Red Green Blue Image Encryption Based on Paillier Cryptographic System

Conference Paper · May 2018

CITATIONS
0

READS
298

6 authors, including:

Henry Ogworonjo
Howard University
6 PUBLICATIONS **4** CITATIONS
SEE PROFILE

Madiha Gul
Howard University
3 PUBLICATIONS **33** CITATIONS
SEE PROFILE

Mandoye Ndoye
Tuskegee University
20 PUBLICATIONS **118** CITATIONS
SEE PROFILE

Mohamed Chouikha
Prairie View A&M University
35 PUBLICATIONS **124** CITATIONS
SEE PROFILE

Red Green Blue Image Encryption Based on Paillier Cryptographic System

Mamadou I Wade*, Henry C. Ogworonjo†, Madiha Gul †,
Mandoye Ndoye †, Mohamed Chouikha †, Wayne Patterson †

Abstract—In this paper, we present a novel application of the Paillier cryptographic system to the encryption of red green blue (RGB) images. In this method, an RGB image is first separated into its constituent channel images and the Paillier encryption function is applied to each of the channels pixel intensity values. Next, the encrypted image is combined and compressed if necessary before being transmitted through an unsecured communication channel. The transmitted image is subsequently recovered by a decryption process. We performed a series of security and performance analyses to the encrypted and recovered images in order to verify their robustness to security attacks. The results show that the proposed image encryption scheme produces highly secured encrypted images.

Keywords—Image Encryption, Paillier Cryptographic System, RGB Image Encryption, Paillier.

I. Introduction

THERE are several schemes that have been proposed in the literature for image encryption. For instance, in [10] Singh et al. proposed an Elliptic Curve Cryptosystem that was applied to a group of pixels to obtain a corresponding cipher-image. Also, chaos-based image encryption schemes are proposed in the literature by many authors [1], [8], [12]. For instance, Pareek et al. [12] proposed an encryption scheme that uses two chaotic maps which are initialized using an 80-bit encryption key. In [5], G. Ye et al. proposed an image encryption scheme based on Autoblocking and Electrocardiography to generate the initial keys and remove the need for manual assignment. In [13], P. P. Dang and P. M. Chau discussed an image encryption scheme which used block cipher Data Encryption Standard (DES) to encrypt an image, and Discrete Wavelet Transform (DWT) for the image compression, before producing a secure image. In [14], R. Tao, X. Meng, and Y. Wang proposed using Multiorders of Fractional Fourier Transforms to encrypt images. The image is encrypted using the summation of different orders inverse discrete Fractional Fourier transform (FRFT) of the interpolated sub-images. In addition to image encryption, the need to protect the images, especially confidential ones, is a common theme in many image applications like satellite imaging, medical imaging, and fingerprint imaging.

M. Wade, H. Ogworonjo, M. Gul, M. Chouikha, and W. Patterson are with the Department of Electrical Engineering and Computer Science, Howard University, Washington, DC. e-mail: wademamadoui@gmail.com

M. Ndoye is with the Department of Electrical, Tuskegee University, Tuskegee, AL.

Manuscript received August 18, 2017; revised August 28, 2017.

In this paper, we investigate this important image security problem. We propose an image encryption approach that uses Paillier cryptographic scheme to encrypt and decrypt images in the visible electromagnetic spectrum range. It should be noted that the Paillier-based encryption scheme is a probabilistic public key algorithm [2], [15], [21]. The rest of the paper is organized as follows: in Section II, we present the proposed approach for the image cryptography. Section III gives the results obtained from different security and performance analyses. We conclude in Section IV.

II. Proposed Image Cryptographic Approach

This section discusses the proposed image encryption. It's a three step process. The first step is the key generation step. This step shows how the public and private keys needed for the encryption and decryption respectively are generated. The second step involves the use of the Paillier encryption function and the public key to encrypt each of the pixel's intensity values from the R, G, and B-Channel images. Finally, the private key and the paillier decryption function are used to decrypt each of the encrypted pixel's intensity values for each channel, and to recover the corresponding images.

A. Paillier Public and Private Keys Generation

For the generation of the public and private keys, two large prime numbers p and s are randomly chosen such that the greatest common divisor (gcd) of ps and $(p-1)(s-1)$ is 1, that is

$$\gcd(pq, (p-1)(s-1)) = 1 \tag{1}$$

When the length of p and s are the same, the above gcd property is satisfied. Next, we compute the keys

$$N = p \times s. \tag{2}$$

$$\lambda = lcm(p-1, s-1), \tag{3}$$

where lcm represents the least common multiple. A random integer $g \in Z^*_{N^2} = \{1, 2, \ldots, (N^2-1)\}$, is selected such that the order l of g is a multiple of N, that is $g^l \equiv 1(\mod N)$, where the symbol $\equiv$ represents the congruent relationship. Also, to ensure that the order l of g is a multiple of N, one

can verify that the modular multiplicative inverse in Eq. (4) exits.

$$\mu = L(g^{\lambda} \mod N^2)^{-1} \mod N, \tag{4}$$

where

- $\mod$ is the modulo operator.
- the function L is

$$L(U) \triangleq \frac{(U-1)}{N} \tag{5}$$

Now, the public key needed to encrypt a pixel's intensity value y is given by

- (N, g),

while the privation key that will be used to decrypt the encrypted quantity C is

- (λ, μ).

It is important to note that when p and s have the same length, the public and private key parameters can be computed as

$$g = 1 + N \tag{6}$$

and

$$\lambda = \phi(N) \tag{7}$$

where

$\phi(N) = (p-1)(s-1)$

is the Euler's function defined as the number of positive integers less than or equal to N and relatively prime to N,

and μ is the multiplicative inverse of $\phi(N) \mod N$ given by

$$\mu = \phi(N)^{-1} \mod N \tag{8}$$

B. Paillier Based RGB Image Encryption Phase

Let $y_{_R}$, $y_{_G}$, and $y_{_B}$ be the intensity values of pixels from the R, G, and B-Channel images. To encrypt each of these pixels' intensity values using Paillier encryption function, one can write:

$$E(y_{_R}) = g^{(y_{_R})} x^N \mod N^2 \tag{9}$$

$$E(y_{_G}) = g^{(y_{_G})} x^N \mod N^2 \tag{10}$$

$$E(y_{_B}) = g^{(y_{_B})} x^N \mod N^2 \tag{11}$$

where x is a random number such that
$x \in Z_N^* = \{1, 2, \ldots, (N-1)\}$.
Assume that we have an 8-bit image for each of the R, G, and B-Channel with intensity values in the range $[0, (L-1)] = [0, 255]$, where L is the number of pixels' intensity levels. The encrypted pixels' intensity values $E(y_{_R})$, $E(y_{_G})$, and $E(y_{_B})$ are out of the range $[0, (L-1)]$ and must be mapped back into this range by applying $\mod p$ to each. We can choose $p = 257$, the smallest prime number greater than L. After this mapping, a pixel intensity value of 256 can be obtained, but it is unlikely because of the associated probability of $\frac{1}{256}$. Even if this mapping back to Z_p by applying $\mod p$ to $E(y_{_R})$, $E(y_{_G})$, or $E(y_{_B})$ produces a pixel's intensity value of 256, there would be no visible effect on the associated recovered images because of redundancies and other factors [11].
Now, one can compute the encrypted pixels intensity values mapped to Z_p that will be transmitted or stored as follows:

$$C_{_R} = E(y_{_R}) \mod p = [g^{(y_{_R})} x^N \mod N^2] \mod p \tag{12}$$

$$C_{_G} = E(y_{_G}) \mod p = [g^{(y_{_G})} x^N \mod N^2] \mod p \tag{13}$$

$$C_{_B} = E(y_{_B}) \mod p = [g^{(y_{_B})} x^N \mod N^2] \mod p \tag{14}$$

Other quantities needed to reconstruct $E(y_i)$ from C_i for $i = R, G, B$ at the receiver side during decryption must also be computed and are given by

$$q_{_R} = \left\lfloor \frac{E(y_{_R})}{p} \right\rfloor \tag{15}$$

$$q_{_G} = \left\lfloor \frac{E(y_{_G})}{p} \right\rfloor \tag{16}$$

$$q_{_B} = \left\lfloor \frac{E(y_{_B})}{p} \right\rfloor \tag{17}$$

where the symbol $\lfloor \ \rfloor$ represents the floor function. The quantities in Eqs. (15), (16), and (17) are not secret but can also be encrypted to increase the security of the cipher images.

C. Paillier Based RGB Image Decryption Phase

For decryption purposes, assume the quantities $C_{_R}$, $C_{_G}$, and $C_{_B}$ are available to the decryption function in addition to $q_{_R}$, $q_{_G}$, and $q_{_B}$. Before applying the Paillier decryption function, one must first reconstruct $E(y_{_R})$, $E(y_{_G})$, and $E(y_{_B})$ as follows

$$E(y_{_R}) = q_{_R} \times p + C_{_R} = \alpha \tag{18}$$

$$E(y_{_G}) = q_{_G} \times p + C_{_G} = \beta \tag{19}$$

$$E(y_{_B}) = q_{_B} \times p + C_{_B} = \gamma \tag{20}$$

Now, the Paillier decryption function can be applied to $E(y_{_R})$, $E(y_{_G})$, and $E(y_{_B})$ in Eqs. (18), (19), and (20) to recover the original pixels' intensity values $y_{_R}$, $y_{_G}$, and $y_{_B}$ as shown

$$y_{_R} = \frac{L(\alpha^{\lambda} \mod N^2)}{L(g^{\lambda} \mod N^2)} \mod N \tag{21}$$

$$y_{_G} = \frac{L(\beta^{\lambda} \mod N^2)}{L(g^{\lambda} \mod N^2)} \mod N \tag{22}$$

$$y_B = \frac{L(\gamma^{\lambda} \mod N^2)}{L(g^{\lambda} \mod N^2)} \mod N \tag{23}$$

For software implementation purposes, it is more efficient to use matrices of pixels' intensity values instead of individual pixels.

The block diagram shown in Figure 1 summarizes the steps of our proposed algorithm for the Paillier based image encryption.

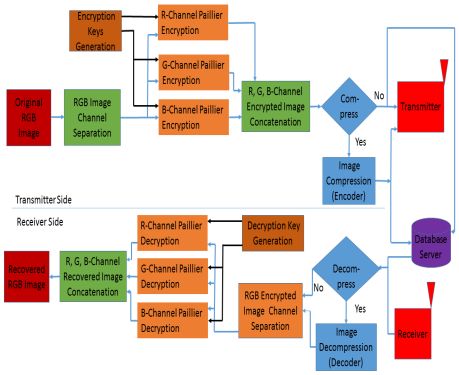


Fig. 1: Proposed Image Cryptographic Scheme

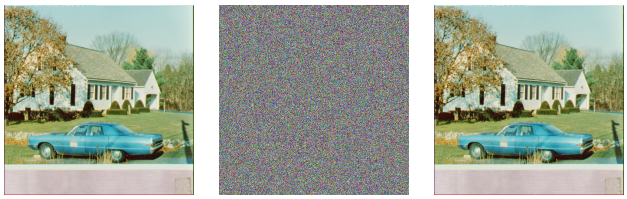
III. SECURITY AND PERFORMANCE ANALYSES RESULTS

A security and performance analysis is performed after implementing the proposed Paillier based image cryptographic scheme discussed in Section II; the implementation code is written using Mathematica 10 software. The images used for the simulation are obtained from the University of Southern California, Signal and Image Processing Institute [19].

We randomly generate the private keys p and s with the same number of digits equal to 100, and the same number of bits of 332 as shown
$p = 814162845482716092464973849579305015007216505536828823638073675453943628217744462074414844021 3280447.$
$s = 692800711178985292011610832678385114999940903671219520087247540278127628826189920279158809915365401.$
The public key used for encrypting the images is $N = p \times s$, and it has 200 digits and 664 bits as shown
$n = 564052598365932021714603554919749640536960000550844476022195943612150045941503834869270660461390508433944183311224421897700614828285917148823640160313774989247295635957084603736171912194771903690142 47.$

A. Plain and Cipher-Images

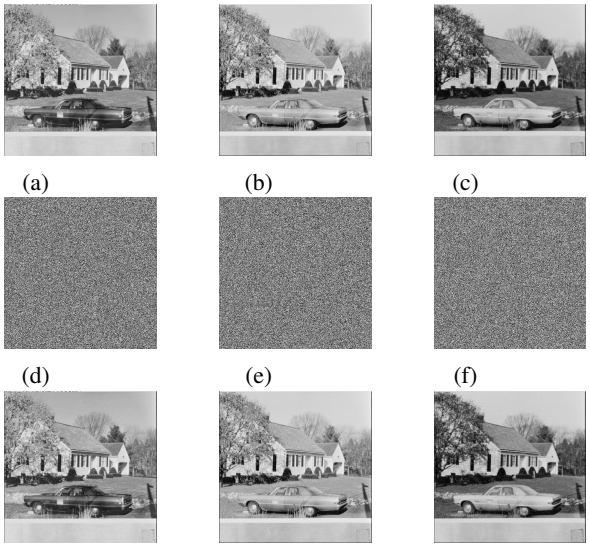
The original RGB image is shown in Figure 2(a), after separating its R, G, and B-Channel images, then encrypting and combining them, we obtained the encrypted RGB images shown in Figure 2(b), while the recovered RGB image is shown in Figure 2(c). Figures 3(a), (d), and (g) show the



(a) (b) (c)

Fig. 2: (a) Original House RGB Image;
(b) Encrypted House RGB Image;
(c) Recovered House RGB Image

R-Channel original image, encrypted image, and recovered image, respectively. Similarly, Figures 3(b), (e), and (h) correspond to the G-Channel original, encrypted, and recovered images, respectively; while Figures 3(c), (f), and (i) provide the B-Channel, original, encrypted, and recovered images, respectively. Figures 2 and 3 confirm visually that our proposed image encryption scheme is able to produce cipher images; which are evaluated and tested further as described in sections that follows.



(a) (b) (c)
(d) (e) (f)
(g) (h) (i)

Fig. 3: (a) Original R-Channel House Image R;
(b) Original G-Channel House Image G;
(c) Original B-Channel House Image B;
(d) Encrypted House Image R;
(e) Encrypted House Image G;
(f) Encrypted House Image B;
(g) Recovered R-Channel House Image R;
(h) Recovered G-Channel House Image G;
(i) Recovered B-Channel House Image B

B. Correlation Analysis

A correlation analysis comparing the correlation coefficients between the original RGB images , each of the channel images and their corresponding cipher-images is performed.This analysis verifies the ability of the proposed encryption scheme to break the correlation of adjacent pixels in the horizontal and vertical directions. An expression for the correlation coefficient in terms of the covariance and standard deviation is as shown" [1], [3], [4], [6], [7], [8], [9], [10], [17], [18]:

$$r_{XY} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \tag{24}$$

where $\sigma_X \sigma_Y$ is product of the standard deviation of X and Y, and $cov(X,Y)$ is the covariance of X and Y. The covariance and standard deviation for pairs of adjacent pixels have the following forms

$$cov(X,Y) = \frac{1}{N}\sum_{i=1}^{N}[(x_i - \mu_X)(y_i - \mu_Y)] \tag{25}$$

where x_i and y_i are values of adjacent pixel pairs selected at random, N the total number of adjacent pixel pairs (x_i, y_i) from the image, μ_X and μ_Y are the mean or expected values of X and Y, respectively, and are given by

$$\mu_X = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \text{and} \quad \mu_Y = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{26}$$

Figures 4 (a), (b), and (c) show the original RGB image, and a scatter plot of its highly correlated random adjacent pairs of 3000 pixels in the horizontal and vertical directions, respectively. The corresponding encrypted RGB image, its horizontal and vertical less correlated scatter plots are shown in Figures 4 (d), (e), and (f), respectively. Similarly, Figures 4 (g), (h), and (i) show the R-Channel original image, its highly correlated plots of 3000 random adjacent pixel pairs in the horizontal and vertical directions, while Figures 4 (j), (k), and (l) give the corresponding encrypted R-Channel image, and its less correlated plot in the horizontal and vertical directions. The correlation coefficients for the RGB and R-Channel encrypted and non-encrypted images in Figure 4 are shown in Table I, and they show that the encrypted RGB image and its associated encrypted R-Channel image have adjacent pairs of pixels that are uncorrelated in the horizontal and vertical directions because their corresponding correlation coefficients are close to zero. Therefore, our proposed images encryption scheme is able to break the correction of adjacent pairs of pixels in the horizontal and vertical directions, leading to production of encrypted images that are resistant to correlation analysis attacks. Similar horizontal and vertical correlation analysis results are also obtained for the G and B-Channel original and encrypted images.

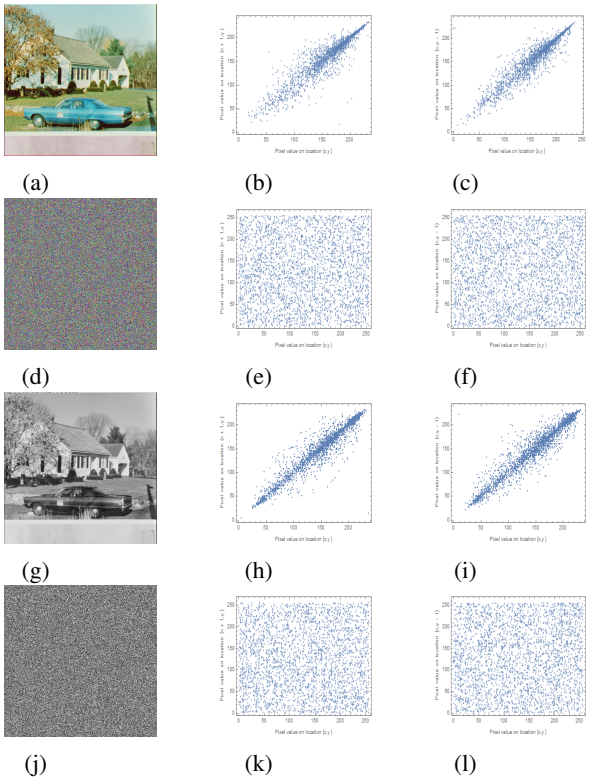


Fig. 4: (a) Original House RGB Image;
(b) Horizontal Correlation of Original House RGB Image ;
(c) Vertical Correlation of Original House RGB Image;
(d) Encrypted House RGB Image ;
(e) Horizontal Correlation of Encrypted House RGB Image;
(f) Vertical Correlation of Encrypted House RGB Image ;
(g) Original House R-Channel Image ;
(h) Horizontal Correlation of Original House R-Channel Image ;
(i) Vertical Correlation of Original House R-Channel Image ;
(j) Encypted House R-Channel Image ;
(k)Horizontal Correlation of Encrypted House R-Channel Image ;
(l) Vertical Correlation of Encrypted House R-Channel Image
N = 3000 Adjacent pixel Pairs

C. Histogram Analysis

A histogram analysis of our original images and their corresponding cipher-images is performed in order to verify that our proposed images encryption scheme is able the produce cipher-images with uniform distribution. Given an images with L intensity levels in the interval $[0, (L-1)]$, its histogram is given by the discrete function $h(k) = n_k$, where $k = 0, 1, 2, \ldots (L-1)$ is the k^{th} intensity level, and n_k represents the number of pixels with intensity level k [16].
Figures 5 (a), (b), and (c) are the R, G, B-Channel original images, with their corresponding non-uniform histograms in Figures 5 (d), (e), and (f), respectively. Figures 5 (g), (h), and (i) are the R, G, B-Channel encrypted images, with their corresponding uniform histograms in Figures 5 (j), (k), and (l), respectively; while Figures 6 (a), (b), (c), and (d) show our original RGB image with its encrypted version and their corresponding histograms, respectively. Since the histograms of all associated encrypted images have uniform distributions, one can conclude that our Paillier based image encryption scheme

TABLE I: Correlation Coefficients of Adjacent Pixels for House Image

RGB Image		
Direction	Non-Encrypted	Encrypted
Horizontal	0.942128	0.0136684
Veritical	0.943238	0.00975569
R-Channel Image		
Direction	Non-Encrypted	Encrypted
Horizontal	0.958567	−0.00643711
Veritical	0.958223	0.0509546

is able produce encrypted images with uniform distribution, and therefore, can protect against histogram analysis attacks.

D. *Cipher Cycle*

A good image encryption scheme produces cipher-images that are very different from their original plain-images. One way to quantify this difference is to use the the Number of Pixel Change Rate (NPCR) and the Unified Average Change Intensity (UACI). The expression for the NPCR measures the average number of pixels in difference of a color component between two images C and C', is given by [4], [8], [9], [17], [18]:

$$NPCR_{R,G,B} = \frac{\sum_{i,j} D_{R,G,B}(i,j)}{N} \times 100\% \qquad (27)$$

where

- N is the image's total number of pixels and

$$D_{R,G,B}(i,j) \triangleq \begin{cases} 0, & if \quad C_{R,G,B}(i,j) = C'_{R,G,B}(i,j) \\ 1, & if \quad C_{R,G,B}(i,j) \neq C'_{R,G,B}(i,j) \end{cases} \qquad (28)$$

- $C'_{R,G,B}(i,j)$ and $C_{R,G,B}(i,j)$ represent the values of corresponding color component R, G, and B in images C and C', respectively.

Given a random RGB images, where each of the R, G, and B channel image is encoded using 8 bits, one can show that the expected value of the NPCR for each channel is given by

$$NPCR_R = NPCR_G = NPCR_B = 99.609375\% \qquad (29)$$

The expression for the $UACI$ is defined as

$$UACI_{R,G,B} = \frac{1}{N}\left[\sum_{i,j} \frac{|C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)|}{2^{L_{R,G,B}} - 1}\right] \times 100\% \qquad (30)$$

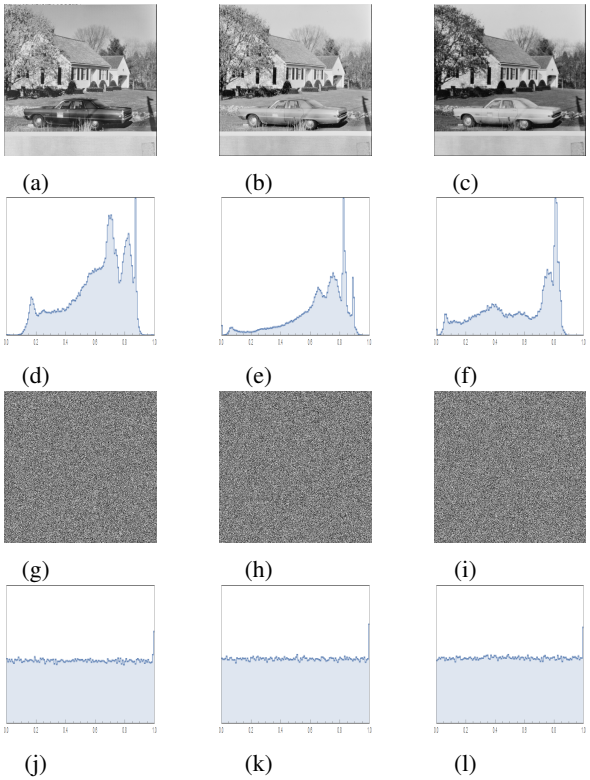


Fig. 5: (a) R-Channel Original Image;
(b) G-Channel Original Image ;
(c) B-Channel Original Image ;
(d) Histogram of R-Channel Original Image ;
(e) Histogram of G-Channel Original Image ;
(f) Histogram of B-Channel Original Image ;
(g) R-Channel Encrypted Image ;
(h) G-Channel Encrypted Image ;
(i) B-Channel Encrypted Image ;
(j) Histogram of R-Channel Encrypted Image ;
(k) Histogram of G-Channel Encrypted Image ;
(l) Histogram of B-Channel Encrypted Image

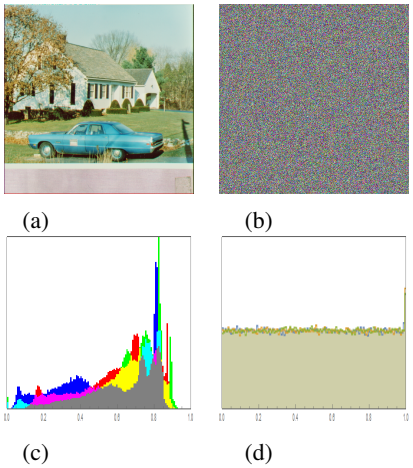


Fig. 6: (a) Original House RGB Image;
(b) Encrypted House RGB Image;
(c) Histogram of Original RGB Image;
(d) Histogram of Encrypted RGB Image

where $L_{R,G,B}$ represents the number of bits used for each color component of Red (R), Green (G), or Blue (B), respectively. One can also show that the expected value of a random RGB image encoded using 8 bits for each channel is given by

$$\mathbb{E}(UACI_R) = \mathbb{E}(UACI_G) = \mathbb{E}(UACI_B) = 33.46354\% \quad (31)$$

Table II shows the NPCR and UACI values for each of the R, G, and B-Channel images. For instance, the NPCI between R-Channel original and encrypted images is 99.6223%, which is close to the expected values of 99.60937%; while the UACI value of 30.2723% is also close to the expected value of 33.4635%. Therefore, our proposed encryption scheme performed very well on producing encrypted pixel intensity values, and average pixel encrypted intensity values, that are different compared to their corresponding non-encrypted versions. Similar NPCR and UACI analyses apply to the G, B-Channel, and RGB image, and results are shown in Table II.

TABLE II: NPCR and UACI for House R, G, and B-Channel Images

R, G, and B-Channel		
Test Type	NPCR (%)	UACI (%)
Original R and Encrypted R	99.6223	30.2723
Original G and Encrypted G	99.6002	31.2944
Original B and Encrypted B	99.609	31.2141
Orig RGB and Encrypt RGB	100	30.3, 31.3, 31.2
Expected Value	99.60937	33.4635

E. Information Entropy

Information entropy can be used to evaluate the degree of uncertainty in a random variable. The entropy $h(S)$ of a source S with N symbols, where $N = 2^k$, and k the number of bits used to represent a symbol S_i, can be obtained as follows [1], [4], [9], [16], [17], [20]:

$$h(S) = -\sum_{i=0}^{N-1} p(S_i) \log_2[p(S_i)] \quad (32)$$

where $p(S_i)$ is the probability of occurrence for the symbol S_i, N the total number of symbols generated by the source, and the $\log_2$ expresses the entropy in bits.
When S is a truly random source, the probability of symbol S_i is $p(S_i) = \frac{1}{2^k}$ for all i, and the Entropy of the source S is found to be

$$h(S) = k \quad (33)$$

For an RGB truly random image where each channel is encoded using 8-bit, the entropy is found to be $h(S) = 8$.
The entropy analysis of our R, G, and B-Channel encrypted images is performed and results are shown in Table III. These entropy values for the channels are very close to the theoretical value of 8, meaning that the encrypted pixels intensity values are truly random; therefore, our proposed image cryptographic scheme produces secure encrypted image that can resist entropy attacks.

TABLE III: Entropy Analysis for House R, G, and B-Channel images

Encrypted Images			
R	G	B	Expected Value
7.94279	7.94323	7.94323	8

F. Recovered Image Quality Analysis

It is important to provide an analysis of the quality of the recovered images in order to find out how much information is lost as a result of applying our encryption, decryption, and other processing actions to the original images. Results of this analysis is shown in Table III, where the quality is label Low when the NPCR value is greater than 1%, High when the NPCR value is between $0.01\%$and 1%; Very High when the NPCR value is less than 0.01%, meaning that more than 99.99% of the pixels' intensity in the recovered and original images are the same and very little information is lost; for this case, the original and recovered images are the same with the human eyes. The quality of the recovered RGB image is low because of limited Salt-and-Pepper noise. But this noise is not present for some simulation runs.

TABLE IV: NPCR and UACI for House R, G, and B-Channel Images

R, G, and B-Channel		
Test Type	NPCR (%)	Quality
Original R and Recovered R	0.402451	High
Original G and Recovered G	0.775909	High
Original B and Recovered B	0.584412	High
Orig RGB and Encrypt RGB	1.5583	Low

G. Image Compression Results

The encrypted R, G, and B-Channel images are combined to form one RGB encrypted image that can be compressed

before transmission through a channel or stored into a file. Compressing the encrypted RGB image before transmission has many advantages such as reducing transmission bandwidth and time, as well as reducing storage capacity for future usage. After compressing the RGB encrypted image, the number of bytes are reduced by a factor 2.92 as shown in Table V.

TABLE V: Image Compression Results

Image			
Encrypted	Before (Bytes)	After (Bytes)	Reduction
RGB	6291616	2156952	1 : 2.92

IV. CONCLUSIONS

This research proposed an RGB image encryption scheme based on Paillier Cryptographic scheme. During the encryption phase, the RGB image is separated into its R, G, and B-Channel images that are encrypted then combine into encrypted RGB images before being transmitted or stored. Before decryption, the encrypted RGB image is separated channel-wise and then decrypted and combined to produce the recovered RGB image. A series of security analyses and and tests such as correlation analysis, histogram analysis, cipher cycle, information entropy, recovered image quality analysis, and image compression analysis, are applied to our cipher-images and recovered image. Results from these tests show that our proposed image encryption scheme produce cipher-images that can resist these security attacks, as well as provide high quality recovered images. Future work could include applying our approach to other image types such as Grey-scale and others, as well as 16-bit images or other image data class. Our main contribution is the development of an image encryption algorithm described in our proposed Paillier based image encryption block diagram provided in chapter II.

REFERENCES

[1] A. Daneshgar and B. Khadem, A self-synchronized chaotic image encryption scheme, Signal Processing: Image Communication 36 (2015) 106-114.
www.elsevier.com/local/image

[2] A. K. A. Hassan, Reliable Implementation of Paillier Cryptosystem, Iraqi Journal of Applied Physics, IJAP, Vol. 10, No. 4, October-December 2014, pp. 27-29

[3] A. Soleymani, Md. J. Nordin, and Z. Md. Ali, A Novel Public Key Image Encryption Based on Elliptic Curves over Prime Group Field, Journal of Image and Graphics, Vol. 1, No. 1, March, 2013.

[4] A. Kanso, M. Ghebleh, A novel image encryption algorithm based on a 3D chaotic map, Commun Nonlinear Sci Numer Simulat 17 (2012) 29432959,
www.elsevier.com/locate/cnsns

[5] G. Ye and X. Huang, An Image Encryption Algorithm Based on Autoblocking and Electrocardiography, Published by the IEEE Computer Society. April-June 2016.

[6] G. Zhang, and Q. Liu, A novel image encryption method based on total shuffling scheme, Optics Communications 284 (2011) 27752780,
www.elsevier.com/locate/optcom

[7] G. Chen, Y. Mao, and C. K. Chui, A symmetric image encryption scheme based on 3D chaotic cat maps, Chaos, Solitons and Fractals 21 (2004) 749761,
www.elsevier.com/locate/chaos

[8] H. S. Kwok, Wallace K. S. Tang, A fast image encryption system based on chaotic maps with finite precision representation, Chaos, Solitons and Fractals 32 (2007) 15181529,
www.elsevier.com/locate/chaos

[9] H. Liu, X. Wang, and A. kadir, Image encryption using DNA complementary rule and chaotic maps, Applied Soft Computing 12 (2012) 14571466,
www.elsevier.com

[10] L. D. Singh and K. M. Singh, Image Encryption using Elliptic Curve Cryptography, Procedia Science 54 (2015) 475-481,
www.sciencedirect.com

[11] M. I. Wade, Distributed Image Encryption Based On a Homomorphic Cryptographic Approach, Ph.D. Dissertation, Howard University, May 2017

[12] N. K. Pareek, V. Patidar, and K. K. Sud, Image Encryption Using Chaotic Logistic Map, Image and Vision Computing 24 (2006) 926-934,
www.elsevier.com/locate/optlaseng

[13] P. P. Dang and P. M. Chau, Image Encryption for Secure Internet Multimedia Applications, IEEE Trans. On Consumer Electronics, Vol. 46. No. 3, August 2000.

[14] R. Tao, X. Meng, and Y. Wang, Image Encryption With Multiorders of Fractional Fourier Transforms, IEEE Trans. Inf. Forensics and Security, Vol. 5, No 4, Dec 2010.

[15] R. Rivest, Lecture Notes 15, Computer and Network Security: Voting, Homomorphic Encryption, October, 2002

[16] R. C. Gonzalez and R. E. Woods, Digital Image Processing.,3rd ed. Person Education Inc., 2008.

[17] R. Rhouma, S. Meherzi, and S. Belghith, OCML-based colour image encryption, Chaos, Solitons and Fractals 40 (2009) 309318,
www.elsevier.com/locate/chaos

[18] S. Mazloom and A. M. E-Moghadam, Color image encryption based on Coupled Nonlinear Chaotic Map, Chaos, Solitons and Fractals 42 (2009) 17451754,
www.elsevier.com/locate/chaos

[19] University of Southern California, Signal and Image Processing Institute,
http://sipi.usc.edu/database/

[20] Y. Zhou, L. Bao, C. L. P. Chen A new 1D chaotic system for image encryption, Signal Processing 97 (2014) 172182,
www.elsevier.com/locate/sigpro

[21] Yi. Xun, P. Russell, and B. Elisa, Homomorphic Encryption and Applications, 2014 XII, 126 p. 23 illus.,
http://www.springer.com/978-3-319-12228-1