Two-level hiding an encrypted image

Faten H. MohammedSediq Al-Kadei
Computer Department, Northern Technical University, Iraq

Article Info

Article history:

Received Aug 16, 2019
Revised Oct 1, 2019
Accepted Nov 2, 2019

Keywords:

Cryptography
Fractal image
Hiding image
Image decryption
Image encryption
Steganography

ABSTRACT

Encryption and hiding images are becoming a hot research area and a broad prospect for application. This article uses a secure algorithm with Low Significant Bit method to hide an encrypted high-resolution color bitmap image in two selectively color images (i.e., two cover images). The paper introduces a two-level hiding encrypted image using MATLAB-GUI programming language. In the beginning, with a key image using XOR bit operation, the original RGB image is encrypted. After that, the encrypted image is hidden into the first cover image. The cover image is then hidden into another cover to make the secret image safer without changing the perceptual quality for both covers. Then, the algorithm is tested on many bitmap images, which can be an important image, fingerprint image, samples of secret medicine, or bank account pattern. The correlation histograms demonstrate a high correlation for all encrypted images. The PSNR is used to find steganography quality for the two cover images after hiding the secret image showing a high quality for the two levels of hiding operation

Corresponding Author:

Faten H. MohammedSediq Al-Kadei,
Computer Department,
Northern Technical University,
Kirkuk, Iraq.
Email: faten.alqadhi@ntu.edu.iq

1. INTRODUCTION

Security became an important issue in image storage and communication, and one of the important ways the security can be ensured is encryption. In many applications, data hiding is essential; the basic methods for hobbyists are steganography and cryptography, secretive data transmission, user privacy, and so on.

The main aim of image encryption techniques via hiding image is converting the original image to another one to be difficult to understand. In other words, its primary goal is keeping image confidentiality between users. Therefore, no one knows its content without a key for decryption. Reliable security is crucial in sending and receiving digital images in different applications such as online personal photo album, medical system images, image communications in military and videos of confidential conferences, etc.

The basic methods for hobbyists are as follows: steganography and cryptography, secretive data transmission, user privacy, etc. It has been proven that steganography is the best method of safety. Furthermore, three different methods of hiding information are generally used: steganography, cryptography and watermarking. There are different types of steganography [1, 2]:

a) Steganography of Text.
b) Steganography Image.
c) Steganography Audio.
d) Steganography of Videos.

The basic principle of steganography in all of these methods is that any secret message can be embedded in another cover object that may have no meaning in such a way that encrypted. Lastly, the displayed data is only the cover data. Therefore, containing hidden information cannot be easily detected unless proper decryption is used. Each steganography has three components:

a) Secret Object.
b) Cover Object of the message.
c) Steganographic objects results.

Using specific techniques, the information that has to be hidden is encoded in cryptography; this information is generally understood as coded data that appears nonsensical. Basically, steganography is information hiding, which cannot be identified as the coded information seems to be abnormal in its presence by sight and undetectable. Furthermore, steganography detection is named steganalysis. Moreover, steganography is one of the powerful techniques used to hide a secret image, audio files, and videos so that no one can find a secret hidden image in another one. Besides, steganography image refers to hiding in another image or video file information (i.e., text messages, images, or audio files) and by proper decoding technique, this hidden information can be retrieved. Therefore, steganography is defined as “the art of hiding data media files”, which means the way of hiding any message (e.g., text, image, audio, etc.) in any file (e.g., .mp3, wav or .png) [3, 4]. In the literature, there are several methods of steganography performance; however, the Least Significant Bit (LSB) is considered as the most famous one. This is mainly because there are three components in each color image; this pixel data is saved in one byte in an encoded format. It is possible to modify the low bits that contain little information for each pixel to store the hidden text. The precondition for the stored text that it must be smaller or equal to the image that is used to hide the text. There are two different image steganography methods:

a) Methods of space.
b) Transform methods: LSB method is the most common method used in the spatial substitution method.
c) Retrieve the message images from recombining the lower four-bit planes.

The image pixels are assigned to be stored in the form of bits. The intensity of pixels is stored in an 8-bit (i.e., 1 byte) image on a grayscale. Similarly, each pixel requires 24 bits (8 bits for each layer) for color (red, green, and blue (RGB)) images. When the LSB bit is modified, the Human Visual System (HVS) is unable to detect these changes in pixels’ intensity or color. This is obscure visual redundancy, which can be used to hide information in these bits with no significant difference is observed in the image. Accordingly, it is possible to modify the first bits to carry the information to store a hidden text in each pixel or a picture (see Figure 1). By replacing the LSB in a BMP type picture, it embeds data into the photo [1].

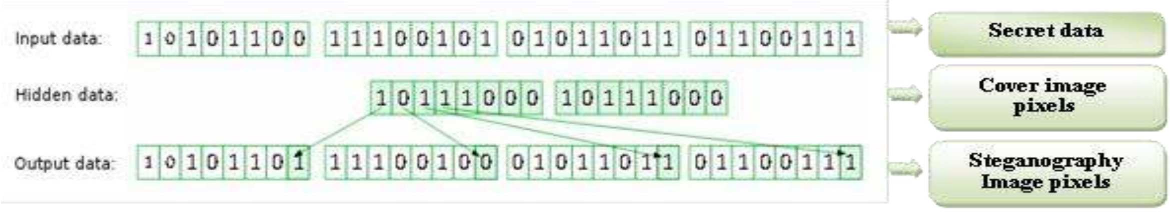


Figure 1. Hiding information in the LSB

Since it has the smallest effect on the amount of color, the smallest possible impact on the picture will replace this bit with a bit of the hidden data. Using this method, the image steganography embeds the secret in each pixel of the cover image in the LSBs. It is well known that the LSB based steganography is one of the simplest techniques to hide pixel values without perceptible distortions from a secret message in the LSBs. The changes in LSB value are invisible to a human eye.

In the literature, cryptography is considered as an effective way to securely transfer information. It scrambles the image information before transmitting it to change its structure. Therefore, the attacker cannot be hacked because it is hard to get back the original image. So, it provides an image’s modified shape, but it does not hide the image even if it is a good security method. The main objective is to provide the original image with better protection [5, 6].

Image encryption is a process of information transformation (known as a plain image) by using an algorithm (referred to as a cipher), making it incomprehensible to any person except those with special knowledge (known as a key). This process results in an encrypted image (called a cipher image). The process of converting the cipher-image back to its original shape so that it can be perceived is called Decryption. Furthermore, image encryption is usually used to protect images (computer and storage devices images) when personal records are exposed to loss or theft of laptops or backup drives. It helps to guard against uncovering and sharing. Moreover, encryption is also applied to protect the transit data, such as data transmitted through networks (e.g., e-commerce and internet), wireless systems, mobile phones, Bluetooth devices, etc. Besides, the cryptography technique is further classified into two major types: symmetric and asymmetric key cryptography. For purposes of both encryption and decryption, symmetric-key cryptography requires a single private key. On the other hand, the asymmetric key uses two keys first for encryption and the other for decryption purposes [7, 8].

The fractal image can be efficiently used as a key in the encryption methods because of its difficulty to break against attacks. The generation of the fractal image can be changed significantly when a small change occurred in any of its parameters. Therefore, it is suitable to implement a secure encryption key. In the literature, several papers have been proposed on image encryption using a fractal key for image encryption [9-12]. On the other hand, steganography is a data hiding technique internet. For example, CAPTCHA codes are generated and later send them in an encrypted version. These codes are embedded ASCII into a cover image with an encrypted form resulting stego image. Thus, using image steganography, attackers cannot fetch the actual CAPTCHA resulting in a secured transmission of confidential data via the internet [13].

A modification of the pixel value Differencing (PVD) algorithm is used for Image Steganography in the spatial domain. It is normalizing secret data value by encoding method to make the new pixel edge difference. Consequently, it is less among three neighbors (horizontal, vertical and diagonal) and embedding data only to less intensity pixel difference areas or regions. The strength of this scheme is that any random hidden secret data do not make any shuttle differences to Steg-image compared to the original image. Furthermore, the bit plane slicing is used to analyze the maximum payload that has been embedded into the cover image security [14].

Since the last four decades, many efforts have on developing been paid to develop several Public Key Crptography (PKC) algorithms, where comparative trends of these algorithms were based on each algorithm. The roadmap of PKC algorithms with the most chosen algorithms among previous researchers has very recently been presented in [15]. In this respect, the stegSVM model as an embedding technique in steganography that has exploited the human visual system through Shifted LSB has shown an expected performance. The performance evaluation of stegSVM based on imperceptibility and robustness and compared to the other previous models in the image steganography domain is best described in [16]. It is worth mentioning that the steganography is the art or technique of hiding message date inside a carrier file. It can be performed in such a way the unauthorized (or unsolicited) personal is not capable of detecting the presence of data inside the carrier file. The method provides improved security and improved high embedding capacity image steganography through the usage of Integer Wavelet Transform (IWT) and Chaotic Logistic map. LSB technique is used to replace the bits in the coefficient of a detail band [17].

The secure algorithm to encrypt images after compressing using fractal image compression (FIC) is a good compression and encryption properties. The decoding process is composed of iterated contract transforms. The initial image range block is transformed by the iterated fractal transform, which makes the initial image contracted to the original image. The uses of FIC offer the advantage of increased security because of the use of the fractal codes instead of the original image. Finally, when some of the fractal coefficients changed using the encryption process, the image cannot be transformed correctly to the original image [18-25].

2. PROPOSED METHOD

Image encryption techniques attempt to convert the original image to another image that is hard to understand. Accordingly, it keeps the image to be confidential among users to ensure information security when much sensitive information is stored on computers and sent over the internet. Therefore, our method is proposed used to hide a bitmap color image (original) in another bitmap color image (cover image) to make the hidden image (secret image) more secure. As shown in Figure 2, in the first stage, the generated key image is used to encrypt the original image creating the secret image. The second stage of security works is to hide the secret image after encrypting it into two cover images. The first image is used as cover images to the secret image, while the second one is used as the cover image to hide the first cover; in consequence, providing two-level hiding systems.

The encryption is then achieved by first generating of the encryption key image using different computation methods and the XOR operation. Then, encrypted image pixels are hidden in another image (i.e., cover1) in the LSB to create the level-1 image. This image is then hidden in another image (i.e., cover 2) to generate the level-2 image to add an extra layer of security (see Figure 3). Since each pixel needs 8 bytes to be hidden in the cover image, the secret image should be less than the cover image (Cover1) at least by 1/8. Additionally, Cover1 must be less than the Cover2 by 1/8; in consequence, these covers are stretched to be of the proper size. As illustrated in Figure 2, the main encryption and hiding algorithm are summarized in the following steps:

step-1: Load the color bitmap image (Original image).
step-2: Generate a key image.
step-3: Encrypt the original image with the generated key to get the Secret one.
step-4: Load the cover image (Cover1).
step-5: Hide the encrypted Secret image in Cover1 and generating a level-1 stego image.
step-6: Load the second cover image (Cover2).
step-7: Hide the level-1 stego image in Cover2 and generating a level-2 stego image.
step-8: Output Stego image.
step-9: End.

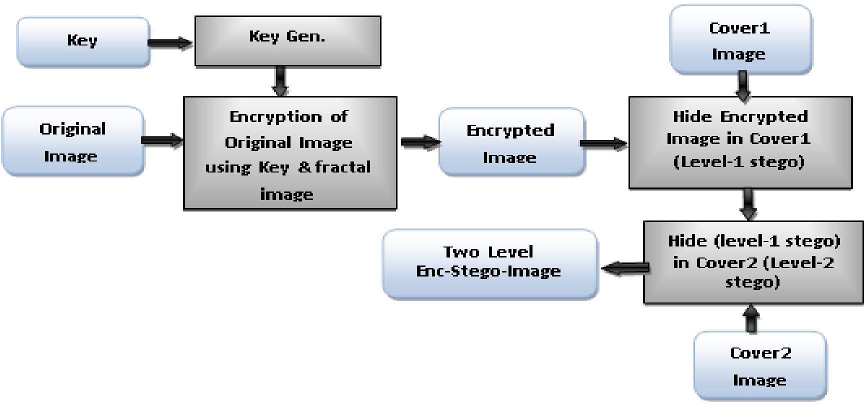

Key
Key Gen.
Original Image
Encryption of Original Image using Key & fractal image
Encrypted Image
Cover1 Image
Hide Encrypted Image in Cover1 (Level-1 stego)
Hide (level-1 stego) in Cover2 (Level-2 stego)
Two Level Enc-Stego-Image
Cover2 Image

Figure 2. Encryption and hiding steps diagram

The key generation Algorithm mentioned in step-2 in the above algorithm is applied using the following steps (Figure 3):

step-1: Load the color bitmap image (Original image).
step-2: Apply the function (Reshape) on the Secret image; this function changes the size and the position of the image pixels to obtain the Final Secret image.
step-3: Find the mean factor by using Mean function.
step-4: Input two External Keys to be used in key image generation.
step-5: Apply calculations using the above-selected keys and mean factor on the Original image to create a key image by taking the mod of each pixel.
step-6: Rotated the key image (180) degree using rotation function (imrotate).
step-7: Combine the images from step-1 and step-5 using bitwise XOR function to obtain the Secret image.
step-8: Output Secret image.
step-9: End.

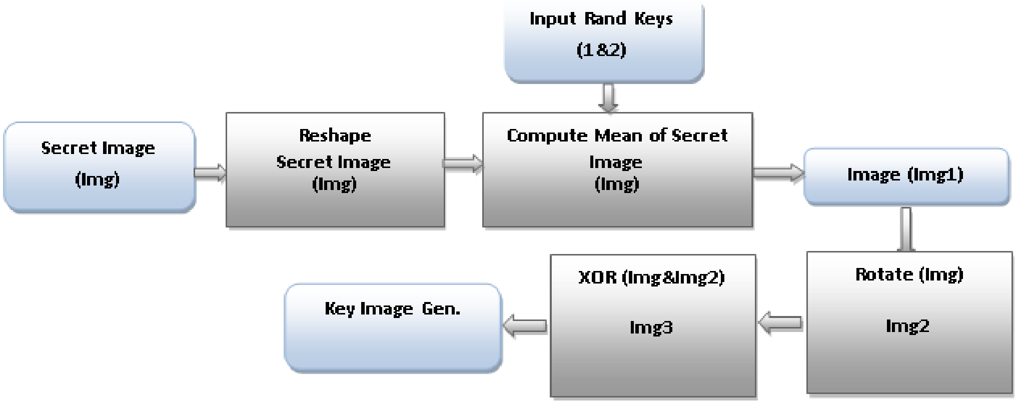


Figure 3. Key image generation diagram

To get back the original image, the Decryption and Un-hiding steps are applied in reverse sequence as shown in the following algorithm (see Figure 4):

step-1: Input level-1 Stego image (Cover1).
step-2: Get level-1 stego image from the Level-2 stego image (hidden in Cover2).
step-3: Get the secret encrypted image from the level-1stego image.
step-4: Input secret keys
step-5: Generate the Key image using the above generation algorithm.
step-6: Decrypt the secret image using the key image in reverse steps of the Encryption to get the Original image.
step-7: Output the original image.
step-8: End

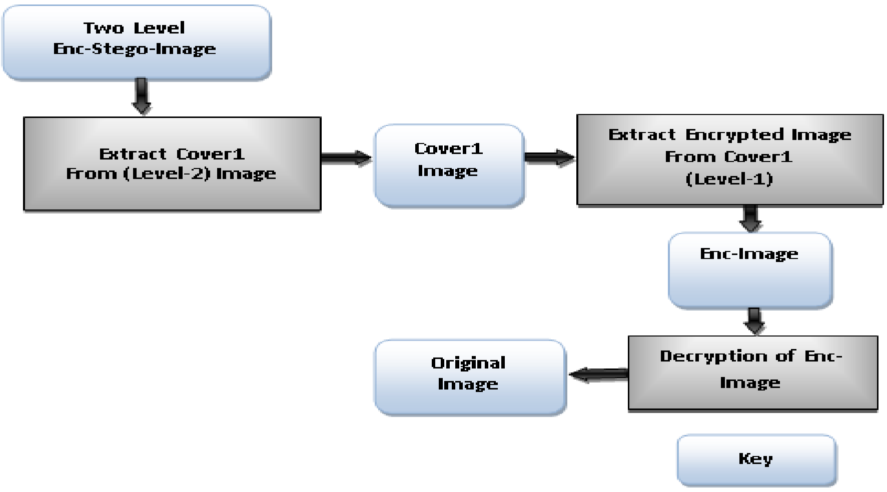


Figure 4. Decryption and un-hiding steps diagram

3. THE GUI SYSTEM FOR PROPOSED METHOD

All operations of the system are performed through a designed system using Matlab Graphic User Interface (GUI) tools to make all systems operations easier. Several examples are illustrated in Figures A1 to A6 in Appendix A. The system operations that the designed GUI system can perform are listed below:

a) Selecting, loading, and displaying the original image.
b) Encrypting the original image with input keys and display the encrypted image.
c) Displaying the histograms of both the original and encrypted images.
d) Selecting and load Cover1.
e) Hiding the encrypted image in cover-1(level-1 stego image) and display it.
f) Selecting and loading Cover2.
g) Hiding the encrypted Cover1 (level-1 stego image) in Cover-2 (level- stego image).

h) Displaying the final stego image.
i) Unhiding the Cover1 from level-2 stego image.
j) Unhiding the secret image from (level-1 stego image).
k) Decrypting the secret image using the generated key image.
l) Lastly, calculate the PSNR for cover images.

4. RESULTS AND ANALYSIS

The proposed method has been tested with different secure images (Fprint, Medim, GISim, Codeim and Eyeim) and different cover images (Nature, Bird, fruit, Baboon, Car, etc.). The resulted images are tested using histograms in Matlab application and human vision as follows:

a) The first security stage is Encryption operation. Using two random secret keys to create secret image encryption gave the system a random key image. The system has tested several times with different keys, and each time it provided a different key image (Figure 5).
b) As shown in Figure 6, the encrypted secret images are not intelligible in vision because of using different methods on the secure image before encrypting it with the generated key image.
c) Comparing histograms of the original images and the encrypted image showed a high correlation for the encrypted image pixels (Figure 6).
d) The second security stage is hiding operation has done on two levels. The first level is hiding the secret image in the first cover, while the second level is hiding this cover again in another cover. The PSNR has then calculated to find the quality of each cover image (cover image level-1 & level -2). All tested results showed a high quality for all covers in both levels of hiding operation, and it was more than 50 Db (Figure 7).

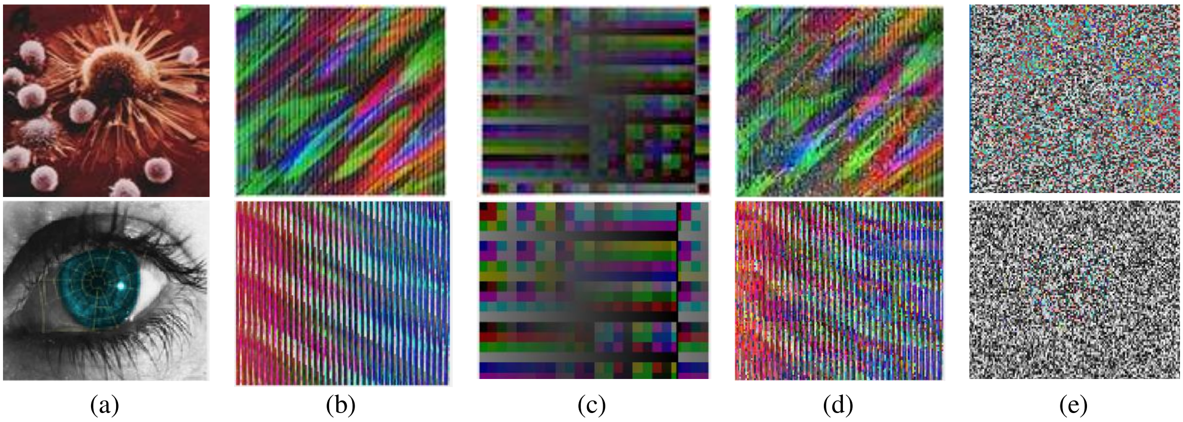


Figure 5. Secret image encryption steps (a) Original image (b)Reshape image (first image) (c) Generated secret image using key1&2 (d) Combine first & secret images (e) Encrypted image

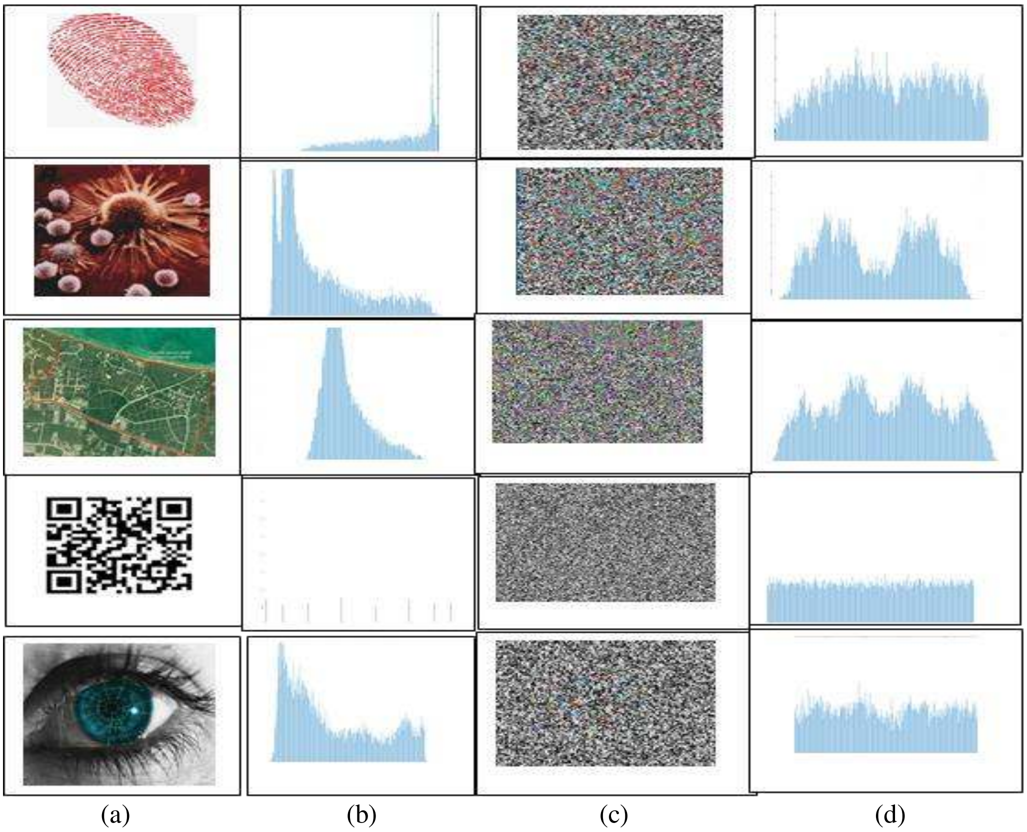


Figure 6. Histograms of original secret images and encrypted images (a) Secret image (b) Secret image histogram (c) Encrypt image (d) Encrypt image histogram

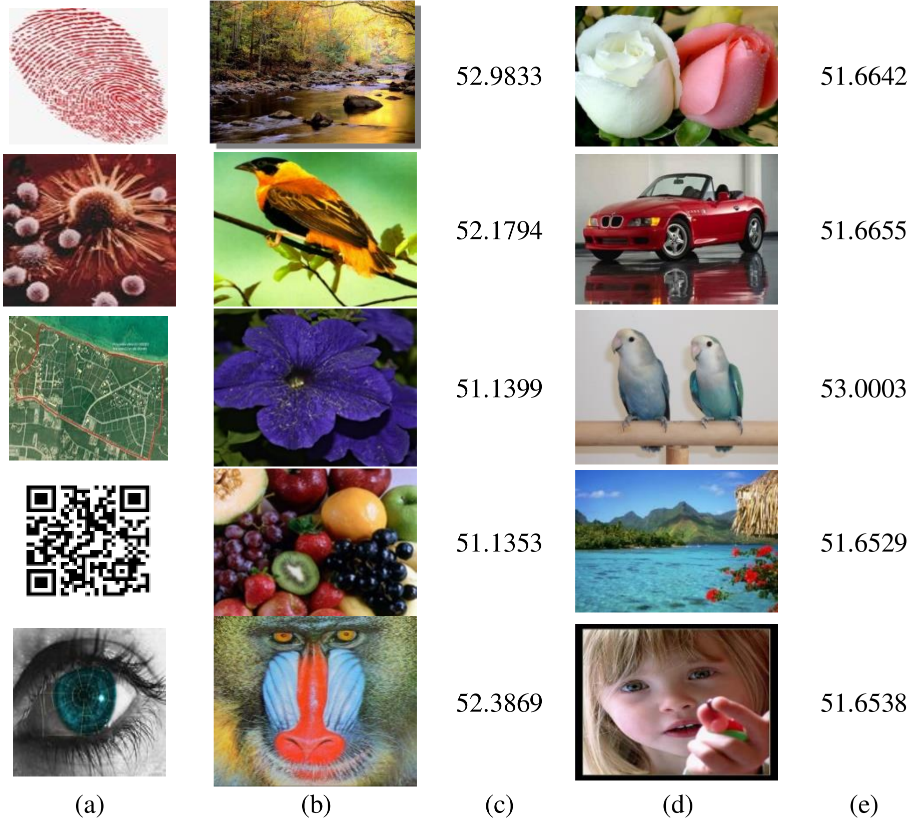


Figure 7. The PSNR of the cover images (Cover1and Cover2) after hiding operation (a) Secure image (b) Cover image level-1 (c) PSNR of cover1 (d) Cover image level-2 (e) PSNR of cover2

5. CONCLUSION

In this article, we used new symmetric cryptography algorithms to encrypt and hide a secure color image in another color image. The findings showed that the proposed encryption technique had provided the system with a high secured image. This approved by the high correlation shown in the resulted histograms that prevent attacks on the secret image.

Using two-level hiding to hide a color image provided efficient security for the secret image. Both used cover images were at high-quality PSNR after hiding the secret image information, because of using the LSB technique. The broad range of cipher key ensures the secret image's integrity and authenticity.

REFERENCES

[1] S. Bhallamudi, "Image Steganography," in *EE7150 – Digital Image Processing*, 2015, pp. 1-17.

[2] Z. Alqadi, B. Zahran, Q. Jaber, B. Ayyoub, J. Al-Azzeh, and A. Sharadqh, "Proposed Implementation Method to Improve LSB Efficiency," *Int. J. Comput. Sci. Mob. Comput.*, vol. 8, no. 3, pp. 306-319, 2019.

[3] M. G. Anchal Chander Lekha, "Hiding an Image Data into Video Stenography Using Different Algorithm and MATLAB: A Review," *Int. J. Comput. Sci. Trends Technol.*, vol. 6, no. 2, pp. 12-16, 2018.

[4] A. B. M.S. Bouridah, T. Bouden, "*Fractional Chaos Synchronization for Color Image Encryption*," in Third International Conference on Technological Advances in Electrical Engineering (ICTAEE'18.), 2018, pp. 1-8.

[5] K. D. Patel and S. Belani, "Image encryption using different techniques: A review," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 1, no. 1, pp. 30-34, 2011.

[6] R. Kaur and E. K. Singh, "Image encryption techniques: a selected review," *J. Comput. Eng.*, vol. 9, no. 6, pp. 80-83, 2013.

[7] A. Nag *et al.*, "*Image encryption using affine transform and XOR operation*," in 2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies, 2011, pp. 309-312.

[8] R. M. Rad, A. Attar, and R. E. Atani, "A new fast and simple image encryption algorithm using scan patterns and XOR," *Int. J. Signal Process. Image Process. Pattern Recognit.*, vol. 6, no. 5, pp. 275-290, 2013.

[9] S. K. Mitra, C. A. Murthy, and M. K. Kundu, "A study on partitioned iterative function systems for image compression," *Fundam. Informaticae*, vol. 34, no. 4, pp. 413-428, 1998.

[10] S. K. Abd-El-Hafiz, A. G. Radwan, S. H. A. Haleem, and M. L. Barakat, "A fractal-based image encryption system," *IET Image Process.*, vol. 8, no. 12, pp. 742-752, 2014.

[11] K. S. N. Raju, M. V. A. Kumar, and M. V. V. M. Latha, "Image Encryption and Decryption Using Scan Pattern," *Int. J. Electron. Electr. Comput. Syst. IJEECS*, vol. 5, no. 5, 2016.

[12] G. B. Huntress, "Encryption using fractal key." *Google Patents*, 24-Aug-2004.

[13] S. Pramanik, R. P. Singh, and R. Ghosh, "A new encrypted method in image steganography," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 14, no. 3, pp. 1412-1419, 2019.

[14] S. S. N. Bhuiyan, N. A. Malek, O. O. Khalifa, and F. D. A. Rahman, "An Improved Image Steganography Algorithm based on PVD," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 10, no. 2, pp. 569-577, 2018.

[15] J. I. Ahmad, R. Din, and M. Ahmad, "Analysis review on public key cryptography algorithms," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 12, no. 2, pp. 447-454, 2018.

[16] H. S. Hussain, R. Din, M. H. Ali, and N. Balqis, "The Embedding Performance of Stego SVM Model in Image Steganography," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 12, no. 1, pp. 233-238, 2018.

[17] N. Krishnaveni and S. Periyasamy, "A novel and innovative approach for image steganography with chaos," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 11, no. 1, pp. 263-267, 2018.

[18] A. M. F. and A. A.-D. Heyam Maraha, Kameran Ali Ameen, Dalya Raad Abbas, Raghad Zuahir Yousif, "Evaluating the Robustness of Image Watermarking System based on Multilevel Wavelet Transform," *J. Eng. Appl. Sci.*, vol. 14, no. 23, pp. 8712-8720, 2019.

[19] H. Gao, Y. Zhang, S. Liang, and D. Li, "A new chaotic algorithm for image encryption," *Chaos, Solitons & Fractals*, vol. 29, no. 2, pp. 393-399, 2006.

[20] N. A. Yousif and F. H. Al-Qadhee, "Applying Encryption Method to Color FIC," *Al-Ma'mon Coll. J.*, no. 26, pp. 248-268, 2015.

[21] C.-H. Chuang, Z.-Y. Yen, G.-S. Lin, and Z.-W. Hong, "A Virtual Optical Encryption Software System for Image Security," *J. Converg. Inf. Technol.*, vol. 6, no. 2, 2011.

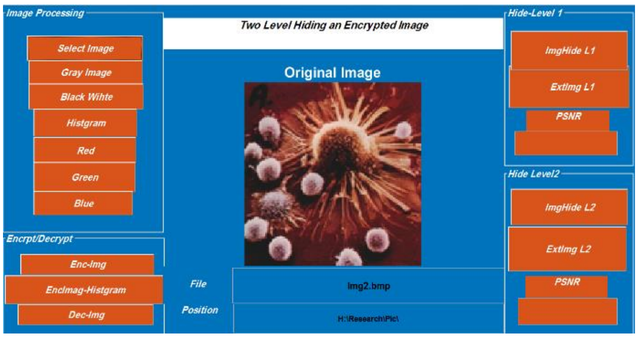
[22] N. A. Minas, F. H. MohammedSediq, and A. I. Salih, "Color Image Encryption Using Hybrid Method of Fractal-Based Key and Private XOR Key," *Kirkuk Univ. J. Sci. Stud.*, vol. 13, no. 1, pp. 104-117, 2018.

[23] L. Abraham and N. Daniel, "Secure image encryption algorithms: A review," *Int. J. Sci. Technol. Res.*, vol. 2, no. 4, pp. 186-189, 2013.

[24] H. M. Al-Najjar, "Digital image encryption algorithm based on multi-dimensional chaotic system and pixels location," *Int. J. Comput. Theory Eng.*, vol. 4, no. 3, p. 357, 2012.

[25] A. Akgul, S. Kacar, and B. Aricioglu, "A new two-level data hiding algorithm for high security based on a nonlinear system," *Nonlinear Dyn.*, vol. 90, no. 2, pp. 1123-1140, 2017.

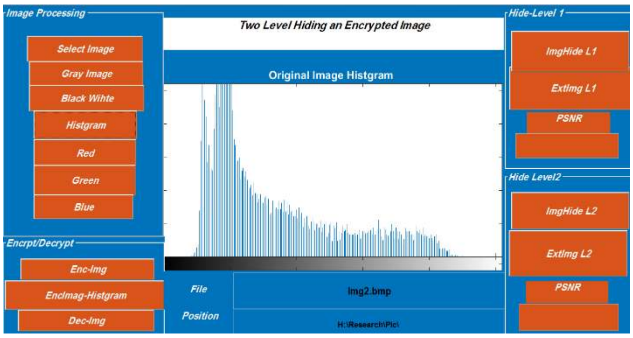
APPENDIX (A)



File Img2.bmp

Position H:\Research\Pic\

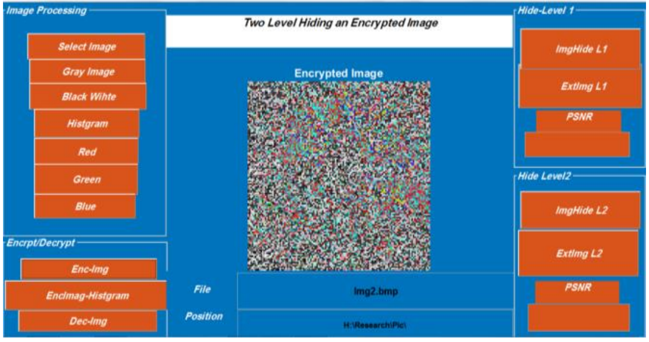
Figure A1. GUI of the system showing the original loaded image



File Img2.bmp

Position H:\Research\Pic\

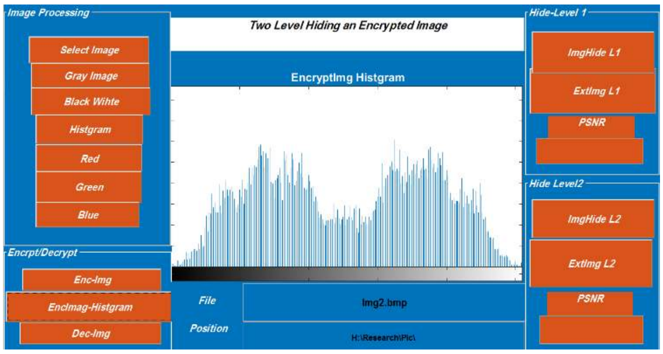
Figure A2. GUI of the system showing the Histogram of the original secret image



File Img2.bmp

Position H:\Research\Pic\

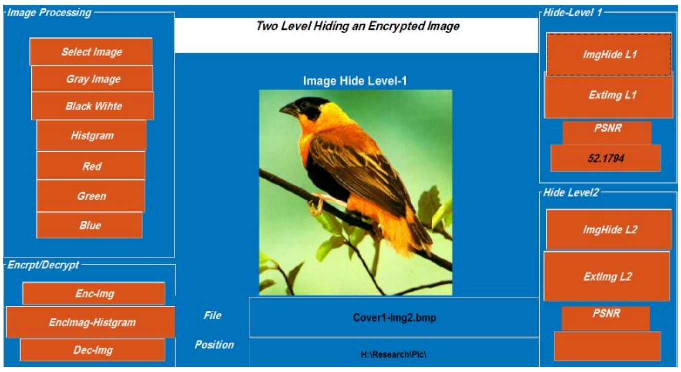
Figure A3. GUI of the system showing the encrypted secret image



File Img2.bmp

Position H:\Research\Pic\

Figure A4. GUI of the system showing the Histogram of the encrypted secret image



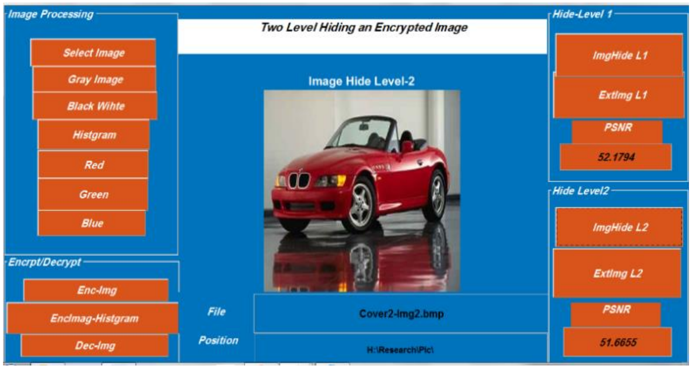
File: Cover1-Img2.bmp

Position: H:\Research\Pic\

PSNR: 52.1794

PSNR:

Figure A5. GUI of the system showing the Cover1 that contains the encrypted secret image



File: Cover2-Img2.bmp

Position: H:\Research\Pic\

PSNR: 52.1794

PSNR: 51.6655

Figure A6. GUI of the system showing the Cover2 that contains the Cover1 image