

A Multi-Stage Encryption Technique to Enhance the Secrecy of Image

Arindom Mondal¹, Kazi Md. Rokibul Alam¹, G. G. Md. Nawaz Ali^{1,2*}, Peter Han Joo Chong³ and Yasuhiko Morimoto⁴

¹Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Bangladesh
[e-mail: arindom.csekuet@gmail.com, rokib@cse.kuet.ac.bd]

²Department of Automotive Engineering, Clemson University, USA
[e-mail: gga@clemson.edu]

³Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand
[e-mail: peter.chong@aut.ac.nz]

⁴Graduate School of Information Engineering, Hiroshima University, Higashi-Hiroshima 739-8521, Japan
[e-mail: morimo@hiroshima-u.ac.jp]

*Corresponding author: G. G. Md. Nawaz Ali

*Received January 2, 2018; revised June 3, 2018; accepted July 22, 2018;
published May 31, 2019*

Abstract

This paper proposes a multi-stage encryption technique to enhance the level of secrecy of image to facilitate its secured transmission through the public network. A great number of researches have been done on image secrecy. The existing image encryption techniques like visual cryptography (VC), steganography, watermarking etc. while are applied individually, usually they cannot provide unbreakable secrecy. In this paper, through combining several separate techniques, a hybrid multi-stage encryption technique is proposed which provides nearly unbreakable image secrecy, while the encryption/decryption time remains almost the same of the exiting techniques. The technique consecutively exploits VC, steganography and one time pad (OTP). At first it encrypts the input image using VC, i.e., splits the pixels of the input image into multiple shares to make it unpredictable. Then after the pixel to binary conversion within each share, the exploitation of steganography detects the least significant bits (LSBs) from each chunk within each share. At last, OTP encryption technique is applied on LSBs along with randomly generated OTP secret key to generate the ultimate cipher image. Besides, prior to sending the OTP key to the receiver, first it is converted from binary to integer and then an asymmetric cryptosystem is applied to encrypt it and thereby the key is delivered securely. Finally, the outcome, the time requirement of encryption and decryption, the security and statistical analyses of the proposed technique are evaluated and compared with existing techniques.

Keywords: Image secrecy, Multi-stage encryption, Visual Cryptography, One time pad, Steganography, Least Significant Bit

1. Introduction

Nowadays, due to diversified and extensive users of computer networks and internet, the number of intrusions is increasing day-by-day. Therefore, while image, message etc. are transmitted over computer networks, secure communication is essential. Note that an image consists of a number of pixels which are highly correlated, whereas a message consists of characters and/or binary and/or integer and/or hexadecimal values [1]. Thus an image is distinct from a message and hence their encryption techniques are also somehow different. In addition very often, hugely it is required to share the image over the network. Thereby similar to message encryption, image encryption is another significant branch of cryptography. RSA, ElGamal, AES, DES etc. are examples of some well-known message encryption techniques. To encrypt an image, techniques like visual cryptography (VC) [1], steganography [2], watermarking [3], deep learning based ones [23, 24], chaotic system and symmetric/asymmetric cryptosystem based ones [16, 17] etc. have been developed. However while image is retrieved through decryption, many of them are not extremely efficient. The increased secrecy of image over public network is necessary, for instance, transmitting bank cheque image, hand-written signature, biometric authentication etc.

Intuitively, VC, watermarking, steganography etc. techniques are exclusively associated with image encryption and their advantage is: to conduct their encryption and decryption processes usually they do not need to rely on any specific key/keys. For example while encryption, VC splits an image into n shares and decrypts it by superimposing the shares [12]. The limitation of VC is: its original formation is restricted to binary images. Also, the alignment of two shares is not so easy to perform unless some special alignment marks are provided [5]. Steganography hides an image within another cover image to protect its contents [2]. Hence, the handling of the cover image along with the input image makes the encryption and decryption process bulky [10]. Watermarking technique combines cover image with a watermark, which is hard to be detected or removed. By the way, the owner of the image can prove its copyright by extracting the watermark from the watermarked image. However, note that the watermarking technique is not free from attacks. The attacks associated with this technique are: compression, blurring, noise, distortion, sharpening, scaling, cropping etc. [11]. Thus, these techniques are substantially weaker than cryptography based message encryption techniques. Thereby while they are applied separately or jointly, they possess various limitations.

In order to encrypt the image more securely, this paper proposes a new multi-stage encryption technique that unifies ‘the benefits of cryptosystems not relying on any specific key/keys’ along with ‘the strength of the cryptosystem relying on a specific key’. For this purpose, it combines VC, steganography along with one time pad (OTP). Note that OTP technique possesses ‘perfect secrecy’ and cannot be cracked [8]. At first employing VC, it splits pixels of the input image into multiple shares to make it unreadable. Now from each share, the pixels are transformed into their corresponding binary values. Then applying steganography, it detects least significant bits (LSBs) from each chunk of binary values of each share. Finally, it applies OTP (i.e., XOR-ing) encryption technique on LSBs using randomly generated OTP secret key (it is binary value) to generate the ultimate cipher image which is sent to the receiver. Besides, prior sending the random OTP secret key to the receiver, it is transformed into integer value to encrypt it using asymmetric cryptosystem like Paillier [13] to use it for the image decryption purposes. Thus the level of secrecy of the input image is

increased significantly. The proposed technique is tested under Histogram Analysis, Salt & Pepper Noise attack and Chosen-Plaintext Attack (CPA) to verify the strength of the proposed encryption/decryption under noisy channel environment or cipher image leak. The results show a satisfactory performance with the acceptable encryption/decryption time.

The rest of this paper is organized as follows. Section 2 discusses the related works. Section 3 explains the cryptographic tools required to developing the proposed technique. Section 4 describes the proposed technique. Section 5 illustrates the experimental analysis, Section 6 presents the security and statistical analyses and finally, Section 7 concludes the paper.

2. Related Works

To ensure the secrecy of image, researchers have exploited techniques like chaotic system and symmetric/asymmetric cryptosystem based ones, VC, steganography, watermarking, VC followed by steganography, steganography followed by VC, machine learning based ones etc.

The technique of VC introduced in [1] is for monochrome image. For encryption it divides an image consists of random white and black pixels into n shares and then for decryption, the superimposing of all shares are required. For each pixel of each share, two blocks are generated in the corresponding location. It is assumed that in each share any white pixel is transparent and any black pixel is opaque which are stored as binary 0 and 1 values, respectively [12]. The major drawbacks of VC based techniques are pixel-expansion, low resolution, alignment problems etc. [5].

Steganography embeds an image within another cover image in such a way that the intruder cannot identify the existence of the input image from the embedded image [2]. Among many variations of steganographic techniques already proposed, the technique proposed in [7] directly hides the image by replacing the LSBs of each pixel of the cover image. It embeds the same amount of bits of the input image that makes a minor change of the cover image. The main drawbacks of steganographic techniques are: the use of cover image makes the image encryption and decryption process bulky and increase the processing time and storage capacity.

The watermarking technique hides an image into another cover image. The technique proposed in [3] divides a cover image into two blocks and each block is transformed with a two-dimensional discrete cosine transform (DCT) to classify as smooth block or edge block. Then biometric features are embedded in the low frequency coefficients of the 8×8 DCT blocks while the edge blocks are eliminated. However, the elimination of the edge blocks degrades the quality of the input image. Also, attacks like random cropping or shuffling can destroy the coded watermark [11]. The watermarking technique proposed in [21] is a multiple staged one. To endow the watermarked document, it produces a watermark specification by creating a template specification that illustrates the way to combine the watermark into the targeted document.

Exploiting face image, another multi-stage face recognition technique proposed in [22] adopts a local structure which is based on a multi-phase collaborative representation technique. It studies the local structure connection-ship of overlapping patches. Besides currently due to extensive popularity of devices like digital cameras, intelligent mobile devices, techniques proposed in [25, 26] deals with social/community-contributed image retrieval/understanding focusing on the transformation of images and by analyzing the content. The technique proposed in [25] is based on the deep learning framework known as weakly-supervised deep metric learning. The technique proposed in [26] proposes a weakly supervised deep matrix

factorization algorithm that can deal with the incomplete, noisy or subjective tags while eliminating the redundant or noisy visual features.

To provide the secrecy of an image, the technique proposed in [4] combines steganography and VC, that splits the image into two shares, where each share is stored in different databases. Later on while retrieving the image, the superimposing of both shares is required. Here for storing the shares in two different databases, the storage cost is increased.

To ensure the image secrecy, the hybrid technique proposed in [6] is known as VC followed by steganography. It splits the input image into multiple shares using VC and converts pixels of each share into binary values. Similarly, pixels of the cover image that is used for transmitting the input image is also converted into binary values. Then LSBs of each chunk of each share are computed from that binary values of cover image and replaced one by one bit with the binary values of the input image. Here, steganography hides share images generated by VC into the LSBs of the pixel values in the cover image.

The hybrid technique proposed in [7] is known as steganography followed by VC where the input image is embedded inside a cover image using steganography and then the embedded image is divided into different shares using VC. At first both the input image and the cover image is converted into pixel and then to binary data. After computing the LSBs of each chunk of the cover image and replacing one by one bit with the binary value of input image, this binary image is split into multiple shares.

To maintain the privacy of the image while transmitting it over the internet, the technique proposed in [23] performs image compression as well as encryption sequentially. To do so, it employs deep learning based algorithms. At first, it uses Stacked Auto Encoder (SAE) to compress the image. Then to encrypt this compressed one, it uses chaotic logistic map. To encrypt a batch of images where each image is encrypted with an autonomous sequence, another deep learning based technique is proposed in [24] that also employs a SAE to generate two chaotic matrices. The first matrix is used to generate a total shuffling matrix that shuffles the pixel positions on the input image. Then the second matrix is used to generate the series of autonomous sequences that establishes confusion between the shuffled image and the cipher image. However the shortcoming of this technique is, it cannot handle input images with various sizes.

Different from the above techniques, to encrypt the image more securely, the hybrid technique proposed in this paper consecutively combines VC, steganography and OTP altogether. Here at first VC splits pixels of the input image into 2^n shares, where $n \geq 1$. Note that while the value of n increases, the level of secrecy also increases. Now pixels of each share are transformed into their corresponding binary values. Then instead of using a cover image, it only detects LSBs from each chunk of binary values of each share exploiting steganography. Lastly, it applies OTP encryption technique on LSBs using random OTP secret key to generate the final cipher image. Besides before sending the OTP key to the receiver, it is transformed into integer value to encrypt it using Paillier cryptosystem for the decryption purpose. Thus the successive exploitation of several techniques increases the level of secrecy of the input image significantly while maintaining its quality.

3. Cryptographic Building Blocks















The proposed technique exploits several cryptographic tools. These are: VC, steganography, OTP and Paillier cryptosystem. This section describes them.

3.1 Visual Cryptography (VC)

The basic model of VC is introduced in [1]. Here for encryption, an input image is divided into 2^n shares, $n \geq 1$. Later on while decryption, one with all shares can only retrieve the image, no one with any $2^n - 1$ share(s) will be able to reveal any information about the original image.

For monochrome image, an image is a collection of binary data 0 and 1 displayed as black and white pixels. Where, VC splits each pixel into white and black sub-pixels as shown in **Table 1**. If the pixel is white, then any one row among the top two rows is chosen to generate share 1 (S_1) and share 2 (S_2). If the pixel is black then any one row among the bottom two rows is chosen to generate S_1 and S_2 . At the time of superimposing of each pixel of S_1 and S_2 , the retrieval of the pixel is shown in the last column of **Table 1**.

Table 1. VC technique for encoding the pixels into two shares [1].

Pixel		S1	S2	S1 + S2
	$p = 0.5$			
	$p = 0.5$			
	$p = 0.5$			
	$p = 0.5$			

For color image as described in [12], there are mainly three inputs in the system i.e., RGB as well as RGBA color model. VC splits the input image into n shares and each RGB share is converted into 24-bit color image. The main parameters of VC include image contrast and the number of sub pixels of the retrieved image. The contrast of color image is comparatively different between the original and the retrieved images. A source image of $m \times n$ pixels needs two pieces of host images of the same size, where m and n represents the number of rows and columns, respectively. Mainly three processes build the overall system: pixel extraction, encryption and decryption.

In the pixel extraction phase the same positioned pixel from the RGB images are extracted and taken into account for the encryption process. For the inputs of the RGBA values, the technique reads from the pixel (0, 0) to the pixel (m , n). For example when a particular positioned pixel is scanned, a green and a brown pixel are obtained from the hosts, and a blue pixel is obtained from the source image. Now the problem is to encrypt the blue pixel with these share pixels so that the superimposed pattern reveals a bluish pattern.

In the encryption process, two share images of size $4m \times 2n$ are created. At first, the host pixels are expanded according to **Table 1**. Now the expanded pixel patterns are selected randomly. But whichever it takes, after the superimposing, the block turns to black. It happens because the opposite diagonal positions are colored black.

Next, the source pixel is expanded. It is done according to **Fig. 1**. In this process, four pixels are used to represent one pixel. Among the four pixels, one is the source color, one is black and

the rest two is transparent. The patterns are selected carefully so that the superimposing gives two black pixels and two color pixels. After the expansion process, three pieces of 2×2 blocks of pixels are obtained. These blocks are used to create the pixel patterns for the shares. The overall process is shown in Fig. 2.

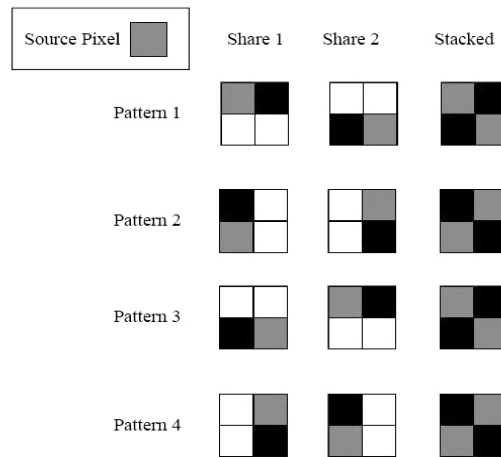


Fig. 1. Expansion and encryption of source pixel [12].

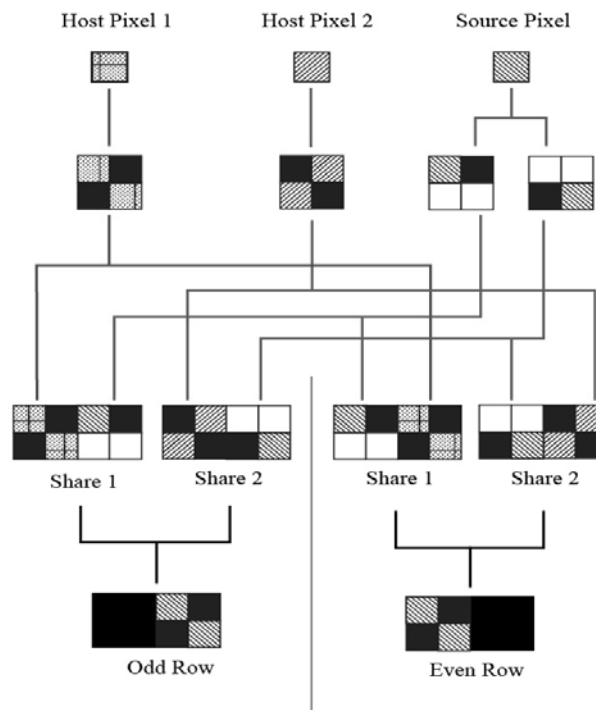


Fig. 2. Encryption and decryption for color image [12].

Say H_1 and H_2 are the host patterns and S_1 and S_2 are the source pixel patterns selected for the procedure. The final pixel patterns for the shares are generated by placing H_1 and S_1 together and H_2 and S_2 together. For the odd rows, S_1 and S_2 are placed at the right of H_1 and H_2 ,

respectively. For the even rows, it does the opposite. Two different actions for different rows are done for the perfect hiding of the source pixel. Now whatever the row is, the stacked pattern which is of the size 4×2 , consists of 2 color pixels and 6 black pixels.

3.2 Steganography

Steganographic technique [7] hides the input image inside another image known as cover image so that it reduces the suspicion of the intruder. LSB is one of the most common techniques used in steganography. In this technique, the LSBs from the pixel of the input image are replaced with the message information so that it cannot be observed by the human visual system. The reason is that the amplitude of the change is very small [9].

However in the proposed technique, instead of using the cover image, random secret key is exploited to encrypt LSBs of each chunk of each share applying OTP encryption technique which increases the security of the image as well as makes the encryption and decryption process faster than that of using the cover image. Here the number of LSBs of each chunk of each share must be equal to the number of bits of OTP key.

3.3 One Time Pad (OTP)

OTP [8] encryption technique is applicable for binary data, and it possesses perfect secrecy. Here the same secret key is shared separately by the sender and the receiver for encryption and decryption purposes respectively. Recalling that the length of the key is as same as the length of the message to be encrypted. In this technique, a plaintext is paired with the secret key where usually XOR operation is applied. Thus, each bit of the plaintext is encrypted by combining it with the corresponding bit from the pad. The data encrypted with the key based on the randomness have the advantage that theoretically there is no way to “break the code” by analyzing a succession of data.

3.4 Paillier Cryptosystem

Paillier [13] is a public key cryptosystem which is described below.

1. Key Generation:

Two large primes p and q are chosen randomly and independently of each other such that $\gcd((p-1), (q-1)) = 1$. Now $N = p \cdot q$ and $\lambda = \text{lcm}(p-1; q-1)$ is computed. A random integer is selected g where g 's order is a non-zero multiple of N (since $g \neq 1 \pmod{N}$). Ensuring that N divides the order of g by checking the existence of the following modular multiplicative inverse: $u = L(g^\lambda \pmod{N^2})^{-1} \pmod{N}$, where function L is defined as (Lagrange function) $L(u) = u^{-1} \pmod{N}$ for $u \equiv 1 \pmod{N}$. Now the public key is (N, g) and the private key is (p, q, λ) .

2. Encryption:

Plaintext is m where $m < N$. Finding a random r . Then the ciphertext is $c = g^m \cdot r^N \pmod{N^2}$.

3. Decryption:

The ciphertext $c < N^2$. Retrieval of plaintext $m = L(c^\lambda \pmod{N^2}) / L(g^\lambda \pmod{N^2}) \pmod{N}$.

4. Proposed Technique

The proposed image encryption technique consists of several stages. These are: random OTP secret key generation, image encryption and image decryption, as described below.

4.1 Random OTP Secret Key Generation

To generate the OTP key to be used for both encryption and decryption purpose, a random secret key is generated where in binary the length of the key is equal to the number of LSBs within each chunk of each share. Here already mentioned that at first the exploitation of VC generates n share images from an input image of $w \times z$ sized pixels where each share also consists of the same number of pixels. Here w and z both are positive integers where the image consists of w rows and z columns. Now for RGB or RGBA color model, steganography is applied which generates $(24 / 8) = 3$ LSBs from each chunk. Thereby from each share image, there exists $w \times z \times 3$ LSBs. Therefore, the number of bits of OTP key will also be equal to $w \times z \times 3$ bits. Before sending the key to the receiver, it is transformed from binary to integer value and then encrypted using the public key of Paillier cryptosystem. The receiver decrypts it using its secret decryption key of Paillier cryptosystem, transforms it from integer to binary to be used for the purpose of image decryption. Fig. 3 shows the key generation process.

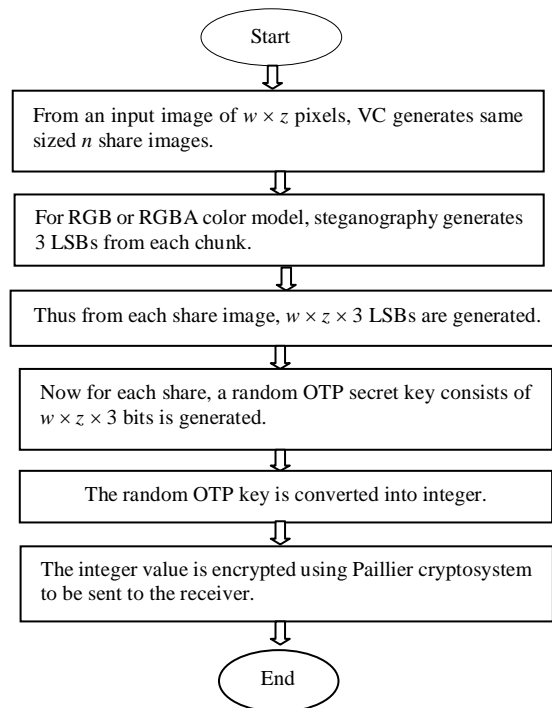


Fig. 3. Flow chart of random OTP secret key generation process.

4.2 Image Encryption Technique

This section describes the technique to encrypt the image. Fig. 4 depicts the process.

Step 1: In the proposed technique at first the sender scans the input image of $w \times z$ pixels. Now VC technique is applied on the input image that generates 2^n share images consisting of $w \times z$ pixels where n is greater than or equal to one.

Step 2: Converting pixels of each share image into binary values.

Step 3: Using steganography, the LSBs from each chunk of each share are computed. Here for RGB or RGBA color model, each chunk generates 3 LSBs. Thus for each share totally there are $w \times z \times 3$ LSBs.

Step 4: Now for each share, a random binary OTP secret key of $w \times z \times 3$ bits are generated. Thus for n shares, n times $w \times z \times 3$ bits are generated separately.

Step 5: For each share, the final cipher image is generated by applying OTP technique over LSBs along with OTP secret key. For example, for the first share image first $w \times z \times 3$ bits and similarly for the n -th share image n -th $w \times z \times 3$ bits are used as the OTP keys.

Step 6: At last the OTP keys are also encrypted by using asymmetric cryptosystem to send them to be used by the receiver as described in section 4.1.

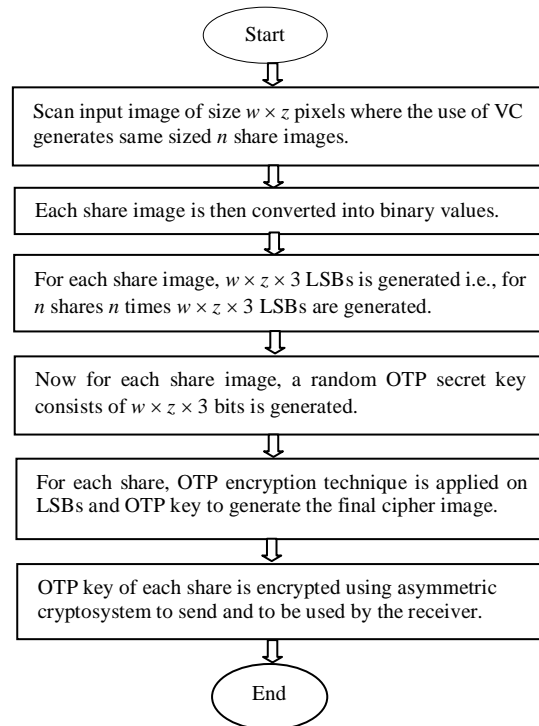


Fig. 4. Flow chart of image encryption technique.

4.3 Image Decryption Technique

This section describes the decryption process of the image. **Fig. 5** depicts the process.

Step 1: At first the receiver decrypts n different OTP keys using its secret key of asymmetric cryptosystem.

Step 2: Each different OTP key is converted from integer to its corresponding binary value. Thus it generates n different OTP keys of $w \times z \times 3$ bits.

Step 3: The receiver detects the LSBs of cipher image of each share image, and then applies OTP decryption technique (XOR operation) on them along with OTP secret key.

Step 4: This operation retrieves n different binary share images each consists of $w \times z \times 3$ bits of the input image.

Step 5: The binary image of each share image is converted into pixels which generate the share image of size $w \times z$ pixels.

Step 6: Finally the superimposing of all share images retrieve the original input image.

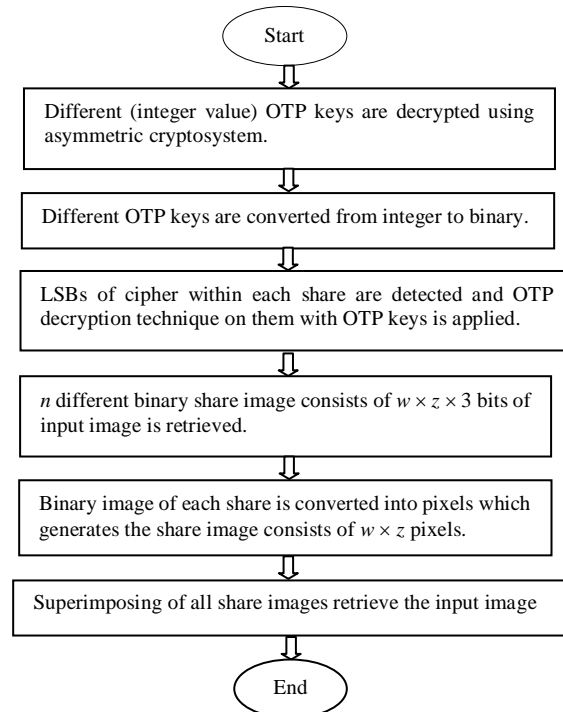


Fig. 5. Flow chart of image decryption technique.

5. Experimental Analysis

5.1 Experimental Setup

A prototype system of the proposed technique has been developed under the environment on Intel^(R) CoreTM i5-6200U 2.40 GHz x64-based processor with 8 GB of RAM running on Windows 10 operating system. The prototype has been developed in programming language MATLAB 16.0 [14]. Color images of various formats namely, 'jpg', 'bmp', 'png', 'gif', 'tif' etc. of same dimensions i.e., 200×200 pixels have been considered for the experiment. Here some images have been collected from our own sources that are shown in first column of **Fig. 6 (a)** and **Fig. 6 (b)** and are used to depict the output of the proposed image encryption and decryption techniques. Whereas some other images are obtained from [15] and they are used to show the required time of the proposed encryption and decryption techniques.

5.2 Output of Encryption Technique

This section presents the output of the encryption technique where consecutively VC, steganography and OTP are applied.

5.2.1 Output after applying VC

Step 1: Scan the input images of **Fig. 6 (a)** and **Fig. 6 (b)** where the sizes of the images are different but as already said that their dimensions of pixels are the same. Here the format of the first image is 'jpeg' while the second one is 'png'.

Step 2: After applying VC, two share images are generated which are presented in the third and fourth columns of **Fig. 6**.

5.2.2 Output after transforming into binary image

The pixels of the share images presented in the third and fourth columns of Fig. 6 are now converted into binary images. Here the output of a portion of share 1 image of Fig. 6 (a) is as follows:

```
01110101011110000111011101110111011110010111100110111101001111001011110
0100111010110
```

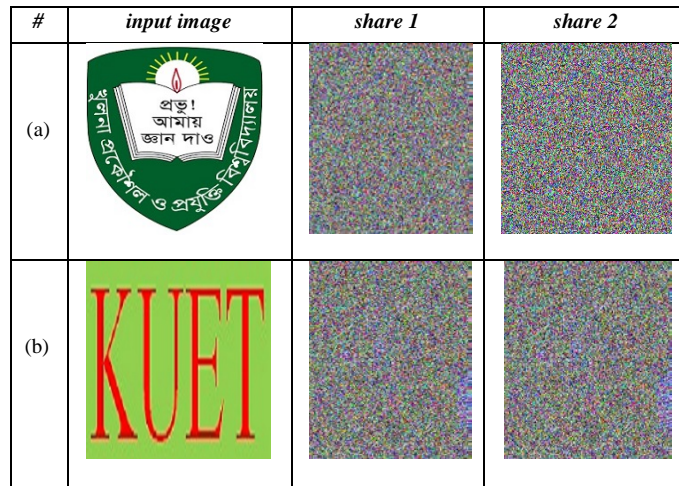


Fig. 6. Input image and the output after applying VC technique.

5.2.3 Output after applying steganography with OTP

Step 1: Applying steganography, LSBs from each chunk of binary image of each share are detected where the numbers of LSBs within each share are $150 \times 188 \times 3 = 84600$. Here LSBs within a portion of the binary image of share 1 of Fig. 6 (a) are shown as bold:

```
01110101011110000111011101110111011110010111100110111101001111001011110
0100111010110
```

Step 2: A random OTP secret key is generated where the number of bits is 84600. A portion of the key is as follows:

```
11010001010101010110010101010111010100110101001001010010110111000010111
0010111010
```

Step 3: On LSBs, OTP encryption operation (i.e., XOR operation) is applied using OTP key. A portion of the generated cipher is shown in Table 2.

Table 2. Generated final cipher of a share (a portion).

LSBs		0111010 1 01111000 0 111011 1 01111011100 1
Key	\oplus	1 1 0 1 0
Cipher		0 1 1 0 1

Now a portion of the final cipher is shown below where the changed bits are shown using red color:

```
01110100011110010111011101110110011110010111100110111101001111001011110
0100111010110
```

Also the pixel appearance of this binary image is shown in **Fig. 7**.

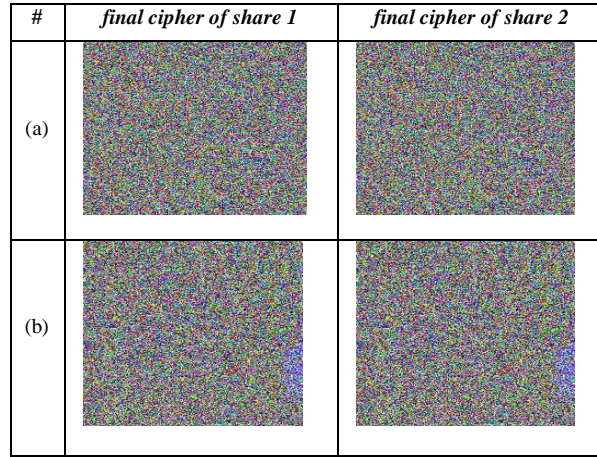


Fig. 7. Pixel appearance of final cipher of share images.

5.3 Sending OTP secret key to the receiver

Step 1: The sender transforms OTP secret key (consists of 84600 bits) from binary to integer.

Step 2: The sender encrypts the OTP key using the public encryption key of Paillier cryptosystem and sends it to the receiver.

Step 3: The receiver decrypts the integer value of OTP key using the secret decryption key of Paillier cryptosystem and transforms it into binary value to decrypt the image.

However, the OTP key encryption process is required only to transmit it securely, not the main part of image encryption and decryption; therefore these results are not presented herein.

5.4 Output of Decryption Technique

5.4.1 Output after applying steganography with OTP

Step 1: The receiver identifies LSBs from each chunk of each share of the final cipher. For example a portion of share 1 of **Fig. 7 (a)** is as follows:

011101000111100101110111011101110011110010111100110111101001111001011110
0100111010110

Step 2: On LSBs, OTP decryption operation (i.e., XOR operation) is applied using the same random OTP secret key. A portion of the retrieved plaintext of share 1 of **Fig. 7 (a)** is shown in **Table 3**.

Table 3. Retrieved plaintext of a share image (a portion).

LSBs	01110100011110010111011101110111001111001				
Key \oplus	1	1	0	1	0
Plaintext	1	0	1	1	1

Now a portion of the finally retrieved plaintext of share 1 is shown below where the changed bits are shown using red color:

011101010111110000111011101110111011110010111100110111101001111001011110
0100111010110

5.4.2 Output after transforming binary image into pixels

The binary image of each share is converted into pixels that are shown in first and second columns of Fig. 8. Thus share images are re-generated.

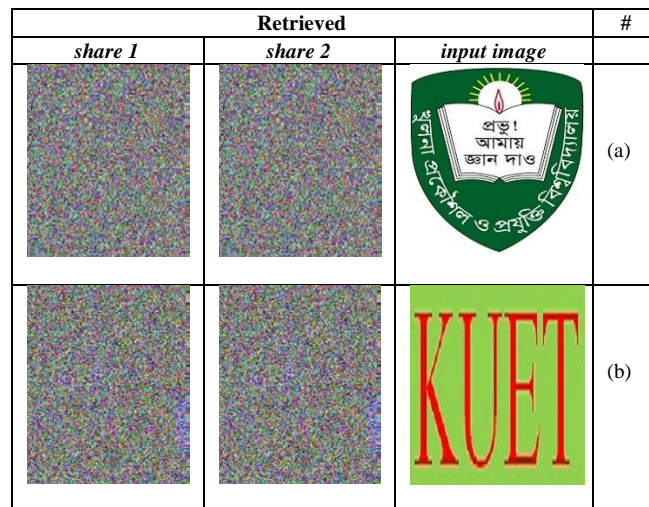


Fig. 8. Retrieved share images and input image.

5.4.3 Output after applying VC

Finally, the superimposing of share images retrieves the input image which is shown in the third column of Fig. 8.

5.5 Experimental Results and Comparisons

For the proposed technique, the time requirement of encryption and decryption processes has been presented in Fig. 9 and Fig. 10, respectively. Here, the encryption/decryption time has been shown under different input image sizes. As expected, with the increasing image size, encryption/decryption time also increases. However, comparatively encryption process requires more time than decryption. The reason is, in encryption stage the generation of shares of the image takes larger time than that of superimposing of shares in decryption stage.

Moreover considering the time requirement of encryption and decryption processes, the proposed technique has been compared with some other techniques proposed in [12], [6], [7] and [23] and the results of comparison has been shown in Fig. 11 and Fig. 12, respectively. Here to compare the techniques, images of same pixel sized i.e., 200 x 200 has been considered as the input although their sizes are different namely, 'jpg' (20.7KB), 'png' (64.2KB), 'bmp' (117KB), 'tif' (78.2KB) and 'gif' (18.7KB). The figure shows that techniques proposed in [12], [7] and [23] require faintly less time than the proposed technique. Where the technique proposed in [12] is only VC for RGB and RGBA color model, the

technique proposed in [6] is VC followed by steganography, the technique proposed in [7] is steganography followed by VC and the technique proposed in [23] is based on deep learning algorithms i.e., firstly it compresses the image applying SAE and then encrypts this one using chaotic logistic map. Although the proposed technique requires slightly more overall execution time than several compared techniques, it is not so high and a quiet reasonable one. Recalling that for the sake of providing secured image transmission over public medium, the proposed technique successively combines a number of techniques i.e., VC, steganography and OTP altogether. That's why, it requires more overall execution time.

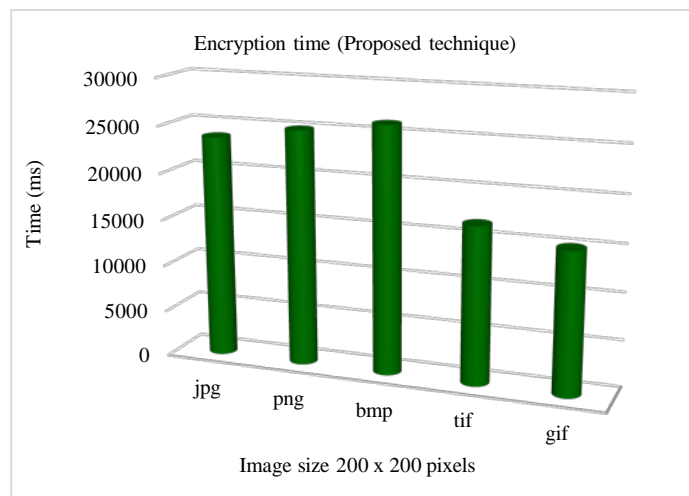


Fig. 9. Time requirement of encryption process for various images by the proposed technique.

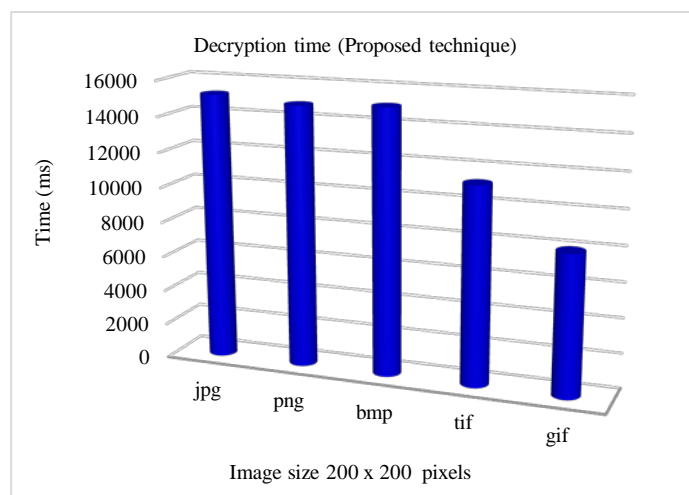


Fig. 10. Time requirement of decryption process for various images by the proposed technique.

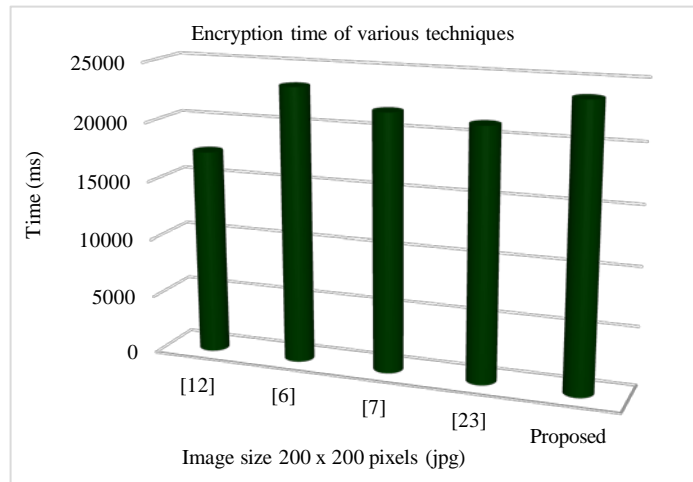


Fig. 11. Comparison in case of encryption time requirement among various techniques.

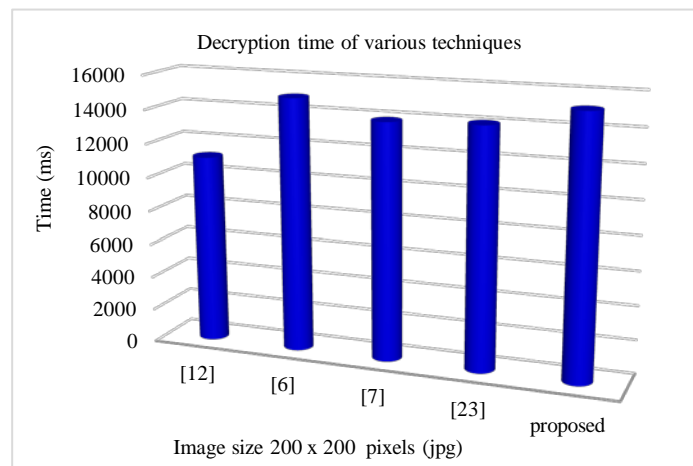


Fig. 12. Comparison in case of decryption time requirement among various techniques.

6. Security and Statistical Analyses

6.1 Histogram Analysis

Histogram of an image is used to plot the frequency distribution of its pixel values. Usually in an input image, the frequency distribution of pixel values remains imbalanced. But within the cipher image, because of the exploitation of encryption techniques, the distribution of pixel values exists uniformly. Thereby an adversary is unable to extract any useful information from the cipher image [17]. The proposed encryption technique also generates the cipher image with a uniform distribution of pixel values. For the input image of Fig. 6 (a), the histogram plot of both the original input image and the cipher image are presented in Fig. 13 (a) and Fig. 13 (b), respectively.

By comparing histogram plots i.e., by analyzing **Fig. 13**, it can be observed that there are significant differences between the original input image and the cipher image. It guarantees that the proposed encryption technique completely changes the characteristics of the input image. Finally after decryption, the histogram of the retrieved image is shown in **Fig. 13 (c)**. Here analyzing histogram plots it is also noticed that both the retrieved image and the original input image possess almost alike characteristics.

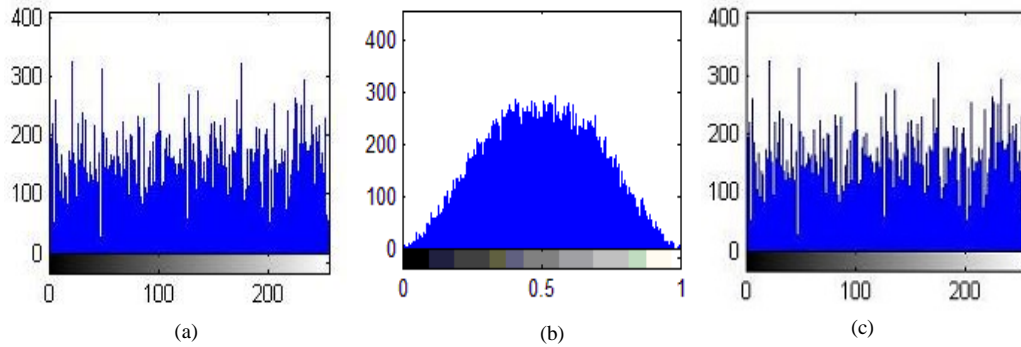


Fig. 13. Histogram plot for the image of **Fig. 6 (a)**:
 (a) Input image, (b) Cipher image, and (c) Retrieved image.

Two simple and very common attacks in the domain of internet are ‘Salt and Pepper noise attack’ and ‘chosen-plaintext attack’. For the image of **Fig. 6 (a)**, these attacks are simulated and described below.

6.2 Salt & Pepper Noise Attack

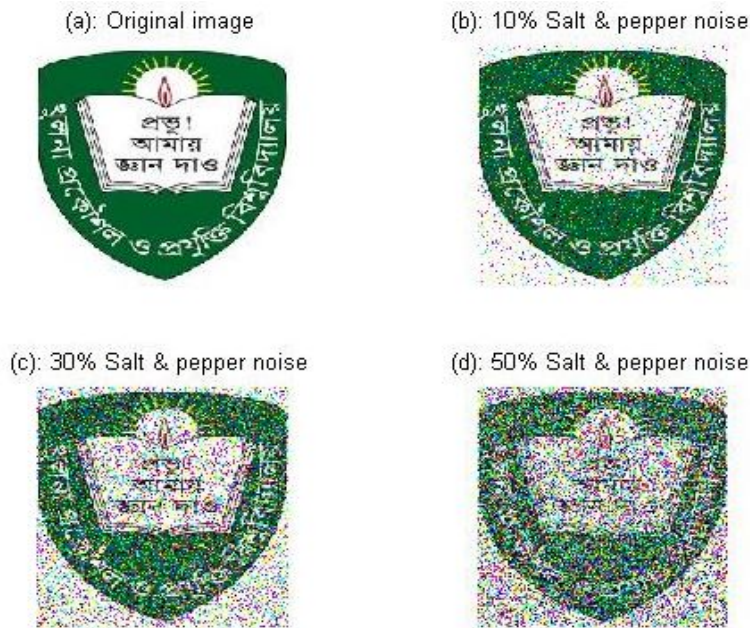


Fig. 14. (a) Original image. Salt and pepper noise attacked images: (b) 10%, (c) 30% and (d) 50%.

Salt-and-pepper noise is an impetuous noise which sparsely occurs with white and black pixels of an image [17]. Corruption of image by salt and pepper noise happens because of defective memory locations in hardware, dyeing down of signal in communication links, wounding of channel decoder, transmission over noisy channels, multi path wireless communication [18, 19] etc. Generally the damaged pixels set the value either minimum as 0 or maximum as 255 for salt and 0 ~ 8 for pepper noise [20]. In order to verify the strength of the proposed technique against this attack, the tempering of the cipher image is arranged with different levels of salt and pepper noise attacks namely, 10%, 30%, and 50% and their corresponding retrieved images are shown in Fig. 14 (b), Fig. 14 (c) and Fig. 14 (d) respectively. From these figures, it is noticeable that while the tempering of cipher pixels increases, the amount of corrupted output also increases. However even after these tempering, the cipher images are perceptually identifiable. This ensures that the proposed technique is capable to survive in noisy channels and defective storages where salt and pepper noise attack with different noise density may exist.

6.3 Chosen-Plaintext Attack (CPA)

In CPA, the attacker somehow obtains the corresponding cipher image for a chosen input image of its' choice [16, 17]. In the proposed technique in order to generate the OTP key to be used for both encryption and decryption purpose, a random secret key is generated in binary, where the length of the key is equal to the number of LSBs within each chunk of each share. As secret random value is used to generate the key value, in fact, no two different images will use the same random value. Thereby apparently every image will employ a distinct key value. Although the intruder may have subsequent access to the targeted chipper image, in no way the image as well as its key value is related to any other chipper image. Thus, the proposed encryption technique can survive the attack.

7. Conclusions

The proposed multi-stage encryption technique enhances the level of secrecy of image by combining VC, steganography and OTP consecutively. Thus it accumulates advantages of cryptosystems not relying on a specific key/keys with the strength of the cryptosystem relying on a specific key. According to the simulation results, although the time requirement of the proposed technique is alike or slightly higher than other related techniques, the level of its secrecy is certainly higher than those techniques. The underlying reason is that for encryption at first it adopts VC, then steganography and finally OTP technique where OTP technique cannot be cracked. While the encryption completes, the ultimate cipher image generated by the proposed technique is so confusing that it is quite impossible for any entity to guess the input image from the cipher image. The security and statistical analyses also ensure that it is almost impossible for the intruder or attacker to mount any form of attack on it. Thus the proposed technique possesses good imperceptibility and high-level security. A future plan of improvement is to incorporate an appropriate image compression technique with the proposed image encryption technique. Expected that through compression, the capacity of the image will be reduced which will lead to decrease the time requirement in the encryption and decryption of the proposed technique.

References

- [1] M. Naor and A. Shamir, "Visual cryptography," in *Proc. of Advances in Cryptology–EUROCRYPT'94*, pp. 1–12, Springer Berlin/Heidelberg, 1995.
[Article \(CrossRef Link\)](#)
- [2] G. Huayong, M. Huang and Q. Wang, "Steganography and Steganalysis Based on Digital Image," in *Proc. of 4th Int. Congress on Image and Signal Processing*, Vol. 1, pp. 252–255, IEEE, 2011.
[Article \(CrossRef Link\)](#)
- [3] M. Paunwala and S. Patnaik, "Biometric template protection with DCT based watermarking," *Machine Vision and Applications*, Vol. 25, No. 1, pp. 263–275, 2014. [Article \(CrossRef Link\)](#)
- [4] D. Aeloor and A. Manjrekar, "Securing Biometric Data with Visual Cryptography and Steganography," in *Proc. of Int. Symposium on Security in Computing and Communication*, pp. 330–340, Springer Berlin/Heidelberg, 2013. [Article \(CrossRef Link\)](#)
- [5] P. Marwaha and P. Marwaha, "Visual Cryptographic Steganography in Images," in *Proc. of Int. Conf. on Computing, Communication and Networking Technologies*, pp. 1 - 6, IEEE, 2010.
[Article \(CrossRef Link\)](#)
- [6] M. Pramanik and K. Sharma, "Analysis of Visual Cryptography, Steganography Schemes and its Hybrid Approach for Security of Images," *Int. Journal of Emerging Technology and Advanced Engineering (IJETA)*, ISSN 2250-2459, ISO 9001:2008, Vo. 4, No. 2, February 2014.
- [7] V. Lokeswara Reddy, A. Subramanyam and P. C. Reddy, "Implementation of LSB Steganography and its Evaluation for Various File Formats," *Int. Journal of Advanced Networking and Applications*, Vol. 02, No. 05, pp. 868-872, 2011.
- [8] S. Patil and A. Kumar, "Modified One Time Pad Data Security Scheme: Random Key Generation Approach," *Int. Journal of Computer and Security*, Vol. 3, No. 2, 2009.
- [9] E. Walia and P. J. Navdeep, "An Analysis of LSB & DCT based Steganography," *Global Journal of Computer Science and Technology*, Vol. 10, No. 1, pp. 4–8, April 2010.
- [10] J. Jesalkumari and R. R. Sedamkar, "Modified Visual Cryptography Scheme for Colored Secret Image Sharing," *Int. Journal of Computer Applications Technology and Research*, Vol 2, No. 3, pp 350 – 356, 2013. [Article \(CrossRef Link\)](#)
- [11] F. Liu and C–K. Wu, "Robust visual cryptography-based watermarking scheme for multiple cover images and multiple owners," *IET Information Security*, Vol. 5, No. 2, pp. 121–128, 2011.
[Article \(CrossRef Link\)](#)
- [12] M. T. I. Siyam, K. M. Rokibul Alam and T. Al Jami, "An Exploitation of Visual Cryptography to Ensure Enhanced Security in Several Applications," *IJCA*, ISSN: 0975 – 8887, Vol. 65, No.6, March 2013.
- [13] P. V. Parmar, S. B. Padhar, S. N. Patel, N. I. Bhatt and R. H. Jhaveri, "Survey of Various Homomorphic Encryption algorithms and Schemes," *Int. Journal of Computer Applications*, Vol. 91, No. 8, 2014.
- [14] "MATLAB 16.0" Retrieved on January 15, 2017, from [Article \(CrossRef Link\)](#).
- [15] Sample images (accessed on October 05, 2017) from [Article \(CrossRef Link\)](#).
- [16] D. S. Laiphrakpam and M. S. Khumanthem, "Cryptanalysis of symmetric key image encryption using chaotic Rossler system," *Elsevier, Optik*, Vol. 135, pp. 200-209, 2017.
[Article \(CrossRef Link\)](#)
- [17] D. S. Laiphrakpam and M. S. Khumanthem, "A robust image encryption scheme based on chaotic system and elliptic curve over finite field," *Springer, Multimedia Tools and Applications*, pp. 1–24, May 2017. [Article \(CrossRef Link\)](#)

- [18] S. S. Al-amri, N.V. Kalyankar and S. D. Khamitkar, "A Comparative Study of Removal Noise from Remote Sensing Image," *Int. Journal of Computer Science*, Vol. 7, Issues 1, pp. 33-36, 2010. [Article \(CrossRef Link\)](#)
- [19] G. Judith and N. Kumarasabapathy, "Study And Analysis of Impulse Noise Reduction Filters," *An Int. Journal of Signal & Image Processing (SIPIJ)*, Vol. 2, No.1, March 2011.
- [20] S. Rohith and K. H. Bhat, "A simple robust digital image watermarking against salt and pepper noise using repetition codes," *Int. Journal on Signal and Image Processing*, Vol. 3, No. 1, 2012.
- [21] X. Wang, S. A. Okamoto, R. R. Lear and N. L. Ishigo, "Multi-stage watermarking process and system," *U.S. Patent 6, 912, 294*, issued June 28, 2005.
- [22] F. Liu, J. Tang, Y. Song, Y. Bi, and S. Yang, "Local structure based multi-phase collaborative representation for face recognition with single sample per person," *Information Sciences*, Vol. 346, pp.198-215, 2016. [Article \(CrossRef Link\)](#)
- [23] F. Hu, C. Pu, H. Gao, M. Tang, and L. Li, "An image compression and encryption scheme based on deep learning," arXiv preprint arXiv:1608.05001, 2016.
- [24] F. Hu, J. Wang, X. Xu, C. Pu and T. Peng, "Batch Image Encryption Using Generated Deep Features Based on Stacked Autoencoder Network," *Mathematical Problems in Engineering*, Vol. 2017, ID 3675459, pp. 1-12, 2017. [Article \(CrossRef Link\)](#)
- [25] Z. Li, and J. Tang, "Weakly supervised deep metric learning for community-contributed image retrieval," *IEEE Trans. on Multimedia*, Vol. 17, No. 11, pp. 1989-1999, 2015. [Article \(CrossRef Link\)](#)
- [26] Z. Li, and J. Tang, "Weakly supervised deep matrix factorization for social image understanding," *IEEE Trans. on Image Processing*, Vol. 26, No. 1, pp. 276-288, 2017. [Article \(CrossRef Link\)](#)



Arindom Mondal is currently a student of M. Sc. program in the Dept. of Computer Science and Engineering of Khulna University of Engineering & Technology, Bangladesh. He has completed his B.Sc. degree in Computer Science and Engineering from Patuakhali Science and Technology University, Bangladesh. His research interests include applied cryptography and information security.



Kazi Md. Rokibul Alam is currently a professor in the Dept. of Computer Science and Engineering of Khulna University of Engineering & Technology, Bangladesh. He received Dr. (Eng.) degree in System Design Engineering from University of Fukui, Japan, and M.Sc. and B. Sc. degrees both in Computer Science and Engineering from Bangladesh University of Engineering & Technology and Khulna University, Bangladesh in 2010, 2004 and 1999 respectively. His research interests include applied cryptography, information security and machine learning.



G. G. Md. Nawaz Ali received his B.Sc. degree in Computer Science and Engineering from the Khulna University of Engineering & Technology, Bangladesh in 2006, and the Ph.D. degree in Computer Science from the City University of Hong Kong, Hong Kong in 2013 with the Outstanding Academic Performance Award. He is currently a postdoctoral research fellow with the Department of Automotive Engineering, The Clemson University International Center for Automotive Research (CU-ICAR), Greenville, SC, USA. From October 2015 to March 2018, he was a postdoctoral research fellow with the School of Electrical and Electronic Engineering of Nanyang Technological University (NTU), Singapore. He is a reviewer of a number of international journals including the IEEE Transactions on Intelligent Transportation Systems and Magazine, IEEE Transactions on Vehicular Technology, Wireless Networks etc. His current research interests include Vehicular Cyber Physical System (VCPS), wireless broadcasting, mobile computing, and network coding.



Peter Han Joo Chong is currently a Professor and Head of Department of Electrical and Electronic Engineering at Auckland University of Technology, Auckland, New Zealand. He received the B.Eng. (with distinction) in Electrical Engineering from the Technical University of Nova Scotia, Halifax, NS, Canada, in 1993, and the M.A.Sc. and Ph.D. degrees in Electrical and Computer Engineering from the University of British Columbia, Vancouver, BC, Canada, in 1996 and 2000, respectively. He has visited Tohoku University, Japan, as a Visiting Scientist in 2010 and Chinese University of Hong Kong (CUHK), Hong Kong, between 2011 and 2012. He is currently an Adjunct Faculty at the Department of Information Engineering, CUHK. He was previously an Associate Professor (tenured) from 2009 to 2016 and Assistant Professor from 2002 to 2009 in the School of Electrical and Electronic Engineering at Nanyang Technological University (NTU), Singapore. Between 2011 and 2013, he was an Assistant Head of Division of Communication Engineering. Between 2013 and 2016, he was a Director of Infinitus, Centre for Infocomm Technology. He was the recipient of 'EEE Teaching Excellence Award' and 'Nanyang Award Excellence in Teaching' in 2010, and 'Nanyang Education Award (College)' in 2015. In 2015, he became a Fellow of the Teaching Excellence Academy in NTU. From February 2001 to May 2002, he was a Research Engineer at Nokia Research Center, Helsinki, Finland. Between July 2000 and January 2001, he worked in the Advanced Networks Division at Agilent Technologies Canada Inc., Vancouver, BC, Canada. He is the Co-Founder of P2 Wireless Technology based in Hong Kong. He is an Editorial Board Member of Security and Communication Networks, Wireless Sensor Network, and an Editor of Far East Journal of Electronics and Communications, and KSII Transactions on Internet and Information Systems. His research interests are in the areas of mobile communications systems including radio resource management, multiple access, MANETs/VANETs, multihop cellular networks and Internet of Things/Vehicles.



Yasuhiko Morimoto is currently a Professor at Hiroshima University. He received his B.E., M.E. and Ph.D degrees from Hiroshima University in 1989, 1991 and 2002 respectively. From 1991 to 2002, he had been with IBM Tokyo Research Laboratory where he worked for data mining project and multimedia database project. Since 2002, he has been with Hiroshima University. His current research interest includes data mining, machine learning, geographic information system and privacy preserving information retrieval.