Journal of Computer Science

Original Research Paper

Extensive Analysis on Images Encryption using Hybrid Elliptic Curve Cryptosystem and Hill Cipher

Saniah Sulaiman and Zurina Mohd Hanapi

Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti of Putra Malaysia, Serdang, Selangor, Malaysia

Article history
Received: 14-09-2020
Revised: 27-11-2020
Accepted: 11-03-2021

Corresponding Author:
Zurina Mohd Hanapi
Department of Communication Technology and Network,
Faculty of Computer Science and Information Technology,
Universiti of Putra Malaysia,
Serdang, Selangor, Malaysia
Email: zurinamh@upm.edu.my

Abstract: The advancement of communication technology helps individual to share images through an Internet. However, the sharing through insecure channels may expose the images to certain attacks that will compromise their confidentiality. Image encryption is one of the methods used to protect against the confidentiality threat. A Hill Cipher has been applied in image encryption because of its simple operation and fast computation, but it also possesses a weak security level which requires the sender and receiver to use and share the same private key within an unsecure channel. Thus, there are many solutions been proposed in utilizing hybrid approach of Hill Cipher where one of them is Elliptic Curve Cryptosystem together with Hill Cipher (ECCHC) to utilize the beauty of Hill Cipher while managing its weaknesses. However, the ECCHC only been experimented over four images which leads to inaccuracy of the results. Thus, this study extended the experiments on 209 images from USC-SIPI database in order to investigate the efficiency of ECCHC. The result shows the ECCHC produces poor performances on security analysis on grayscale and RGB images, which then concludes it is not suitable to encrypt grayscale and RGB images.

Keywords: Hill Cipher, Elliptic Curve Cryptography, Entropy, Peak Signal to Noise Ratio, Unified Average Changing Intensity

Introduction

Most of current communication technology use images to convey information, which include email, social media, text message and so on. However, sharing through unsecure channels may expose the images and other data to untrusted parties or attackers. Thus, the data i.e., images should be protected from adversary to avoid further cyber-attacks (Thein *et al*., 2017). The images should only be known to the authorized sender and receiver. To address the issue, cryptography technique is the most common way to secure the images transmissions occurring over the Internet (Patel and Belani, 2011).

There are two common types of cryptography, which are symmetric and asymmetric encryption. Symmetric concept uses the same private key for encryption and decryption process. For asymmetric, the sender will use public key to enable the encryption process, while the receiver will use his/her private key to enable the decryption process. Elliptic Curve Cryptosystem (ECC) which was proposed separately by (Miller, 1985; Koblitz, 1987) is an example of effective public key cryptography technique (Dawahdeh *et al*., 2018). ECC possesses the advantages of small key, rapid computation and high level security (Zhang *et al*., 2012).

Related Works

A Hill Cipher algorithm is one of the symmetric techniques which been used by several researches to encrypt an image due to its simple structure and fast computations. The image encryption using an Advanced Hill Cipher algorithm has been introduced by (Acharya *et al*., 2007) to solve the problem of the inverse key matric that usually did not exist in Hill Cipher algorithm, hence, eliminating the computation need to be done by the receiver to compute the inverse key. Acharya *et al*. (2010) was then modified the original Hill Cipher with an involuntary key by using iterations and interlacing. However, the solution only focused on a single grayscale image, thus making the result unreliable due to limited analysis. At the same time, (Hamissa *et al*., 2011) also overcame the weakness of the original Hill Cipher by introducing a new architecture of encoder-decoder named Coincidence to secure JPEG images by modifying the dynamic key generation, which produced the enhanced secure key matrix that resisted against the so-called plaintext-ciphertext attack. Moreover, to improve the entropy of the cipher image, few encryption techniques which combined Hill Cipher algorithm with bit rotation and reversal technique have been proposed (Panduranga and Kumar, 2012; Panduranga *et al*., 2012).

Futhermore, (Mahmoud and Chefranov, 2014) then modified the Hill Cipher to make Hill Cipher algorithm secure over the brute force, statistical attack and plaintext-ciphertext attack. This algorithm is known as HCM-PRE scheme. Sun and Guo (2015) later employed the Hill Cipher in their proposed image encryption algorithm by combining it with steganography technique based on contourlet. At the same time, hybrid of Chaos and Hill Cipher Based Image Encryption which included permutation and diffusion processes was proposed by (Naveenkumar *et al*., 2015) is another alternative to conventional Hill Cipher algorithm. Moreover, (Sazaki and Putra, 2016) combined the method of affine transform with the Advanced Hill Cipher. Goutham *et al*. (2017) is then modified the technique used by (Acharya *et al*., 2007) with a slight modification in terms of the key used which was 128 bits. However, this technique is vulnerable to statistical model-based attack, as the attackers will study the predictability of particular elements or the predictable relationship of some data segments between plain and cipher image due to result of correlation between plain image and cipher image, which is high, compared to other methods.

Even Hill Cipher algorithm is one of the popular symmetric techniques with a simple structure, high throughput and high speed, however it possesses weak security because it requires the sender and receiver to use and share the same private key via unsecured channels (Hill, 1929; Acharya *et al*., 2009; Dawahdeh *et al*., 2018). Due to this matter, (Dawahdeh *et al*., 2018) introduced a new image encryption technique combining Elliptic Curve Cryptosystem with Hill Cipher (ECCHC) to overcome the weakness mentioned. Thus, this combination technique makes the ECCHC technique as an asymmetric technique. This paper then extends the analysis by using images from USC SIPI database and MATLAB software for ECCHC encryption as well as decryption process to examine the reliability of ECCHC. The USC-SIPI image database is a collection of digitized images maintained primarily to support research in image processing, image analysis and machine vision (USC-SIPI, 2020).

Elliptic Curve Cryptosystem with Hill Cipher

There are three algorithms used; particularly Key Generation Algorithm, Encryption Algorithm and Decryption Algorithm. The initial stage of this technique requires the usage of the same Elliptic Curve Function *E* agreed by both sender and receiver and the domain parameters {*a*, *b*, *p* and *G*} are shared by the sender where *a* and *b* is the coefficient of the elliptic curve function, *p* is the large prime number and *G* is the generator point. In the case of key generation, user *A* represents the sender and user *B* represents the receiver. The private key from interval [1, *p*-1] is chosen by each user; n_A represents private key for user *A*, meanwhile n_B is the private key for user (Dawahdeh *et al*., 2018) defined the public key for each user as in Eqs. (1-2):

$$P_A = n_A \cdot GP_A = n_A \cdot G \tag{1}$$

$$P_B = n_B \cdot GP_B = n_B \cdot G \tag{2}$$

The private key of each user is multiplied with the public key of other user to produce the initial key K_1K_1 as in (3) before further computed in Eqs. (4-5):

$$K_1 = n_A \cdot P_B = n_B \cdot P_A = n_A \cdot n_B \cdot G = (x, y) \tag{3}$$

$$K_1 = x \cdot G = (k_{11}, k_{12}) K_1 = x \cdot G = (k_{11}, k_{12}) \tag{4}$$

$$K_2 = y \cdot G = (k_{12}, k_{12}) K_2 = y \cdot G = (k_{21}, k_{22}) \tag{5}$$

In this stage, self-invertible 4×4 key matrix is generated to enable the encryption and decryption of the image. Self-invertible matrix is a matrix equal to its inverse matrix or $K = K^{-1}K = K^{-1}$. To generate a self-invertible 4×4 key matrix K_m, a proposed method by (Acharya *et al*., 2007) is used. K_m is a self-invertible that can be viewed as Eq. (6):

$$K_m = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} \tag{6}$$

A self-invertible portioned that can be viewed as Eq. (7):

$$K_m = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \tag{7}$$

where:

$$K_m = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$$

Meanwhile, the rest of the elements are:

$$K_{12} = I - K_{11}K_{12} = I - K_{11},$$
$$K_{21} = I + K_{11}K_{21} = I - K_{11}$$

and:

$$K_{11} + K_{22} = 0K_{11} + K_{22} = 0$$

where, *I* is the identity matrix. The details algorithm of the key generation is determined as in Step 1-3 in (Dawahdeh *et al*., 2018).

The next stage requires the use of Hill Cipher Algorithm. The linear algebra equation for Hill Cipher can viewed as Eq. (8) (Agrawal and Gera, 2014).

$$C = KP \bmod 26 \tag{8}$$

Where:
C = The ciphertext
K = The key matrix
P = The plaintext

In this technique, each letter in the plaintext is assigned to numerical value. Then, the plaintext is divided into blocks consist of the same size m as the key matrix size, *m* × *m*. For example, the block size of four ($P_{4\times1}$) required the key matrix ($P_{4\times4}$) of size 4×4 (Dawahdeh *et al*., 2018). Based on Eq. (8), the ciphertext block with size 4×1 is generated as follows (Agrawal and Gera, 2014):

$$P = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} and\ K = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} then$$

$$C = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \bmod 26$$

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} (k_{11}p_1 + k_{12}p_2 + k_{13}p_3 + k_{14}p_4) \bmod 26 \\ (k_{21}p_1 + k_{22}p_2 + k_{23}p_3 + k_{24}p_4) \bmod 26 \\ (k_{31}p_1 + k_{32}p_2 + k_{33}p_3 + k_{34}p_4) \bmod 26 \\ (k_{41}p_1 + k_{42}p_2 + k_{43}p_3 + k_{44}p_4) \bmod 26 \end{bmatrix}$$

As proposed by (Dawahdeh *et al*., 2018), the image pixel values are separated into blocks of size four and each block are converted to a vector of size 4×1 (P_1, P_2, P_3,…). Ciphertext image *C* (C_1, C_2, C_3,…) is generated by multiply self-invertible key matrix, K_m by each vector and taking the modulo 256. Then, the ciphered image *C* is reconstructed from the values in the ciphered vectors and send it to the receiver. To generate the complete ciphered image, the following calculation are repeated for each block (Dawahdeh *et al*., 2018):

$$\text{Let } P_1 = \begin{bmatrix} p_{11} \\ p_{21} \\ p_{31} \\ p_{41} \end{bmatrix} \text{ then}$$

$$C_1 = K_m \cdot P_1 = \begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \\ c_{41} \end{bmatrix}$$

$$\begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \\ c_{41} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{21} \\ p_{31} \\ p_{41} \end{bmatrix} \bmod 26$$

$$\begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \\ c_{41} \end{bmatrix} = \begin{bmatrix} (k_{11}p_{11} + k_{12}p_{21} + k_{13}p_{31} + k_{14}p_{41}) \bmod 26 \\ (k_{21}p_{11} + k_{22}p_{21} + k_{23}p_{31} + k_{24}p_{41}) \bmod 26 \\ (k_{31}p_{11} + k_{32}p_{21} + k_{33}p_{31} + k_{34}p_{41}) \bmod 26 \\ (k_{41}p_{11} + k_{42}p_{21} + k_{43}p_{31} + k_{44}p_{41}) \bmod 26 \end{bmatrix}$$

The decryption process starts when the recipient receives the ciphered image. In the decryption process, the ciphered image undergoes the same techniques as the encryption process in order to generate the decrypted image (Dawahdeh *et al*., 2018).

Implementation

To illustrate the example of ECCHC implementation, we used the same example as described by (Dawahdeh *et al*., 2018) but with different image. Assume that User *A* wants to send an image *M* to user *B* by using the elliptic curve function where they both agreed on, which is:

$$E : y^2 \equiv x^2 + x + 3 (\bmod 31)$$

where:

$$A = 1, B = 3, p = 31$$

and these values satisfied the condition:

$$4A^3 + 27B^2 = 4(1)^3 + 27(3)^2$$
$$= 4 + 243$$
$$= 243 \bmod 31 = 30 \neq 0$$

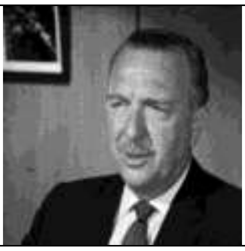
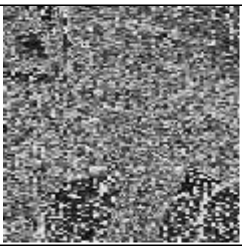
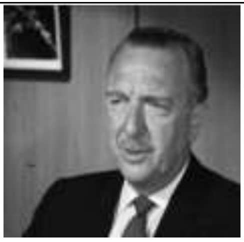
The original image	The encrypted image	The decrypted image
		

Fig 1: The original, encrypted, decrypted image of Walter Cronkite (Frame 1) Image

Table 1: Point of the function E: $y^2 \equiv x^3 + x + 3$ (*mod* 31)

(1,6)	(6,15)	(15,13)	(21,4)	(26,11)
(1,25)	(6,16)	(15,18)	(21,27)	(26,20)
(3,8)	(9,11)	(17,2)	(22,3)	(27,11)
(3,23)	(9,20)	(17,29)	(22,28)	(27,20)
(4,3)	(12,10)	(18,5)	(23,14)	(28,2)
(4,28)	(12,21)	(18,26)	(23,17)	(28,29)
(5,3)	(14,8)	(20,5)	(24,5)	(30,1)
(5,28)	(14,23)	(20,26)	(24,26)	(30,30)

(Source: Dawahdeh *et al*., 2018)

The points of elliptic curve $E_{31}(1,3)$ are shown in Table 1. The generator point $G(1,6)$ is chosen. This is because, the order of the elliptic curve $E_{31}(1,3)$ is 41, which is a prime number, thus any point from Table 1 can be chosen as a generator point. So that, the domain parameters for E are $\{A, B, p, G\} = \{1, 3, 31, (1, 6)\}$ (Dawahdeh *et al*., 2018).

If user A wants to send any grayscale or RGB colour image with any size of 256×256, 512×512 or 1024×1024 to User B, both users should apply ECCHC technique on the image described in the next steps. To illustrate the implementation of ECCHC technique, Walter Cronkite (Frame 1) image from USC SIPI Database is used. This image is a grayscale image with size of 256×256 pixels. Fig. 1 shows Walter Cronkite (Frame 1) images that have been encrypted and decrypted by the ECCHC.

Step 1: Key Generation

1.1 User A (The sender)

1.1.1 Choose the private key $n_A = 13 \in [1, 30]$

1.1.2 By using Eq. (1), compute the public key $P_A = n_A{\cdot}G = 13(1,6) = (3,23)$

1.1.3 By using Eq. (3), compute the initial key $K_1 = n_A{\cdot}P_B = 13(24,5) = (20, 5) = (x, y)$

1.1.4 By using Eq. (4) and (5), compute
$K_1 = x{\cdot}G = 20(1,6) = (4, 28) = (k_{11}, k_{12})$
and
$K_2 = y{\cdot}G = 5(1,6) = (15, 18) = (k_{21}, k_{22})$

1.1.5 Thus, based on Eq. (7) $K_{11} = \begin{bmatrix} 4 & 28 \\ 15 & 18 \end{bmatrix}$, then as per Eq. (6), the self-invertible key matrix

$$K_m = \begin{bmatrix} 4 & 28 & 253 & 228 \\ 15 & 18 & 241 & 239 \\ 5 & 28 & 252 & 228 \\ 15 & 19 & 241 & 238 \end{bmatrix}$$

1.2 User B (The receiver)

1.2.1 Choose the private key $n_B = 17 \in [1, 30]$

1.2.2 By using Eq. (2), compute the public key $P_B = n_B{\cdot}G = 17(1,6) = (24.5)$

1.2.3 By using Eq. (3), compute the initial key $K_1 = x{\cdot}G = 20(1,6) = (4, 28) = (k_{11}, k_{12})$

1.2.4 By using Eq. (4) and (5), compute
$K_1 = x{\cdot}G = 20(1,6) = (4, 28) = (k_{11}, k_{12})$
and
$K_2 = y{\cdot}G = 5(1,6) = (15, 18) = (k_{21}, k_{22})$

1.2.5 Thus, by implementing Eq. (7), $K_{11} = \begin{bmatrix} 4 & 28 \\ 15 & 18 \end{bmatrix}$, then based on Eq. (6) the self-invertible key matrix

$$K_m = \begin{bmatrix} 4 & 28 & 253 & 228 \\ 15 & 18 & 241 & 239 \\ 5 & 28 & 252 & 228 \\ 15 & 19 & 241 & 238 \end{bmatrix}$$

Step 2: Encryption (User A)

2.1 2.1. Separate Walter Cronkite (Frame 1) image pixel values into blocks of size four

	1	2	3	4	5	6	7	8	9
1	15	13	12	12	12	12	14	16	…
2	20	18	16	15	14	14	15	18	…
3	23	21	18	17	16	15	15	18	…
4	23	21	19	17	16	15	15	16	…
5	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

2.2 Based on highlighted blocks above:

$$P_1 = \begin{bmatrix} 15 \\ 13 \\ 12 \\ 12 \end{bmatrix}, P_2 = \begin{bmatrix} 12 \\ 12 \\ 14 \\ 16 \end{bmatrix}$$

2.3 Based on Equation (8), the multiplication of K_m with the first vector P_1 is computed and modulo 256 is computed later. The same process will be repeated for the rest of the vectors:

$$C_1 = K_m \cdot P_1 = \begin{bmatrix} 4 & 28 & 253 & 228 \\ 15 & 18 & 241 & 239 \\ 5 & 28 & 252 & 228 \\ 15 & 19 & 241 & 238 \end{bmatrix} \begin{bmatrix} 15 \\ 13 \\ 12 \\ 12 \end{bmatrix} \bmod 256 = \begin{bmatrix} 52 \\ 75 \\ 55 \\ 76 \end{bmatrix}$$

2.4 The pixel values for the encrypted image are

	1	2	3	4	5	6	7	8	9
1	52	75	55	76	150	170	148	166	…
2	116	129	120	132	155	187	154	183	…
3	150	164	155	168	191	235	192	232	…
4	147	149	151	153	48	48	49	49	…
5	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Step 3: Decryption (User B)

3.1 Separate the cipher image pixel values into blocks of size four

	1	2	3	4	5	6	7	8	9
1	52	75	55	76	150	170	148	166	…
2	116	129	120	132	155	187	154	183	…
3	150	164	155	168	191	235	192	232	…
4	147	149	151	153	48	48	49	49	…
5	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

3.2 Based on highlighted blocks above:

$$C_1 = \begin{bmatrix} 52 \\ 75 \\ 55 \\ 76 \end{bmatrix}, C_2 = \begin{bmatrix} 150 \\ 170 \\ 148 \\ 166 \end{bmatrix}$$

3.3 By using Eq. (8), the multiplication of key matrix K_m with the first vector C_1 is computed and modulo 256 is computed later. The same process will be repeated for the rest of the vectors:

$$P_1 = K_m \cdot C_1 = \begin{bmatrix} 4 & 28 & 253 & 228 \\ 15 & 18 & 241 & 239 \\ 5 & 28 & 252 & 228 \\ 15 & 19 & 241 & 238 \end{bmatrix} \begin{bmatrix} 52 \\ 75 \\ 55 \\ 76 \end{bmatrix} \bmod 256 = \begin{bmatrix} 15 \\ 13 \\ 12 \\ 12 \end{bmatrix}$$

3.4 The original image P is constructed from the values in the deciphered vectors (P_1, P_2, P_3,….)

	1	2	3	4	5	6	7	8	9
1	15	13	12	12	12	12	14	16	…
2	20	18	16	15	14	14	15	18	…
3	23	21	18	17	16	15	15	18	…
4	23	21	19	17	16	15	15	16	…
5	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Security Analysis

In this section, the security analyses for the proposed technique are been elaborated; entropy, PSNR and UACI of the encrypted images.

Entropy

The first security parameter is entropy. Generally, the information entropy is an evaluation of the uncertainty degree in the system, which also can be used to express the uncertainty in image encryption. Zhang *et al*. (2010). Theoretically, eight is the ideal entropy value for the grayscale image with size 256×256 (Dawahdeh *et al*., 2018). This concept is also applied on the RGB color images. If the entropy value of the cipher image is nearly equal to 8, the information leakage during encryption process is negligible (Zhu *et al*., 2011). The formula of the entropy is shown Eq. (9) (Dawahdeh *et al*., 2018):

$$Entropy(E) = \sum_{x=0}^{255} \left[P(x) \times \log_2 \left(\frac{1}{P(x)} \right) \right] \qquad (9)$$

where, $P(x)$ is the probability of the pixel value x and computed by Eq. (10):

$$P(x) = \frac{The\ frequency\ of\ the\ pixel\ value\ x}{Total\ number\ of\ the\ image\ pixels} \qquad (10)$$

Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio, often-abbreviated PSNR, is the ratio between the plain image and the cipher image. The higher the PSNR value, the closer the cipher image to its plain image (Sethi and Vijay, 2013). Higher values of PSNR mean that the loss data in cipher image is negligible, which means the cipher image is almost identical to plain image (Rajput and Gulve, 2014). For good image encryption technique, PSNR value should be the most minimum. Lower PSNR values of cipher image would make it difficult to recover the plain image from its corresponding cipher image, without the knowledge of correct key of decryption. The Eq. (11) is used to compute PSNR (Taneja *et al*., 2012).

$$PSNR = 20 \times \log_{10} \left[\frac{255}{MSE} \right] \qquad (11)$$

where, *MSE* is Mean Square Error between the plain image and the cipher image and computed by Eq. (12):

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(A_{ij} - B_{ij} \right)^2 \qquad (12)$$

Where:
A_{ij} = The pixel value of the plain image
B_{ij} = The pixel value of the encrypted image

H = The total number of pixels in horizontal position
W = The total number of pixels in vertical position

We can observed that, if MSE is increased, it will cause the PSNR decrease. The high value of MSE and low value of PSNR indicates that the plain and cipher image are different and not identical and this leads to an efficient encryption technique (Dawahdeh *et al*., 2018; Naskar and Chaudhuri, 2014).

Unified Average Changed Intensity (UACI)

UACI is the common value used to evaluate the performance of image encryption technique against the differential attack (Wu *et al*., 2011). Differential attack is a chosen plaintext attack, where the cryptanalyst has made a slight change on the chosen plain image to produce the cipher image. Then, a cryptanalysis will deduce the statistical relationship between cipher image and the corresponding chosen plain image by computing the differences between them. UACI reflects the difference of average intensity between cipher image and plain image (Taneja *et al*., 2012). To determine the robustness of the image encryption technique against differential attack and attain higher sensitivity of the alteration on the plain image, UACI should be nearly equal to 33.5 (Kabirirad and Hajiabadi, 2015). UACI can be calculated using the Eqs. (13-14) (Li and Lo, 2018):

$$D(i,j)=\begin{cases}0, if \;\; A(i,j)=B(i,j)\\ 1, if \;\; A(i,j)\neq B(i,j)\end{cases} \tag{13}$$

$$UACI: U(A,B)=\frac{\sum i,j\frac{|A(i,j)-B(i,j)|}{255}}{H\times W}\times 100\% \tag{14}$$

Where:
$A(i,j)$ = The pixel value of the plain image
$B(i,j)$ = The pixel value of the cipher image

Security Analysis based on Limit Value

In this analysis, 209 images from (USC-SIPI, 2020) database were used as a dataset. These images are grayscale and RGB color images with variety size of 256×256, 512×512 and 1024× pixels. The results of the experiments consist of a wide range of value for each security parameter as it involves multiple numbers of images. Thus, a limit value (benchmark) to determine a good result must be set. The limit value is chosen based on the results of (Dawahdeh *et al*., 2018) due to the fact that the author claimed the produced results are good. The following values were chosen:

i. 7.9848 was pinpointed as limit value for entropy
ii. 9.7483 was pinpointed as limit for PSNR
iii. 26.9087 was pinpointed as limit value for UACI

These three limit values were chosen among the others because they are the farthest from their ideal value, besides they are also significant as the original author claimed them as good results.

Table 2 shows that, 109 grayscale images show good results of entropy because the entropy of this encrypted images are greater equal to 7.9848. On the other hand, 50 grayscale images acquire bad results of entropy as they produced entropy values lower than 7.9848.

This shows that, 69% of the total number of grayscale images produced good results of entropy, meanwhile 31% of the total number of grayscale images presented bad results of entropy. As this study focuses on the confidentiality of the encrypted images, thus, the acceptable percentage for confidentiality is 100% because we cannot afford any compromise on the confidentiality. Based on security risk assessment survey done by (Casas, 2006), 100% of the respondents agreed that confidentiality is important despite any situation. Due to that, ECCHC technique should provide 100% of confidentiality and this percentage will be a benchmark to analyze further result regarding percentages. Thus, the analysis concludes that the ECCHC technique produces inefficient result of entropy for grayscale image since 69% is lower than 100%.

Table 3 shows that 126 grayscale images have good results for PSNR because the values for these encrypted images are lower equal to 9.7483, on the other hand, 33 images obtained bad results of PSNR because their PSNR values are higher than 9.7483. This shows that the percentage for good results of PSNR is 79% and the percentage for bad results of PSNR is 21%. Based on this analysis, it is appropriate to conclude that ECCHC technique is unable to produce good result of PSNR for grayscale image because 79% is considered as low percentage because it is lower than 100%.

Table 2: The number of encrypted grayscale images based on limit value of entropy

Entropy	No. of images
>= 7.9848	109
<7.9848	50

Table 3: The number of encrypted grayscale images based on limit value of PSNR

PSNR	No. of images
<= 9.7483	126
>9.7483	33

Table 4: The number of encrypted grayscale images based on limit value of UACI

UACI	No. of images
>= 26.9087	124
<26.9087	35

By referring to Table 4, 124 grayscale images guaranteed good results of UACI due to values to be greater equal to 26.9087. However, 35 grayscale images produced bad results of UACI as UACI of these encrypted images are lower than 26.9087. This shows that 78% of the total number of grayscale images have good results of UACI, while 22% from total number of grayscale images represent the bad results of UACI. Therefore, ECCHC technique is also unable to produce good result of UACI since 78% is lower than 100%.

Table 5 exhibits that 31 RGB colour images show good results of entropy as they are greater or equal to 7.9848, but 19 RGB colour images obtained bad results of entropy as these images produced entropy lower than 7.9848. This shows that 62% of total number of RGB colour images have good results of entropy and 38% of total number of RGB colour images have bad results of entropy. Thus, the ECCHC technique is again unable to produce good result of entropy for RGB colour image because 62% is a low percentage.

Table 6 shows that 43 RGB colour images obtained good results of PSNR; which is lower than 9.7483. On the contrary, other 7 images produced bad results of PSNR analysis because their PSNR values are higher than 9.7483. This shows that 86% of the total number of RGB colour images represent the good results of PSNR, while the rest of the percentages, 14%, represent the bad results of PSNR. Even though 86% is quite high, yet, in this study, we must attain 100% to ensure that there is no encrypted image identical to its original image. Thus, ECCHC technique is unable to provide good result of PSNR for RGB colour image.

By referring to Table 7, 40 RGB colour image confirmed good results of UACI; where values are greater equal to 26.9087. However, 10 RGB colour images produced bad results of UACI; as they are lower than 26.9087. This shows that the percentage of RGB colour image that produced good results of UACI is 80%, while 20% of the total number of RGB images produced bad results of UACI. Hence, ECCHC technique is also unable to provide good result of UACI for RGB colour image.

From Tables 2-7, it can be observed that several images produced not good results for the three security parameters. Hence, we concluded that ECCHC technique does not provide good results of entropy, PSNR and UACI for both grayscale and RGB colour image. However, the analysis had been extended by categorizing the encrypted images as work well, moderate and not work well to determine the security level for each of the images. The requirements for these three categories are summarized in Table 8.

As shown in Table 8, to ensure the encrypted image is labelled as work well, the entropy of the encrypted image should be greater equal to 7.9848 because work well encrypted images should have entropy that closer to 8. Besides, the encrypted images should have PSNR lower equal to 9.7483 because the lower PSNR value of the cipher image makes it difficult to recover the original image from its corresponding cipher image and the last requirement to be defined as work well category is that UACI should be greater or equal to 26.9087. This is because, UACI should be closer to 33.5 to ensure that the technique is secure against differential attack.

Meanwhile, for a not work well category, all the requirements must contradict to the requirements of work well category. Thus, the encrypted images labelled as not work well should have the entropy lower than 7.9848, PSNR must be higher than 9.7483 and UACI lower than 26.9087. Lastly, for moderate category, the encrypted images should not fulfil at least one requirement of work well category.

The pie charts shown in Figs. 2 and 3 represent the number of encrypted images labelled as work well, not work well and moderate for grayscale and RBG colour images, respectively.

Table 5: The number of encrypted RGB colour images based on limit value of entropy

Entropy	No. of images
>= 7.9848	31
<7.9848	19

Table 6: The number of encrypted RGB colour images based on limit value of PSNR

PSNR	No. of images
<= 9.7483	43
>9.7483	7

Table 7: The number of encrypted RGB colour images based on limit value of UACI

UACI	No. of images
>= 26.9087	40
<26.9087	10

Table 8: Requirements for three categories of security level

Security level	Entropy	PSNR	UACI
Work well	>= 7.9848	<= 9.7483	>= 26.9087
Not work well	<=7.9848	>9.7483	<26.9087
Moderate	Did not fulfill at least 1 requirement of work well		

Based on Fig. 3, there are 30 encrypted RGB colour images labelled as work well, 14 encrypted RGB colour images labelled as moderate and 6 encrypted RGB colour images labelled as not work well. Thus, 60% of the total number of RGB colour images are encrypted well by ECCHC, 28% of the total number of RGB colour images are moderately encrypted and 12% of the total number of RGB colour images are not encrypted well. Based on these percentages, there are several RGB colour images not encrypted well by ECCHC technique. Furthermore, 60% of work well percentage is considered low percentage and unable to support that ECCHC technique is a secure image encryption for RGB colour image.

Therefore, ECCHC technique does not guarantee providing an image encryption with 100% work well on all grayscale and RGB images and does not guarantee that these encrypted images are 100% resist against entropy attack, differential attack (due to entropy and UACI value that is far away from their identical value) and the cipher image might reveal the secrecy of the plain image as well due to high value of PSNR.

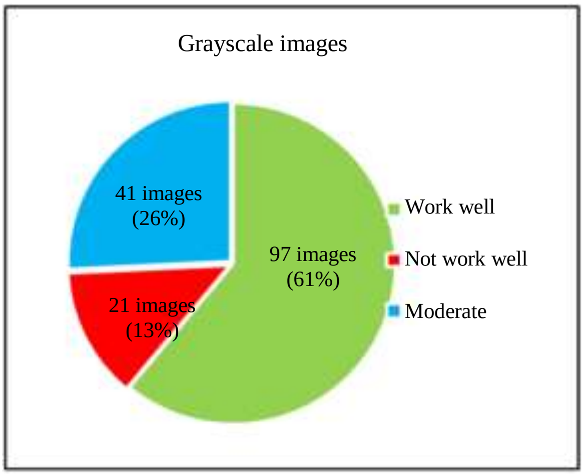


Fig. 2: Pie chart of three categories of encrypted grayscale images

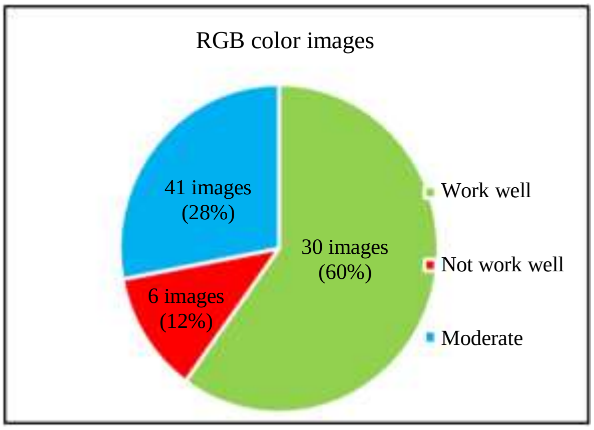


Fig. 3: Pie chart of three categories of encrypted RGB colour images

Analysis on the Actual Data (Encrypted Images)

In this experiment, analysis is done on actual data by observing the encrypted images labelled as work well and moderate. This is to ensure the secrecy of the plain image cannot be seen through encrypted image. Through an observation, for work well encrypted image, there is no secrecy being exposed. However, for moderate encrypted images, there are some images that exposed as part of the secrecy of the plain images. The results of moderate encrypted images are represented by pie charts in Fig. 4 and 5 for grayscale and RGB colour images, respectively.

Based on Fig. 4, 56% of the moderate encrypted grayscale images (23 images) promise the secrecy of the plain image. However, 44% of the moderate encrypted grayscale images (18 images) are not actually encrypted well because these images exposed a part of secrecy of the plain image. Almost half of moderate encrypted grayscale images are not encrypted well and these images should be labelled as not work well.

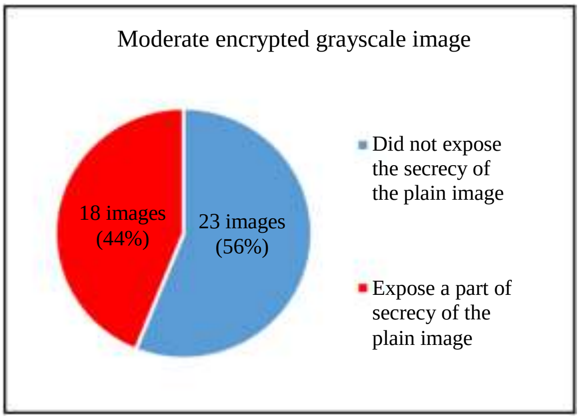


Fig. 4: Pie chart of the number of exposed moderate encrypted grayscale images

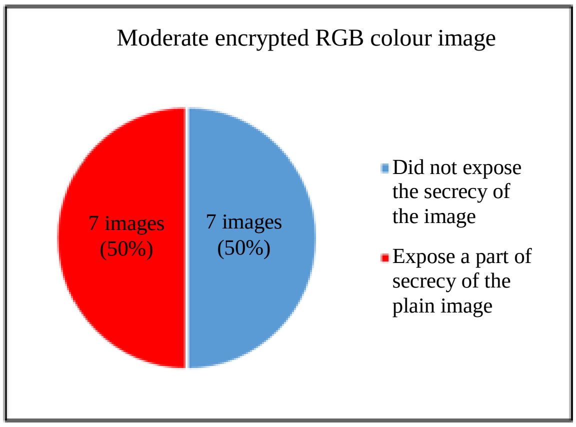


Fig. 5: Pie chart of the number of exposed moderate encrypted RGB colour images

As shown in Fig. 5, 50% of the moderate encrypted RGB colour images (7 images) successfully concealed the secrecy of the plain image and another 50% of the moderate encrypted RGB colour images (7 images) exposed a part of the secrecy of the plain image and are not encrypted well. This shows that half from the total number of moderate encrypted RGB colour images should be labelled as not work well. Based on these two results, we can conclude that the security analysis of ECCHC technique is not reliable and the analysis does not represent the confidentiality of the encrypted images for both grayscale and RGB colour image.

Conclusion

Based on the experiments and results obtained, ECCHC technique is unable to encrypt some of the grayscale and RGB colour images given that some of these encrypted images produced bad results of security analysis. The results from experiments already proved that 21 grayscale and 6 RGB colour images are labelled as not work well; hence, these images are probably vulnerable to entropy and differential attack and the encrypted image might be almost identical to plain image. Therefore, making an attack becomes an easier task for attackers. Furthermore, by extending the analysis on the actual data by doing the observation on the encrypted images, there are several images initially labelled as moderate, but they are not been encrypted well. In consequence, making the security analysis of ECCHC technique does not represent the confidentiality of the encrypted images. Even though ECCHC technique involves the ECC algorithm, ECC only work in key generation however, the encryption process is fully comes from the Hill Cipher algorithm. Therefore, Hill Cipher is not suitable to be deployed to encrypt the images.

Acknowledgement

We would like to acknowledge the author of ECCHC who willing to share the work to be further evaluated.

Funding Information

This work is supported by Geran Putra Berimpak Universiti Putra Malaysia (9659400).

Author’s Contributions

Saniah Sulaiman: Did the extensive experiment and preparing the draft manuscript

Zurina Mohd Hanapi: Is the supervisor of Saniah Sulaiman who supervised the work and prepared a final manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Acharya, B., Panigraphy, S. K., Patra, S.K., & Panda, G (2009). Image encryption using advanced hill cipher algorithm. International Journal of Recent Trends in Engineering, 1(1), 663-667. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.592.6197&rep=rep1&type=pdf

Acharya, B., Rath, G. S., Patra, S. K., & Panigrahy, S. K. (2007). Novel methods of generating self-invertible matrix for hill cipher algorithm. International Journal of Security, 1(1), 14-21. http://dspace.nitrkl.ac.in/dspace/handle/2080/620

Acharya, B., Sharma, M. D., Tiwari, S., & Minz, V. K. (2010). Privacy protection of biometric traits using modified hill cipher with involutory key and robust cryptosystem. Procedia Computer Science, 2, 242-247. https://doi.org/10.1016/j.procs.2010.11.031

Agrawal, K., & Gera, A. (2014). Elliptic curve cryptography with hill cipher generation for secure text cryptosystem. International Journal of Computer Applications, 106(1), 18-24. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.736.364&rep=rep1&type=pdf

Casas, V. (2006). An information security risk assessment model for public and university administrators. https://digital.library.txstate.edu/handle/10877/3674

Dawahdeh, Z. E., Yaakob, S. N., & bin Othman, R. R. (2018). A new image encryption technique combining Elliptic Curve Cryptosystem with Hill Cipher. Journal of King Saud University-Computer and Information Sciences, 30(3), 349-355. https://doi.org/10.1016/j.jksuci.2017.06.004

Goutham, L., Mahendra, M. S., Manasa, A. P., & Prajwalasimha, S. N. (2017). Modified Hill Cipher Based Image Encryption Technique. International Journal for Research in Applied Science & Engineering Technology. 5(Iv), 342–345. https://doi.org/10.22214/ijraset.2017.4063

Hamissa, G., Sarhan, A., Abdelkader, H., & Fahmy, M. (2011, November). Securing JPEG architecture based on enhanced chaotic hill cipher algorithm. In The 2011 International Conference on Computer Engineering & Systems (pp. 260-266). IEEE. https://doi.org/10.1109/ICCES.2011.6141053

Hill, L. S., (1929). Cryptography in an Algebraic Alphabet. The American Mathematical Monthly, 36(6), 306-312. https://doi.org/10.1080/00029890.1929.11986963

Kabirirad, S., & Hajiabadi, H. (2015). Cryptanalysis of An Authenticated Image Encryption Scheme Based On Chaotic Maps And Memory Cellular Automata. IACR Cryptol. ePrint Arch., 2015, 326. https://eprint.iacr.org/2015/326.pdf

Koblitz, N. (1987). Elliptic curve cryptosystems. Mathematics of Computation, 48(177), 203-209. https://doi.org/10.1090/S0025-5718-1987-0866109-5

Li, P. & Lo, K. T. (2018). A Content-Adaptive Joint Image Compression and Encryption Scheme. IEEE Transactions on Multimedia, 20(8), 1960-1972. https://doi.org/10.1109/TMM.2017.2786860

Mahmoud, A., & Chefranov, A. (2014). Hill cipher modification based on pseudo-random eigenvalues. Applied Mathematics & Information Sciences, 8(2), 505-516. https://doi.org/10.12785/amis/080208

Miller, V. S. (1985, August). Use of elliptic curves in cryptography. In Conference on the theory and application of cryptographic techniques (pp. 417-426). Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-39799-X_31

Naskar, P. K., & Chaudhuri, A. (2014). A secure symmetric image encryption based on bit-wise operation. International Journal of Image, Graphics and Signal Processing, 6(2), 30-38. https://doi.org/10.5815/ijigsp.2014.02.04

Naveenkumar, S. K., & Panduranga, H. T. (2015, March). Chaos and hill cipher based image encryption for mammography images. In 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS) (pp. 1-5). IEEE. https://doi.org/10.1109/ICIIECS.2015.7193175

Panduranga, H. T. (2012). Advanced partial image encryption using two-stage hill cipher technique. International Journal of Computer Applications, 60(16),14-19. https://doi.org/10.5120/9775-4341

Panduranga, H. T., Kumar, H. S., & Kumar, S. N. (2012, December). Hybrid approach for dual image encryption using nibble exchange and Hill-cipher. In 2012 International Conference on Machine Vision and Image Processing (MVIP) (pp. 101-104). IEEE. https://doi.org/10.1109/MVIP.2012.6428770

Patel, K. D., & Belani, S. (2011). Image encryption using different techniques: A review. International Journal of Emerging Technology and Advanced Engineering, 1(1), 30-34. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.414.5213&rep=rep1&type=pdf

Rajput, Y., & Gulve, A. K. (2014). A comparative performance analysis of an image encryption technique using extended hill cipher. International Journal of Computer Applications, 95(4), 16-20. https://doi.org/10.5120/16582-6279

Sazaki, Y., & Putra, R. S. (2016, October). Implementation of affine transform method and advanced hill cipher for securing digital images. In 2016 10th International Conference on Telecommunication Systems Services and Applications (TSSA) (pp. 1-5). IEEE. https://doi.org/10.1109/TSSA.2016.7871068

Sethi, N., & Vijay, S. (2013, April). Comparative image encryption method analysis using new transformed-mapped technique. In Proceedings of the Conference on Advances in Communication and Control Systems-2013. Atlantis Press. https://www.atlantis-press.com/proceedings/cac2s-13/6276

Sun, S., & Guo, Y. (2015). A novel image steganography based on contourlet transform and hill cipher. Journal of Information Hiding and Multimedia Signal Processing, 6(5), 889-897. http://www.jihmsp.org/~jihmsp/2015/vol6/JIH-MSP-2015-05-006.pdf

Taneja, N., Raman, B., & Gupta, I. (2012). Combinational domain encryption for still visual data. Multimedia Tools and Applications, 59(3), 775-793. https://doi.org/10.1007/s11042-011-0775-4

Thein, N., Nugroho, H. A., Adji, T. B., & Mustika, I. W. (2017, November). Comparative performance study on ordinary and chaos image encryption schemes. In 2017 International Conference on Advanced Computing and Applications (ACOMP) (pp. 122-126). IEEE. https://doi.org/10.1109/ACOMP.2017.25

USC-SIPI. (2020). The USC-SIPI Image Database. http://sipi.usc.edu/database/

Wu, Y., Noonan, J. P., & Agaian, S. (2011). NPCR and UACI randomness tests for image encryption. Cyber journals: multidisciplinary journals in science and technology. Journal of Selected Areas in Telecommunications (JSAT), 1(2), 31-38. https://www.researchgate.net/profile/Yue-Wu-86/publication/259190481_NPCR_and_UACI_Randomness_Tests_for_Image_Encryption/links/0c96052a80913343e2000000/NPCR-and-UACI-Randomness-Tests-for-Image-Encryption.pdf

Zhang, Q., Guo, L., & Wei, X. (2010). Image encryption using DNA addition combining with chaotic maps. Mathematical and Computer Modelling, 52(11-12), 2028-2035. https://doi.org/10.1016/j.mcm.2010.06.005

Zhang, X., Zhu, G., Wang, W., & Wang, M. (2012, August). Design and realization of elliptic curve cryptosystem. In 2012 International Symposium on Instrumentation & Measurement, Sensor Network and Automation (IMSNA) (Vol. 1, pp. 302-305). IEEE. https://doi.org/10.1109/MSNA.2012.6324573

Zhu, Z. L., Zhang, W., Wong, K. W., & Yu, H. (2011). A chaos-based symmetric image encryption scheme using a bit-level permutation. Information Sciences, 181(6), 1171-1186. https://doi.org/10.1016/j.ins.2010.11.009