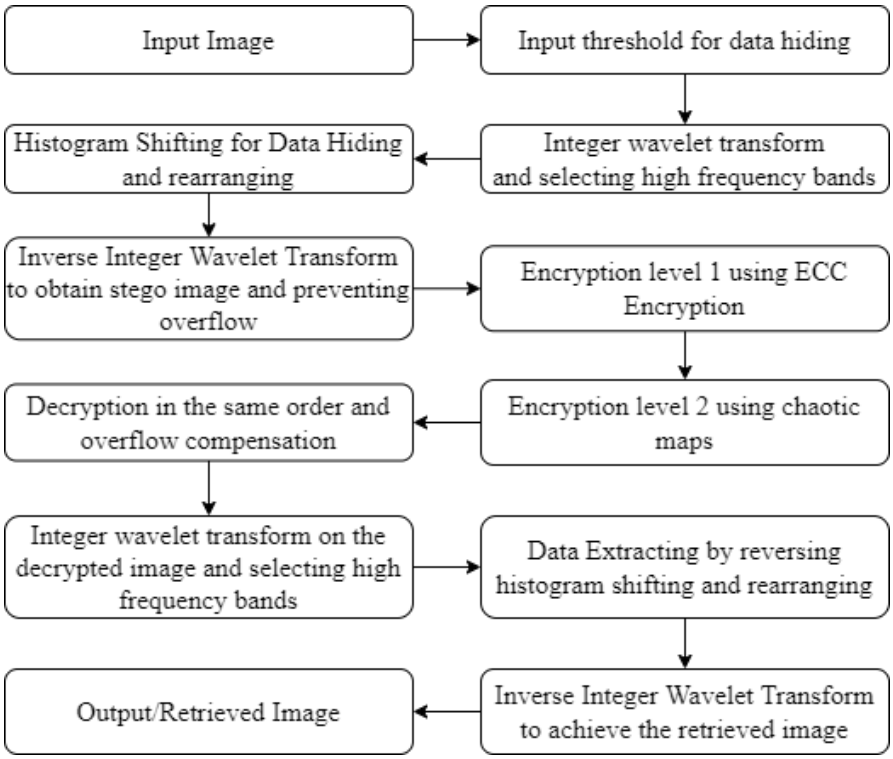


Fig. (5) Flow chart of proposed methodology

4. Results and Discussions

System: intel core (i7), 64- bit OS, 8 GB RAM, 8 test images [23]. The matlab software that has been used to develop and test the algorithm is the recent version MATLAB R2021a.

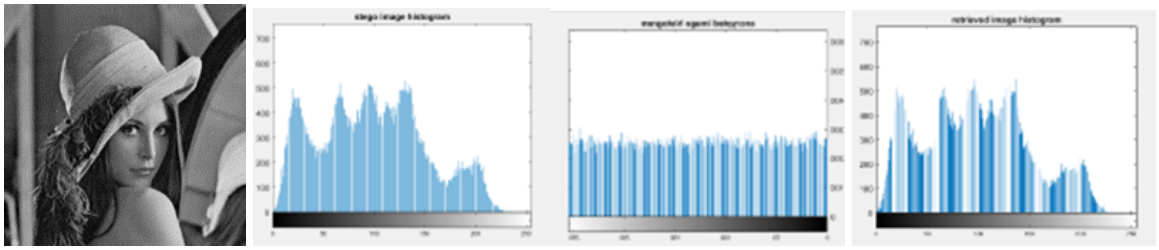
There are not many boundaries to quantify the proficiency of the cryptography utilized. To know the unwavering quality of the cryptosystem comparisons are made between the original and the encrypted image. In this paper various methods are applied to perform Histogram

Analysis, Entropy, Peak Signal to Noise Ratio (PSNR), Unified Average Changing Intensity (UACI) and Noise Pixel Changing Ratio (NPCR) to perform encryption efficiency.

These estimations are portrayed in the following subsections:

4.1 Histogram Analysis

The histogram of original, encrypted and decrypted images over the interval [0-255] is shown. The encrypted image histogram obtained is uniform showing us the level of security it has obtained. Since the histogram of different images is similar it tells us that cryptosystem is working well and making it difficult to crack.



(6.1) (6.2) (6.3) (6.4)

Fig. (6): Experimental results. (6.1) Original image (6.2) Stego Image Histogram

(6.3) Encrypted Image Histogram (6.4) Retrieved Image Histogram.

4.2 Entropy

It estimates the randomness of the image pixel values. More the entropy, lesser the likelihood to see the values. For a plain dark picture, entropy is 0. For a norm 256x256 picture, the ideal and hypothetical worth of entropy is 8

$$E = \sum [P(x)\ 255\ 0 * log2(1\ P(x))] \quad (17)$$

Where P(x) is the probability of the pixel value x and computed by P(x) = the frequency of pixel value x total number of image pixels.

Table (1): Entropy of Encrypted Image and Original Image

Test Images	**Lena.jpg**	**Boat.jpg**	**Bridge.jpg**	**Lighthouse.jpg**	**Baboon.bmp**	**Hill.png**	**Einstein.png**	**Cameraman.gif**
Entropy (En)	7.9975	7.9973	7.9969	7.9967	7.9970	7.9970	7.9973	7.9967

Entropy (Or)	7.5683	7.1782	7.6911	7.5839	6.6962	7.4716	6.8738	6.9046

4.3 UACI & NPCR

Unified Average Changing Ratio and Number of Pixels Change Rate are standards for processing the strength of the encoded pictures from spatial attacks.

Let C1 (i, j) be the pixel value at (i, j) of the original image.

Let C2 (i, j) be the pixel value at (i, j) of the encrypted image.

NPCR & UACI are calculated by using the

$$D(i,j) = \{0, if\ C1(i,j) = C2(i,j)\ 1, if\ C1(i,j) \neq C2(i,j) \quad (18)$$

$$NPCR\ :\ N\ (C1,C2)\ =\ \sum D(i,j)/(\ m\ *\ n\ )*\ 100\% \quad (19)$$

$$UACI: U(C1,C2)\ =\ \frac{1}{W\times H}\sum_{i,j}\frac{|C1(i,j)-C2(i,j)|}{F*m*n} * 100\% \quad (20)$$

In equation (10) and (11) mxm means the total no of pixels where symbol F represents the biggest pixel viable.

NPCR checks the deviation of the generated image from the original image and it ranges from [0,1]. Closer the value is to 1 more the efficiency is.

The average difference between two ciphered images is measured using UACI.

UACI near 30 is considered good for a 256x256 image. Closer the value is to 30, more rigid is the algorithm.

Table (2): NPCR and UACI of Encrypted Images

Test Images	**Lena.jpg**	**Boat.jpg**	**Bridge.jpg**	**Lighthouse.jpg**	**Baboon.bmp**	**Hill.png**	**Einstein.png**	**Cameraman.gif**
NPCR	99.62	99.63	99.63	99.62	99.62	99.63	99.62	99.62
UACI	30.5657	28.3217	29.9265	29.9665	31.9283	29.3672	28.8745	35.1664

4.4 PSNR

Image encryption effectiveness is calculated by PSNR. Distortion is being calculated of the decrypted image with the original image using PSNR. Higher the value the less the distortion.

$$PSNR = 20 * log\ 20[\ 255/MSE] \quad (21)$$

MSE = Mean Square Error between the original and decrypted image

$$MSE = \frac{1}{256 \times 256} \sum_{i=1}^{256} (Aij - Bij) * (Aij - Bij) \quad (22)$$

Aij = pixel value of original image and Bij = pixel value of encrypted image

As MSE increases so the PSNR decreases, showing that the encrypted image is random.

Table (3): PSNR of Encrypted Images

Test Images	**Lena.jpg**	**Boat.jpg**	**Bridge.jpg**	**Lighthouse.jpg**	**Baboon.bmp**	**Hill.png**	**Einstein.png**	**Cameraman.gif**
PSNR	8.5579	9.3076	8.7770	8.7524	8.1541	8.9756	9.1440	7.3391

4.5 Correlation coefficient Analysis

Another common measure used in the assessment of the security level for newly designed image encryption algorithms, is based on the well-known fact that, generally in plain-images, any arbitrarily chosen pixel is strongly correlated with its adjacent pixels (either they are diagonally, vertically or horizontally oriented). Consequently, in the case of high-performance image encryption algorithms, adjacent pixels' correlation scores are expected to be close to zero, i.e., all neighbouring pixels considered in the test are weakly correlated.

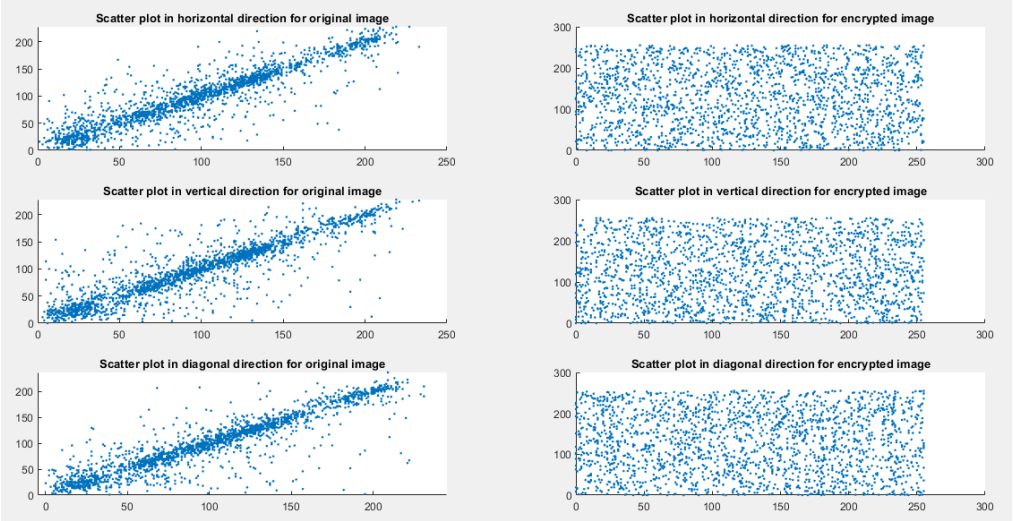


Fig. (7): Experimental results for; Horizontal Correlation-Original Image; Horizontal Correlation-Encrypted Image; Vertical Correlation-Original Image; Vertical Correlation- Encrypted Image; Diagonal Correlation-Original Image; Diagonal Correlation-Encrypted Image.

Table (4): Performance analysis Original and Encrypted Image

Test Images	**Correlation Coefficient**					
	Horizontal Direction		**Vertical Direction**		**Diagonal Direction**	
	Original Image	**Encrypted Image**	**Original Image**	**Encrypted Image**	**Original Image**	**Encrypted Image**
Lena.jpg	0.9231	0.0418	0.8997	0.0076	0.9109	0.0014
Boat.jpg	0.8902	0.0383	0.8379	0.0281	0.8639	0.0143
Bridge.jpg	0.9054	0.0101	0.8257	0.0119	0.8301	0.0136
Lighthouse.jpg	0.7735	-0.0072	0.6775	-0.032	0.6832	-0.0014
Baboon.bmp	0.7995	-0.0154	0.7524	-0.0211	0.7240	-0.0357
Hill.png	0.9222	0.0058	0.8874	0.0085	0.8866	-0.0084
Einstein.png	0.9141	0.0032	0.8706	0.0282	0.8891	-0.0300
Cameraman.gif	0.8971	-9.5737e-04	0.8894	-0.0262	0.8855	0.0104

4.6 Structural Similarity Index (SSIM)

The structural similarity index measure (SSIM) is a method for predicting the quality of various kinds of digital images as well as videos. SSIM is used to measure the similarity of two images. The image quality that is measured is done based on an uncompressed or distortion-free image that is taken as the reference. Hence, it is a full reference metric that requires no less than two images from the very same image capture— a reference image and a processed image.

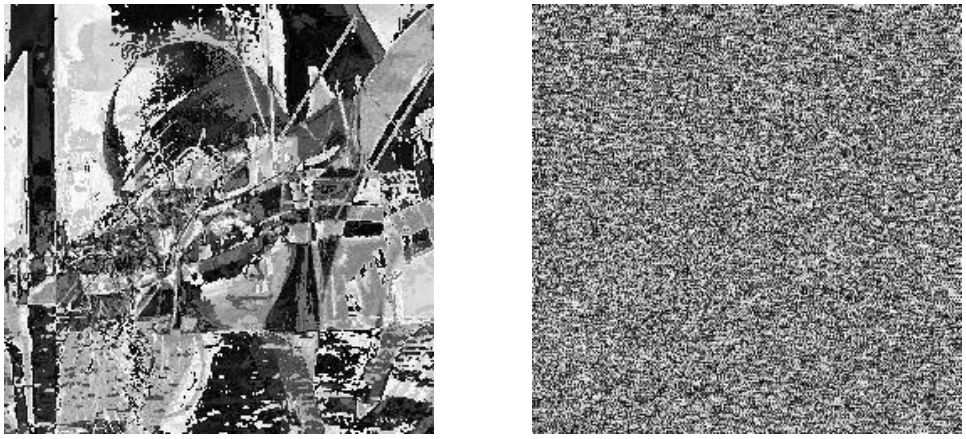
Table (5): SSIM of various test images generated by the proposed model

Test Images	**Lena.jpg**	**Boat.jpg**	**Bridge.jpg**	**Lighthouse.jpg**	**Baboon.bmp**	**Hill.png**	**Einstein.png**	**Cameraman.gif**
SSIM	0.9476	0.9448	0.9383	0.9422	0.9153	0.9301	0.9258	0.9476

4.7 Chosen-Plaintext Attack:

A chosen-plaintext attack (CPA) is a testing model for cryptanalysis which assumes that the attacker can choose random plaintexts that are to be encrypted and obtain the corresponding ciphertexts. The goal of the attacker is to gain access to hidden information which reduces the security of the encryption scheme.

During chosen-plaintext attack, a cryptanalyst can arbitrarily choose plaintext data to be encrypted and receive the corresponding ciphertext. They can try and acquire the secret encryption key, or alternatively, create an algorithm which allows them to decrypt any ciphertext messages encrypted using this key without knowing the actual secret key.



(a) (b)

Fig. (8): Chosen-Plaintext Attack Analysis (a) XOR of lena.jpg and boat.jpg(b) XOR of encrypted lena.jpg and boat,jpg

4.8 Chi-Squared Test:

A chi-square test is a factual test used to contrast noticed outcomes and anticipated outcomes. The motivation behind this test is to decide whether a distinction between noticed information and anticipated that information is expected should risk, or on the other hand assuming it is because of a connection between the factors you are considering.

Formula:

$$x^2 = \sum_{L=0}^{255} \frac{(observed - expected)^2}{expected}$$

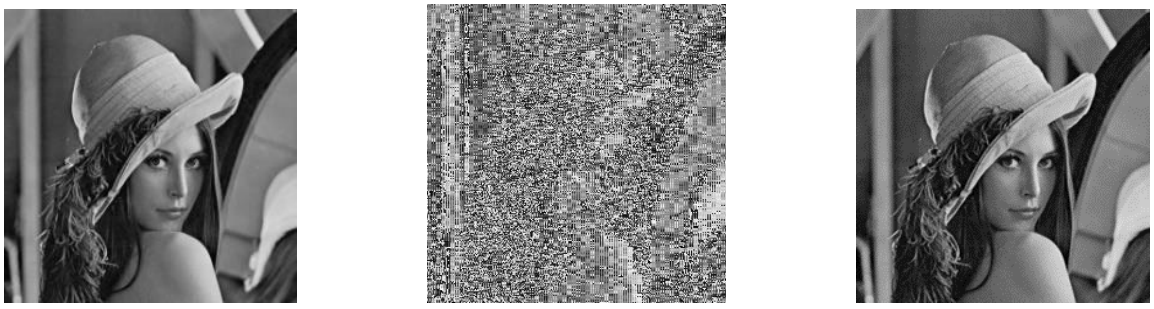
L is here the grey level value.

Table. (6): Chi-Squared Test Analysis for 8 test Images

Test Images	**Lena.jpg**	**Boat.jpg**	**Bridge.jpg**	**Lighthouse.jpg**	**Baboon.bmp**	**Hill.png**	**Einstein.png**	**Cameraman.gif**
Chi-Squared Value	233.4141	279.7969	299.2109	256.8828	256.9297	279.5859	245.4609	247.7891

4.9 Key Sensitivity Analysis

The sensitivity when there is slight change in key used, the algorithm is tested using this analysis. For testing the algorithm data hidden lena test image is utilized for encryption calculation with two different keys K1 and K2 with minor change as displayed in Figure (9).



(a) (b) (c)

Fig. (9): Key Sensitivity Analysis:(a) Original lena.jpg image(b) Decryption fails using wrong keys 5 and 5(c) decrypted Image using correct keys 1 and 2.

Comparative Analysis:

Table (6): Comparative analysis

Lena	**PSNR (Encrypted) (dB)**	**Entropy (encrypted)**	**BPP**

Ref [1]	NA	7.9992	1.711
Ref [18]	51.15	NA	2.628
Ref [19]	43.92	7.9994	1.249
Ref [12]	44.52	NA	1.312
Ref [13]	48.34	NA	1.98
Ref [11]	42.51	NA	2.7669
Ref [7]	43.07	7.9985	NA
Ref [2]	40.086	7.7809	1.6975
Proposed Cryptosystem	42.4927	7.9973	2.5568

Conclusion

The proposed algorithm has been carried out and its performance has been measured and verified against performance measures such as entropy, NPCR, UACI and Correlation coefficient of the pixels in different directions for various cases. This was performed on 8 such images and the results were recorded and compared with each other and the various parameters as mentioned. We concluded a satisfactory and reliable RDH algorithm that can be put to use on medical images to ensure security during transmission.

References:

[1] Zhicheng Ni, Yun-Qing Shi, Nirwan Ansari and Wei Su, IEEE Transactions on Circuits And Systems for Video Technology, Vol. 16, No. 3, March 2006

[2] Ambika Oad, Himanshu Yadav and Anurag Jain, A Review: Image Encryption Techniques and its Terminologies, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-3, Issue-4, April 2014

[3] Prasenjit Kumar Das, Mr. Pradeep Kumar and Manubolu Sreenivasulu, Image Cryptography: A Survey towards its Growth, Advance in Electronic and Electric Engineering. ISSN 2231-1297, Volume 4, Number 2 (2014), pp. 179-184, Research India Publications

[4] Prajakta Jagtap, Atharva Joshi and Shamsundar Vyas, Reversible Data Hiding in Encrypted Images, International Advanced Research Journal in Science, Engineering and Technology Vol. 2, Issue 2, February 2015

[5] Manisha G. Gedam, Shruti M. Rakhunde and Usha P. Kosarkar, Reversible Data Hiding Technique and its Type, a survey, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727 PP 43-48

[6] Liqaa Saadi Mezher and Ayam Mohsen Abbass, Mixed Hill Cipher methods with triple pass protocol methods, International Journal of Electrical and Computer Engineering (IJECE), Vol. 11, No. 5, October 2021, pp. 4449~4457 ISSN: 2088-8708, DOI: 10.11591/ijece.v11i5.pp4449-4457

[7] Fuhu Wu, Xu Zhou Zhili Chen and Baohua Yang, A reversible data hiding scheme for encrypted images with pixel difference encoding, Knowledge-Based Systems 234 (2021) 107583, October 2021

[8] Rupali Bhardwaj and Anjali Singh, An efficient reversible and secure patient data hiding algorithm for E-healthcare, Multimedia Tools and Applications (2021) 80:31687–31703, 19 July 2021

[9] Shuying Xu, Ji-Hwei Horing and Chin-Chen Chang, Reversible Data Hiding Scheme Based on VQ Prediction and Adaptive Parametric Binary Tree Labeling for Encrypted Images, IEEE Access, DOI: 10.1109/ACCESS.2021.3071819, April 2021

[10] Rajaa K. Hasoun, Sameerah Faris Khlebusa and Huda Kadhim Tayyeha, A New Approach of Classical Hill Cipher in Public Key Cryptography, Int. J. Nonlinear Anal. Appl. 12 (2021) No. 2, 1071-1082 ISSN: 2008-6822 (electronic)

[11] Nour Kittawi and Ali Al-Haj, Reversible data hiding using bit flipping and histogram shifting, Multimedia Tools and Applications (2022) 81:12441–12458, February 2022

[12] Wuyue Zhan and Heng Yao, Reversible data hiding for JPEG images with a cascaded structure, IET Image Processing, Wiley, DOI: 10.1049/ipr2.12426, January 2022

[13] Rupali Bhardwaj, Hiding patient information in medical images: an encrypted dual image reversible and secure patient data hiding algorithm for E-healthcare, Multimedia Tools and Applications (2022) 81:1125–1152

[14] Ammar Mohammadi, A general framework for reversible data hiding in encrypted images by reserving room before encryption, J. Vis. Commun. Image R. 85 (2022) 103478

[15] Chunqiang Yu, Xianquan Zhang, Guoxiang Li, Shanhua Zhan and Zhenjun Tang, Reversible data hiding with adaptive difference recovery for encrypted images, Information Sciences 584 (2022) 89–110

[16] V.M. Manikandan and Yu-Dong Zhang, An adaptive pixel mapping based approach for reversible data hiding in encrypted images, Signal Processing: Image Communication 105 (2022) 116690

[17] Shaowei Weng, Ye Zhou and Tiancong Zhang, Adaptive reversible data hiding for JPEG images with multiple two-dimensional histograms, J. Vis. Commun. Image R. 85 (2022) 103487

[18] Xiangguang Xiong, Lihui Wang, Zhi Li, Chen Yea, Yi Chenc, Mengting Fanc and Yuemin Zhub, An adaptive high capacity reversible data hiding algorithm in interpolation domain, Signal Processing 194 (2022) 108458

[19] Kai Gao, Ji-Hwei Horng and Chin-Chen Chang, High-capacity reversible data hiding in encrypted images based on adaptive block encoding, J. Vis. Commun. Image R. 84 (2022) 103481

[20] Abolfazl Kouhi and Mohammad Hossein Sedaaghi, Reversible data hiding based on high fidelity prediction scheme for reducing the number of invalid modifications, Information Sciences 589 (2022) 46–61

[21] Chi-Yao Weng, Hao-Yu Weng and Cheng-Ta Huang, High-fdelity reversible data hiding based on PVO and median preserving, The Journal of Supercomputing (2022) 78:8367–8388

[22] Aditya Kumar Sahu and Gandharba Swain, High fidelity based reversible data hiding using modified LSB matching and pixel difference, Journal of King Saud University – Computer and Information Sciences 34 (2022) 1395–1409

[23] https://sipi.usc.edu/database/database.php?volume=misc