DEVELOPMENT OF AN IMAGE STEGANOGRAPHY USING LEAST SIGNIFICANT BIT

Adenusi Dauda Adeite[1], Adeboje Olawale[2], Ojedapo Halleluya[3], Babatunde Olalekan Lawal [4], Latifat Odeniyi [5]

[1]Adenusi Dauda Adeite, Dept. of Mathematical and Computing Sciences, KolaDaisi University, Ibadan.
Email: dauda.adenusi@koladaisiuniversity.edu.ng
[2]Second Author Affiliation & Address Font size 11
[3]Example: Professor, Dept. of xyz Engineering, xyz college, state, country

---***---

Abstract – *Ensuring information security is a crucial aspect of exchanging messages or communicating information over the internet. Confidentiality of communication is often protected using various techniques, with cryptography being a prominent method used for multimedia. However, in certain scenarios, simply keeping the content of messages confidential may not be sufficient, and concealing the very existence of the message, known as steganography, may offer enhanced security. Steganography involves hiding a confidential message within digital files, such as images, audio, or video files. However, hiding information is not just enough, there is also need for encryption of the secret message before hiding it in the cover image. This research work tries to hide text file and encrypt it using Elliptic Curve Cryptography into a cover image of any formats using least significant bit (LSB). The stego file was evaluated using standard metrics such as Mean Square Error and Peak Signal Noise Ratio.*

Key Words: Cover Image, Stego Key, Stego Image, Steganography, Secret Message

1.INTRODUCTION

Hiding information is a subfield of security which encompasses a wide range of methods for digitally concealing information. One significant area within information hiding is steganography. Steganography deals with writing of hidden messages in a way that no one can suspect the existence of the message, apart from the sender. This is a form of security through obscurity. [1].

Steganography has emerged as a compelling and extensively researched field with diverse applications. An essential aspect of steganography is the careful selection of a suitable cover image to effectively hide a specific secret message, ensuring the security of the resulting stego image [1]. In this paper, an image steganography was developed that utilizes the Least Significant Bit (LSB) steganography technique in combination with cryptography.

Cryptography, the practice of concealing information through encryption, safeguards data by converting it into an incomprehensible form. Its utility lies in ensuring secure transmission across public networks. Through cryptographic algorithms, the original plaintext undergoes transformation into ciphertext, which can only be reversed into plaintext by individuals possessing the secret key. Cryptography serves as a well-established and widely employed method for securing sensitive data. It has been integrated into cloud computing technology by numerous cloud service provider[2].

The two basic information encryption techniques are symmetric encryption and asymmetric encryption. Symmetric encryption is also called secret key encryption and Asymmetric encryption is also called public key encryption. Elliptic Curve Cryptography (ECC) is an asymmetric encryption type, which is the counterpart of modular multiplication in Rivest-Shamir-Adleman (RSA) and the counterpart of modular exponentiation is multiple additions [3]

Elliptic Curve Cryptography (ECC) is one of the strongest and most efficient cryptographic techniques in modern cryptography. Elliptic curve cryptography helps in providing a high level of security with smaller key size compared to other cryptographic technique which depends on integer factorization or discrete logarithmic problem [4].

Least significant Bit (LSB) is the simplest approach for embedding information in cover image is the use of least significant bit (LSB). This technique embeds the bits of the message directly into least significant bit plane of the cover image. In a deterministic sequence, modulating the least significant bit does not result in human perceptible difference because the amplitude of the change is very small. To hide a secret message inside an image, a proper cover image is required, because least significant bit method uses bits of each pixel in the image, and also it is very important to use a lossless compression of a data compression algorithm because when using a lossless compression, it will retain the hidden information but when a lossy compression is used the hidden information will be lost in the transformation process of the lossy compression [5].

2. RELATED WORKS

[6] proposed an algorithm that hides secret information bits in the Least Significant Bit (LSB) of the Inverse Wavelet Transform (IWT)'s approximation coefficients, applicable to both grayscale and color images. The Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) is employed for assessing the statistical variance between the cover media and the generated stego image, whereas cross-correlation quantifies their degree of resemblance.

[7] presented a technique that utilizes the Most Significant Bit (MSB) of the image for data embedding. In this approach, secret bits are stored in the 5th bit position of the cover image, taking into account the difference between the 5th and 6th bits of the secret data. If the difference between the 5th and 6th bits of the stego image deviates from the secret bit, the 5th bit is modified accordingly. By using the MSB instead of the commonly targeted LSB, this method prevents detection of hidden information by attackers who typically focus on LSB manipulation.

[8] proposed a reversible data hiding approach that utilizes encrypted binary images. Binary images, which have minimal redundancy and can result in visible distortions when pixels are randomly switched, are divided into non-overlapping blocks where all pixels are either white or black, and non-uniform blocks where both white and black pixels coexist. A type image is employed to differentiate between these types of blocks. The type image is self-embedded, and a stream cipher is used to encrypt the preprocessed image. Both types of blocks are utilized for embedding data. To retrieve the information, a data embedding key is used for secret data recovery, an encryption key is used for original image recovery, and both keys are arranged in a "T" pattern for secret and original data recovery. Additionally, both the encrypted and decrypted domains can be used to extract secret data.

[9] proposed a model that utilizes autoencoder networks to embed secret information into a selected cover media, generating a stego image at one end of the model network, and then extracting the embedded secret from the generated stego image at the other end using an extractor network. The autoencoder networks reduce the image data to lower dimensions and use that data to train a convolutional neural network (CNN) for the process of hiding and extracting the secret information.

[10] proposed a method that utilizes a convolutional GAN (Generative Adversarial Network) to embed secret information. In their approach, the secret information is first mapped to a noise vector, which is then used by a generator neural network to generate a cover image with features derived from the noise vector. Unlike traditional embedding methods, this approach does not require explicit embedding of the secret information into the carrier image, as the carrier image itself is generated based on the properties of the noise vector containing the secret information. At the receiver end, the carrier image is decoded using another neural network to obtain the noise vector and thereby retrieve the embedded secret information.

3. METHODOLOGY

The research is divided into two modules, namely: the embedding module and the extracting module. In the embedding module, the secrete message (plain text) is encrypted using Elliptic curve cryptography.

Embedding Process:

1. Input the secret text (message) that to be hide in the cover image: here we provide the secret message we want to conceal in the cover image
2. Enter the key for Encoding: The steganography process requires a key (stego-key) to ensure the security of the hidden message. This key acts as additional secret information, like a password or encryption key, that is used during the embedding process.
3. Select the cover image (bmp, tif, png and jpg files) from list of images: we choose a suitable cover image in which we want to hide the secret message. A cover image is an ordinary image that is publicly visible and doesn't look suspicious. It can be in various formats like BMP, TIFF, PNG or JPEG. The secret message will be embedded into the pixels of this cover image, creating the stego-image. For the purpose of this research, we will stick with using high quality PNG image with high bit depth so it can accommodate a lot of colours.

4. Save the stego-image: After the secret message is successfully embedded into the cover image using the stego-key, the resulting image is known as the stego-image or stego-object. This stego-image appears the same as the original cover image to the naked eye but contains the hidden secret message.

Extraction Process:

Extraction refers to the process of retrieving the embedded message from the stego object. In the realm of steganography, terminology surrounding detection and circumvention of steganographic schemes mirrors that of cryptography, albeit with notable distinctions. Similar to how a cryptanalyst employs Cryptanalysis to decipher encrypted messages, a stegonalyst utilizes steganalysis to uncover concealed information.

Following the creation and transmission of the stego-object through a communication channel, assuming an ideal channel scenario, the stego-object is correctly received by the decoder circuit. The decoder, equipped with both the extraction key and the stego object as inputs, yields a single output—the secret text.

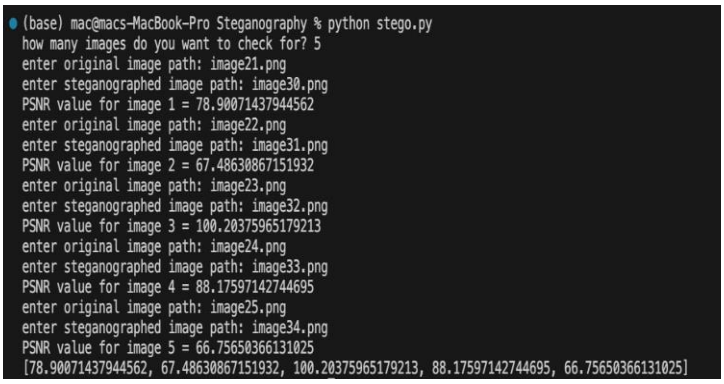
The major algorithm for the Extracting stage can be listed as follow:

1. Enter the key for Decoding (same key for encoding): During the embedding stage, the stego-key was used to hide the secret message within the cover image. Now, during the extraction stage, the same stego-key is required to decode and retrieve the hidden information. This key serves as the cryptographic key to unlock the hidden message from the stego-object.

2. Select the stego-image (image that resulting from encoding stage): here, we select the stego-image, which is the image that was created during the embedding stage. This stego-image contains the hidden secret message, but to the naked eye, it appears like an ordinary cover image, as the hidden data is imperceptible.

3. Get the secret text (message): With the correct stego-key and the stego-image, the decoder circuit processes the stego-image to extract the secret text (message) that was hidden within it during the embedding stage. The extraction process uses the same stego-key to reverse the embedding process and reveal the concealed information.

4. RESULT



```
(base) mac@macs-MacBook-Pro Steganography % python stego.py
how many images do you want to check for? 5
enter original image path: image21.png
enter steganographed image path: image30.png
PSNR value for image 1 = 78.90071437944562
enter original image path: image22.png
enter steganographed image path: image31.png
PSNR value for image 2 = 67.48630867151932
enter original image path: image23.png
enter steganographed image path: image32.png
PSNR value for image 3 = 100.20375965179213
enter original image path: image24.png
enter steganographed image path: image33.png
PSNR value for image 4 = 88.17597142744695
enter original image path: image25.png
enter steganographed image path: image34.png
PSNR value for image 5 = 66.75650366131025
[78.90071437944562, 67.48630867151932, 100.20375965179213, 88.17597142744695, 66.75650366131025]
```

Figure 1: Image showing the peak signal-to-noise ratio (PSNR) of the original image and the steganographed image.

Table 1: Result of Peak Signal-to-noise Ratio

Images	PSNR Value
Image 1	78.90071437944562
Image 2	67.48630867151932
Image 3	100.20375965179213
Image 4	88.17597142744695
Image 5	66.75650366131025

5. CONCLUSIONS

This research demonstrates that LSB steganography is a reliable and effective method for securely hiding messages within images without visibly altering the cover images. The technique's imperceptibility and robustness against visual detection, along with the high PSNR values obtained, further validate its practicality for secret communication and data hiding applications. However, it is essential to remain cautious about potential security risks and take appropriate precautions when utilizing steganography for sensitive information exchange.

REFERENCES

[1] Timothy, Adeboje Olawale, Adetunmbi Adebayo, and Gabriel Arome Junior. "Embedding text in audio steganography system using advanced encryption standard, text compression and spread spectrum techniques in Mp3 and Mp4 file formats." International Journal of Computer Applications 177 (2020): 46-51.

[2] Anjali Krishna, A., & Manikandan, L. C. (2020). A Study on Cryptographic Techniques.

[3] Zhang, Q. (2021, January). An overview and analysis of hybrid encryption: the combination of symmetric encryption and asymmetric encryption. In 2021 2nd international conference on computing and data science (CDS) (pp. 616-622). IEEE.

[4] Yan, Y. (2022, December). The Overview of Elliptic Curve Cryptography (ECC). In Journal of Physics: Conference Series (Vol. 2386, No. 1, p. 012019). IOP Publishing.

[5] Shojae Chaeikar, S., Zamani, M., Abdul Manaf, A. B., & Zeki, A. M. (2018). PSW statistical LSB image steganalysis. Multimedia Tools and Applications, 77, 805-835.

[6] Emad, E., Safey, A., Refaat, A., Osama, Z., Sayed, E., & Mohamed, E. (2018). A secure image steganography algorithm based on least significant bit and integer wavelet transform. Journal of Systems Engineering and Electronics, 29(3), 639-649.

[7] Islam, A. U., Khalid, F., Shah, M., Khan, Z., Mahmood, T., Khan, A., ... & Naeem, M. (2016, August). An improved image steganography technique based on MSB using bit differencing. In 2016 Sixth International Conference on Innovative Computing Technology (INTECH) (pp. 265-269). IEEE.

[8] Ren, H., Lu, W., & Chen, B. (2019). Reversible data hiding in encrypted binary images by pixel prediction. Signal Processing, 165, 268-277.

[9] Subramanian, N., Cheheb, I., Elharrouss, O., Al-Maadeed, S., & Bouridane, A. (2021). End-to-end image steganography using deep convolutional autoencoders. IEEE Access, 9, 135585-135593.

[10] Nie, D., Trullo, R., Lian, J., Wang, L., Petitjean, C., Ruan, S., ... & Shen, D. (2018). Medical image synthesis with deep convolutional adversarial networks. IEEE Transactions on Biomedical Engineering, 65(12), 2720-2730.