Al-Salam Journal for Engineering and Technology
Journal Homepage: http://journal.alsalam.edu.iq/index.php/ajest
e-ISSN: 2790-4822 p-ISSN: 2958-0862

Designing and implementing an information steganography system for transmitting encrypted text messages in digital images

Areej Saad Kahlaf [1]*, Abdul Monem S. Rahma[2], Ahmad Ghandour[1]

[1]Department of Computer and Communication Engineering, Islamic University of Lebanon, Wardanieh, LEBANON.
[2]Computer Science Department, Al-Maarif University College, Anbar, IRAQ.

*Corresponding Author: Areej Saad Kahlaf

DOI: https://doi.org/10.55145/ajest.2024.03.01.007
Received August 2023; Accepted October 2023; Available online November 2023

ABSTRACT: This danger can undoubtedly be eliminated using cryptography and steganography. And Combining encryption with steganography can assist in achieving high levels of data security, making this approach superior to using either of these techniques alone to send data across an unreliable route that is open to hacking. Visible steganography is one of the most secure steganography techniques. When the data text is inserted, the image's color frequency may vary slightly, which would be highly evident to anyone. We suggest levels of data security to get around the image's apparent behavior, where the language of the data is first transformed into an unintelligible cipher. An additional degree of protection is added by incorporating the cipher into an encrypted picture file before dividing it into shares. As a result, the ideas of steganography and cryptography are both applied to offer two levels of security and then a visual cryptography system is used to partition the picture into shares so that it may be transferred over a network channel.

Keywords: Steganography, Cryptography, ciphertext, M-LSBR, PSNR

1. INTRODUCTION

One of the most frequent hazards to a networked system is unauthorized access to data, which can result in losing the assets' availability, confidentiality, and integrity. One method for preventing unwanted access to data and information is steganography. Steganography is a technique for concealing sensitive data in a cover file. Text, digital images, audio, and video files can all be included in the cover file. The stego, or message-wrapped cover, is transmitted securely and with little danger of being intercepted by a recipient via the Internet. Cryptography scrambles data so unauthorized parties cannot decipher it, while steganography goes further. Contrarily, steganography conceals the whole, random [4]. There have been questions raised about the image steganography technique because of random noise issues, which degrades the stego picture quality and is added to the cover during concealment. This paper provides a detection-free method for reducing the impact of random variation in stego images. This goal is accomplished by enhancing the common Littlest Significant Bit Replacement technique and using a 24-bit color bitmap as the cover. The study's goal is to close any research gaps found in previous, relevant papers.[13]

2. STAGES

2.1 ENCRYPTION STAGE

The user enters the original information to be encrypted at this step. The original data (confidential information) is the text the user enters in the available text space [1]. DES is the employed encryption algorithm (Advanced Encryption Standard).

*Corresponding author: areej.saad@baghdadcollage.edu.iq

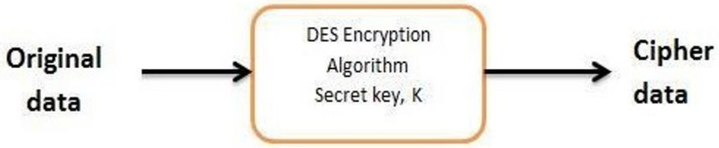


FIGURE 1. - Encryption stage. [downloaded from Google]

2.2 ENCODING STAGE

The gathered encrypted data is written into a suitable picture during the encoding procedure. The data will be encrypted into an image that the user may choose from a selection of images after getting the results of the previous step in the text field. A.png or.jpg image files are loaded and transformed into byte encoding when they are chosen [1]. The acquired byte representation makes it easier to modify images. Additionally, material that is secret or encrypted is converted into byte format. The encryption data is added to the image byte array in small chunks using bit-wise techniques, starting with the least significant bit. [2]. The encrypted text is kept in a picture known as steganography.

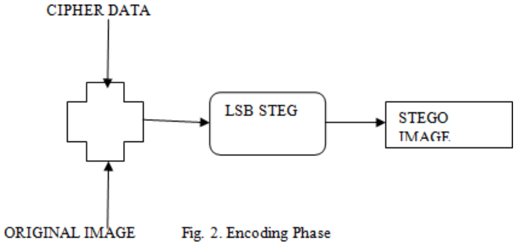


FIGURE 2. - Encoding stage. [Downloaded from Google]

2.3 DECODING STAGE

The encrypted data is extracted from the stego picture during the decoding stage. This is accomplished by using the sender-side encoding method in reverse [5].

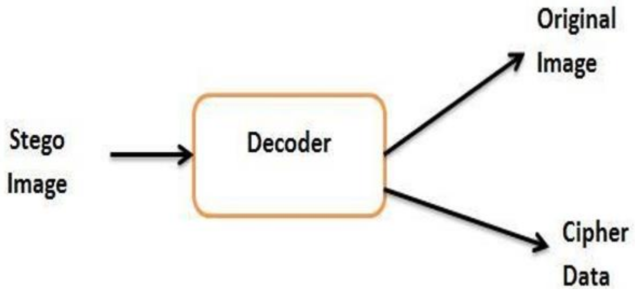


FIGURE 3. - Decoding stage. [Downloaded from Google]

2.4 DECRYPTION STAGE

In the decryption phase, the cipher data is converted into the original data. The technique used to encrypt the original Data utilizes duplicate private keys to decode the data and return it to its original form. The AES algorithm employs the same private keys (encrypting key) in the opposite direction to the encryption [9]. The original text will then be shown.

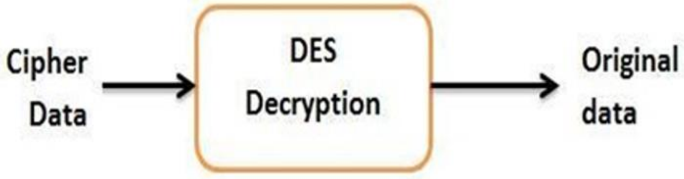


FIGURE 4. - Decryption stage [downloaded from Google]

The Visible Cryptography Scheme (VCS) is an additional sophisticated approach that adds one more layer of protection, even if it can occasionally be difficult to properly move the key from one person to another [7].

3. SCHEME FOR VISIBLE CRYPTOGRAPHY (VCS)

It is a sharing protocol called a visual cryptographic scheme (VCS) that enables the secret encoding of pictures into shares that can be sent to users. It is recommended to utilize this approach because it doesn't require any prior understanding of cryptography. Meaningful shares make up a visual cryptographic scheme's (VCS) components. The method, in this instance, entails creating By embedding random shares into real-world covered shares. A VCS is created [7]. They offer comparable visual quality to other types of encryption systems. The VCS's job is to accept a secret picture as input and produce shares that meet the next specifications.

- Any number of eligible subsets of shares can successfully recover the secret picture.
- Any banned shares subset cannot get information about the hidden picture other than its size.

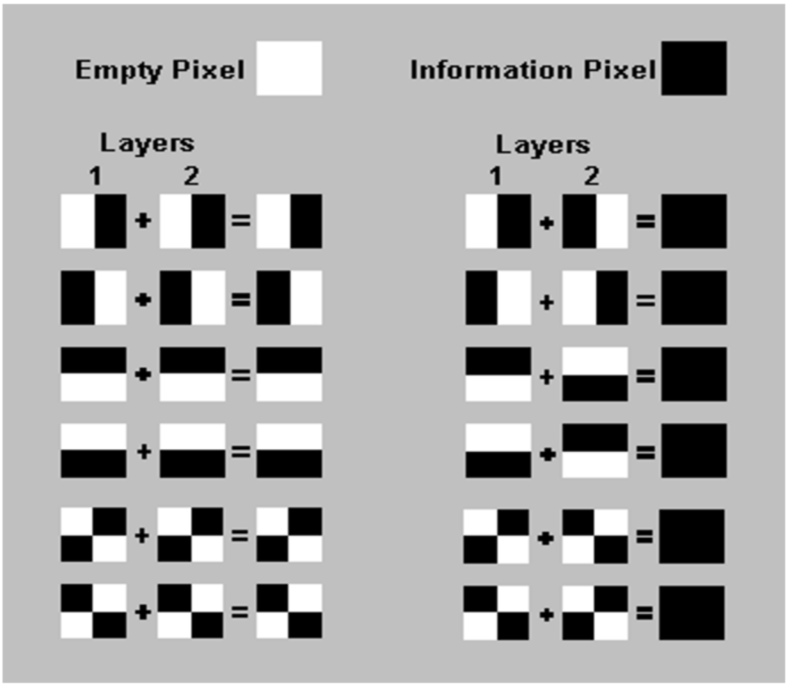


FIGURE 5. - **Explanation of expansion of pixels. [downloaded from Google]**

- The graphic above illustrates the various states that a split pixel may take. As illustrated above, six states may be obtained if the pixel is divided into four equal halves. Layer 1-pixel states may be in a certain state, and Layer 2-pixel states may not. The overlayer pixel will be split evenly between black and white if the pixel in layer 2 is the same as in layer 1.
- A completely dark will result if the pixels in layers 1 and 2 are reversed or in the other direction. [12]

4. THE PHASES OF THE MODIFIED LEAST SIGNIFICANT BIT REPLACEMENT (M - LSBR) APPROACH ARE AS FOLLOWS:

1- Verify that the cover image size is smaller than the concealed message by comparing their dimensions.

2- Split the 24-bit bitmap photo (cover pixels) into RGB components by using a bit to transform the picture's 24-bit bitmap pixels into binary bytes. One of the three components is represented by every byte. (R, G, B)

3- Using an ASCII-to-binary converter, change the secret message from binary to ASCII.

4- Assign the third pair of bits to the B component, the second pair of bits to the G component, and the final two bits of the R element of a pixel with the initial two MSBs of a secret message (M) utilizing M-LSBR.[11]

A process turns a message's binary data encoded using ASCII. Table (1) shows how the secret message "HELLO" was encoded and hidden using sixteen (16) pixels of a 24-bit bitmap picture. A 24-bit bitmap image had sixteen (16) pixels used to encode and conceal the secret message. [8]

Table 1. - Pixels with Hidden Bits

Pixels no	R	G	B
1	10010001	10010010	10010010
2	11100111	11100100	111001111
3	10001100	10001100	10001110
4	11001100	11001100	11001100
5	11110011	11110001	11110011
6	10011010	10011010	10011000
7	10000101	10000111	10000111
8	10011000	10011001	10011000
9	10000000	100000000	100000001
10	10101000	100101010	101010101
11	11001100	11001111	10001000
12	10001000	10001000	10001000

1st character is embedded

2nd character is embedded

3rd character is embedded

5. EXPLANATION OF BIT SIGNIFICANT ALGORITHM -: LEAST

Information is stored and processed in bits in computers, so the bit is theoretically more minor [14-19]. A unit that carries or transmits information or a certain meaning. In practice, in computers and digital processors, a bit is an electrical pulse that is either positive or negative (one pulse is stronger than the other). For example, a 5-volt pulse and a 1-volt pulse) and is symbolized by one of the two binary numbers, either 1 or 0. Each octet (together) is called a byte. A bit is one digit of a binary number. It has only two possibilities, the bit is 0, or it is 1. A byte is a unit commonly used to measure the storage capacity of a computer. Regardless of the type of information stored or the storage medium. 256 = 2 raised to the power of 8 bits, so a byte contains A byte, usually eight different probabilities. A byte stores values from 00000000 to 11111111 to facilitate the writing and reading of the byte A code according to a table of the letter [6].

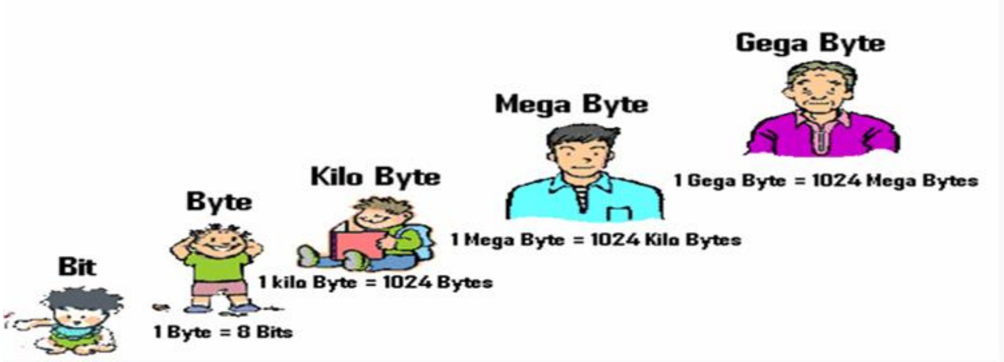


FIGURE 6. - Size of Byte [downloaded from Google]

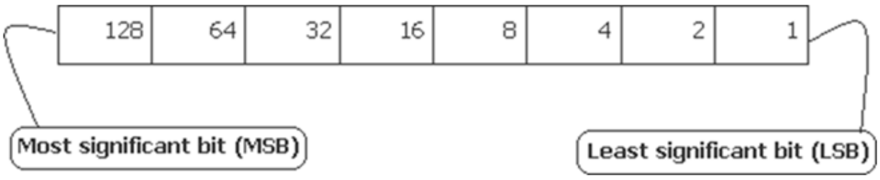
It became clear to all computer science lovers what a byte is and what a bit is, so we will not spend Most of the time talking about them; all we care about here is dividing the byte into 8 bits and knowing which bit is the lower importance, knowing that the bit only accepts 0 or 1. The distribution of the bit values in a byte is as follows:[4]

128	64	32	16	8	4	2	1

This shows that one byte represents 256 values (from 0 if all bits have a value of 0) to (255) when all bit values = 1; if we want to represent the number 65 in binary, of course, it will be:

0	**1**	**0**	**0**	**0**	**0**	**0**	**1**

Try with me to change the number on the right from 1 to 0; the number will be 64 instead of 65, and try With me; if you change the bit on the left from 0 to 1, the number becomes 193. This means that the last bit is the least essential bit (the one on the right), which, if we make zero, becomes the number 64.



I think you have now understood the intent of the bit being less important. You will answer me with certainty that if we change this value, the effect will not be clear to the extent that the addition or change can be revealed. I add that a single byte does not represent the color value of the color image by itself, but it shares two bytes with it. This is what makes us tend to color pictures in the process of much less than if he was alone. I will give you a final example of distributing one byte into 9 bytes representing three elements of an image; as we said earlier that one element is represented by 3 Bytes; note the following example: Code ASCII:(This is what it means B, which is represented by the number 66 in the system The byte that represents the code Representation of the number 66 in the following form: 01000010, meaning that we have eight bits, we will distribute each bit to Color values The values of the following bytes And let's say we have one byte of the first octet of the image, For image elements: 145, 161, 210,80,26,77,10, note the representation of these bytes:[2]

145 161 10 77 26 80 210

10010001, 10100001, 00001010,01001101 00011010,01010000,11010010

To prepare the least significant bit for addition, it must be zeroed-:

10010001 10100001 00001010 01001101 00011010 01010000 11010010

- 1 1 0 1 0 0 0

Byte values after zeroing (preparing) the least significant bit:

144 160 10 76 26 80 210

10010000 10100000 00001010 01001100 00011010 01010000 11010010

Notice the little effect we've made so far. Then we add the octal bits that represent the number 66: -

145 160 10 77 26 80 210

10010001 10100001 00001010 01001101 00011010 01010000
11010010

+ 0 1 0 0 0 1 0

145 161 10 77 26 81 210

10010001 10100010 00001010 01001101 00011010 01010001 11010010

After the substitution process, the values became: and I am sure you notice 210, 81, 26, 77, 10,161, and 145. The straightforward and inconspicuous difference in the process of hiding a letter within three image elements And if I ask, is changing one bit of each byte sufficient to hide the entire data of a text or an image? I will answer you: I have already said that BMP images are Used as a cover file for the large amount of data it collects, and I said that the included images are of JPG type or similar, if not less, in terms of Reach the size 393 x 408 The picture was to scale in the previous. The BMP image is (9) times the size of the exact JPEG measurement, while the summary page is in that bmp (KB) 25, so a small image of type. This search did not reach its size. It hides in its data a text file of more than two pages.[10]

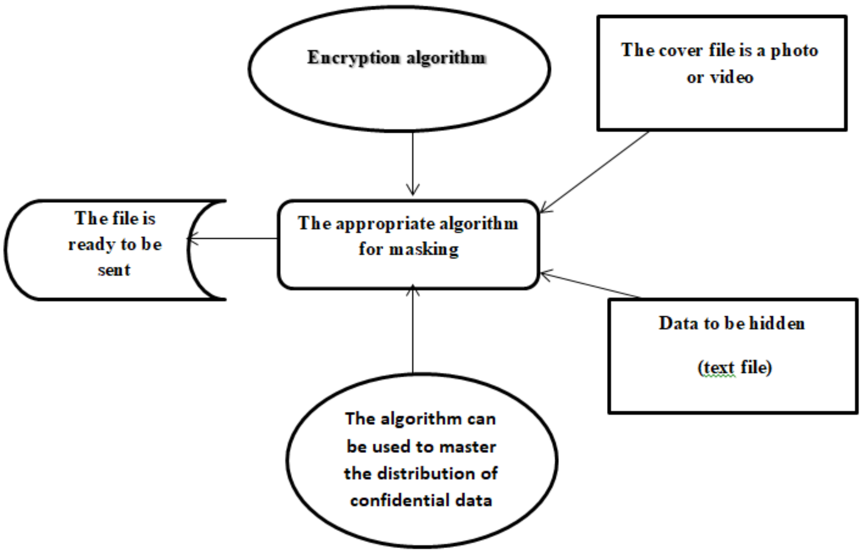

Encryption algorithm
The cover file is a photo or video
The file is ready to be sent
The appropriate algorithm for masking
Data to be hidden
(text file)
The algorithm can be used to master the distribution of confidential data

FIGURE 7. - **Scheme of encrypting and hiding the include file inside the cover file**

6. PRACTICAL APPLICATION OF THE LEAST SIGNIFICANT BIT ALGORITHM

To apply primary Program industry in Visual language We have come to the final stage of our topic .hide Book: This chapter is from the practical application of a topic, And he will compose the least significant bit algorithm, Data inside the picture: If you are a Visual Basic six programmer, you will not tire me in explaining this program, but if you are from in to be days I hope you don't mind my brief explanation, Amla its guests or beginners, The following is sufficient for your development, and then your understanding of it. We will bring the following tools to the program interface:

A . Box Text: to enter text by changing some of its properties such as:

1) Name: To avoid confusing the code with the honorable reader, we changed the tool's name to a text msg.

2) Height: Make it the right height to accept the most significant amount of data.

3) Make it true: Multi Line.

4) Vertical make it: Scroll Bars.

5) Of course, in addition to scanning the tool's content.

B . Five Command Buttons with the following designations:

1) cmd road As for the Caption, make it Load.

2) cmd Save As for the Caption, set it to Save, and the Enabled property, set it to False.

3) cmd stego Set the Caption to Stego, and set the property Enabled to False.

make it Enabled and private

C. Microsoft Common Dialog Control 6.0 (SP3) Tool

For brevity, we change its name to CmnDlg, just for brevity. True, don't forget, and again, I repeat, don't forget that.

D. Picture Box to (Auto Size)Image display tool With the change of the Makes the Pixel 3 property = Mode. Scale So our program interface will look like this:

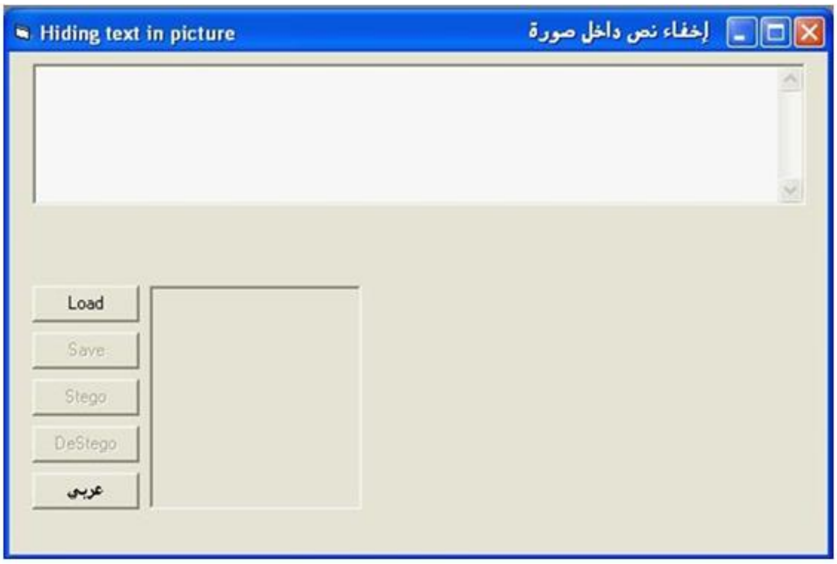
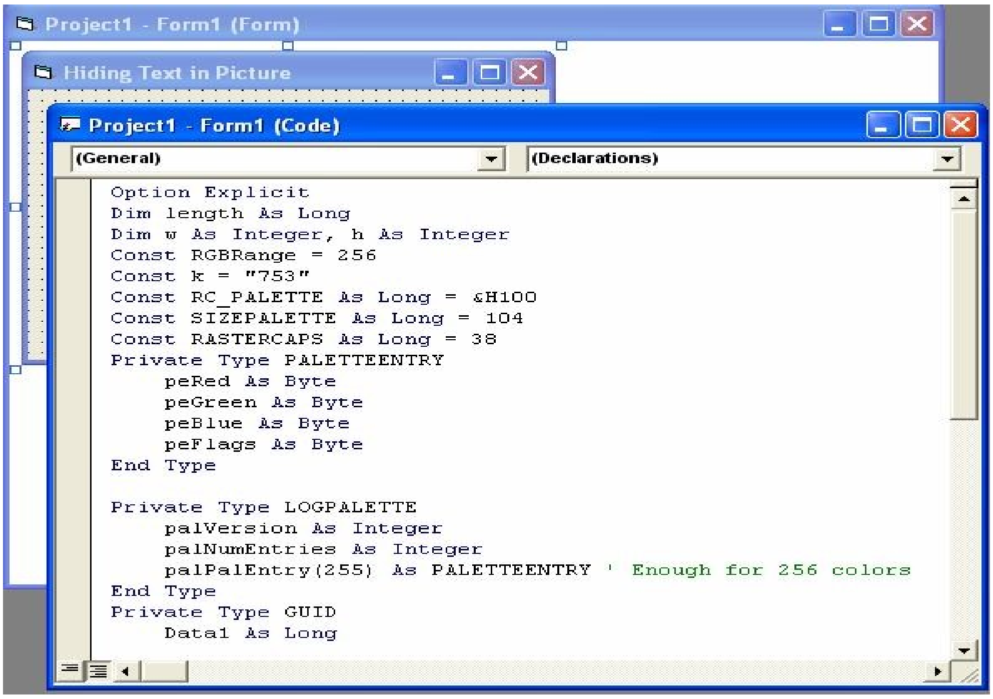


FIGURE 8. - **The program interface, the beginning of the implementation**

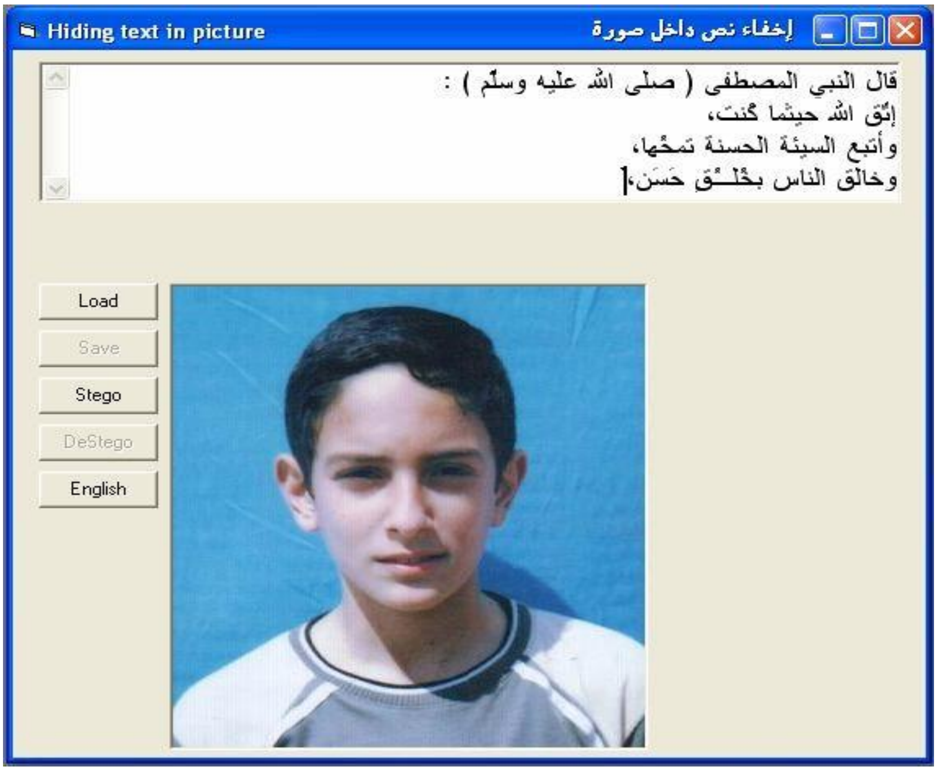
Some definitions are to be written; make one note; these definitions are specific and general. Where you write it in the field for general definitions requested, Variables used definitions for storing the image as BMP, and functions used in the process. Bytes 3 and as analysis and decoding of image elements, as well as analysis and decoding of the color value consisting of Our predecessors, to the three main components, according to the system: (RGB).



```
Option Explicit
Dim length As Long
Dim w As Integer, h As Integer
Const RGBRange = 256
Const k = "753"
Const RC_PALETTE As Long = &H100
Const SIZEPALETTE As Long = 104
Const RASTERCAPS As Long = 38
Private Type PALETTEENTRY
    peRed As Byte
    peGreen As Byte
    peBlue As Byte
    peFlags As Byte
End Type

Private Type LOGPALETTE
    palVersion As Integer
    palNumEntries As Integer
    palPalEntry(255) As PALETTEENTRY ' Enough for 256 colors
End Type
Private Type GUID
    Data1 As Long
```

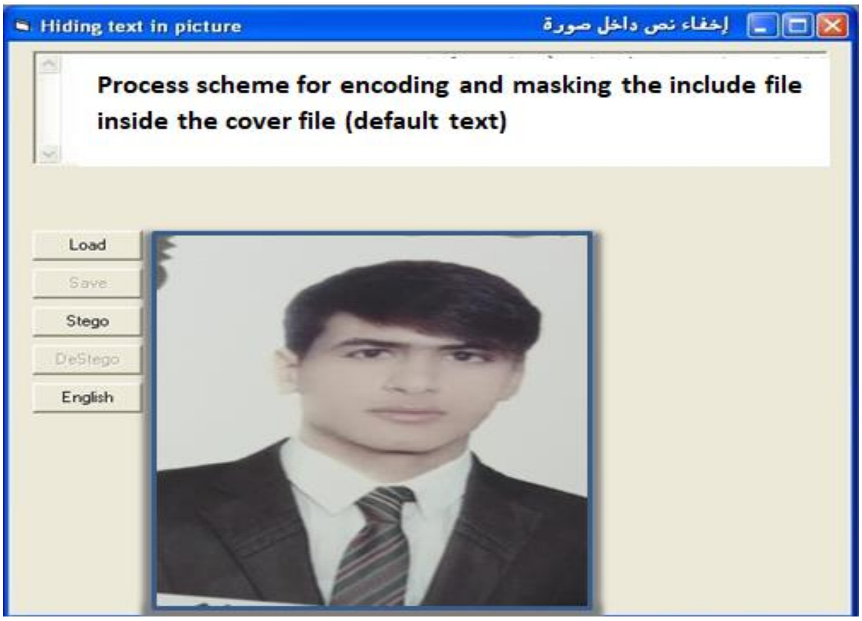
FIGURE 9. - **Writing definitions in the General field**



قال النبي المصطفى (صلى الله عليه وسلّم) :
إتّق الله حيثما كنت،
وأتبع السيئة الحسنة تمحُها،
وخالق الناس بخُلُـق حَسَن،

FIGURE 10. - Notice the active and inactive buttons and how the command button changed from Arabic to English

Save effectively; once the Stego command button is pressed, the command button becomes active. The art of data reduction and masking relies on analyzing the text into characters and converting them into the byte system, then analyzing all the elements of the image to be hidden in it into the three-byte system, and then experimenting with the least essential bit algorithm in masking Each byte of a character (text) within a single byte The three(And so on for all picture elements.) The idea of downsampling differs from the idea of encoding by not distorting the text to be sent and trying not to distort the image to be hidden in it, as far as possible from the technology, in order not to raise any chance of suspicion of the existence of hidden data inside (See Figure 10 and 11).



Process scheme for encoding and masking the include file inside the cover file (default text)

FIGURE 11. - Note the active and inactive buttons and how the command button has changed to English.

7. NEWNESS OF THE SUGGESTED PROCEDURE

This study modifies the Least Significant Bit Replacement (LSBR) steganography technique. The technique improves image quality over the conventional approach (used by the majority of earlier methods); this, as seen in Figure 10 and 11, replaces the tiniest bits of only one of all three colors—Red, Green, and Blue—of each of the pixels in a 24-bit bitmap picture. This changes the tiniest bits of all three colors in each pixel.

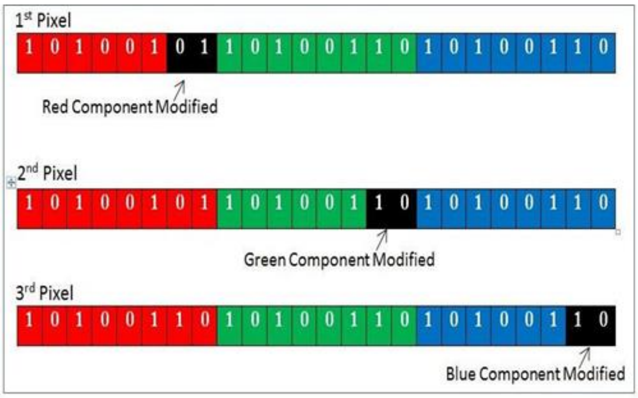


FIGURE 12. - LSBR modification (Proposed Method) [12]

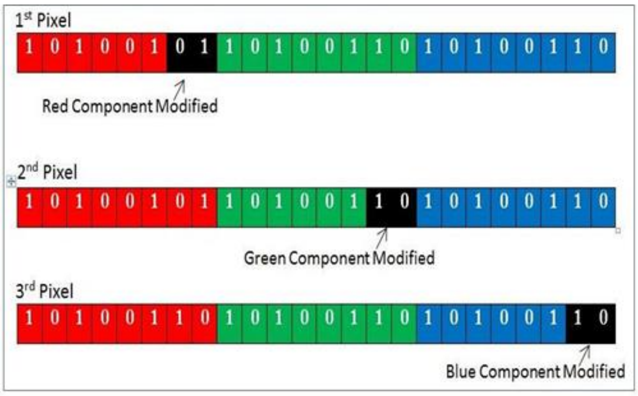


FIGURE 13. - Conventional LSBR Method [12]

Findings and Conclusion To determine if the suggested approach (M-LBSR) covered the recognized gap found in the earlier works, this study intends to compare the outcomes of some of the previous studies with those of the suggested approach. The standard least significant bit (LSB) replacement strategy, used by Champakamala, Padmini, and Radhika (2009), is characterized by a low PSNR value (9.9987 dB only) and poor image quality, as seen in Figure 6. This is following the literature review section. The approach was limited to messages with a maximum size of 15 bytes. Comparatively, the method proposed by this research exhibits a pi in its initial experiment. As seen in Figure 11, when studied in a Matlab environment, the strategy suggested by this research yields a better PSNR value (51.8283 dB). In its initial experiment, it also effectively conceals a secret message of 52 bytes, with no discernible change between the look of the two pictures. (before and after hiding). The lecture hides a 52-byte-long secret message with no observable visual differences.

Table 2. - summary of results (previous research) [11]

s/n	literature	Technique used	Psnr (in dB)	Image quality yy	Message capacity(bytes)
1	Abdul-sada(2017)	LSB replacement	39.087	low	45
2	Ali and saad	Matching method	40.113	high	20
3	Eman al (2016)et	Random substitution	41.48	low	51

4	Korothan et al(2016)	Noisy substitution	41.131	low	43
5	Champakama la et al(2009)	Technique used conventional LSB	19.9987	low	15

8. CONCLUSION

The suggested approach for hiding the secret message was compared to the LSB benchmarking method, in which the secret message is directly concealed in at least two significant bits of the image pixels. Using the recommended LSB hiding methods, five (5) different Bmp photos were employed to hide hidden messages of various sizes. Peak Signal Noise Ratio measurements were used to analyze and appraise the outcomes of the suggested LSB hiding strategies. The proposed approach is determined to be more effective, simple, suitable, and accurate than the approaches investigated since it embeds consistently and prevents overusing one specific component while disregarding the other. As a result, when data is buried, each element offers the same level of intensity, modifying the overall impression. The LSB benchmarking approach, which directly conceals the secret message in the least two significant bits, was contrasted with the suggested method. Due to the equal contributions of each component to intensity and data hiding, there is little change in picture resolution; this increases the security of the concealed information and alters the picture. The LSB benchmarking approach, which directly conceals including at least two crucial pieces of the secret message, was contrasted with a suggested method. The suggested method is found to be more efficient, straightforward, appropriate, and accurate than the techniques looked into since it embeds consistently and prevents overusing one particular component while ignoring the other. Since each element contributes equally to intensity while obscuring data, there is little change in visual quality; this increases the security of the hidden information.

9. FUTURE WORK AND SUGGESTIONS

Steganography should be seen as a supplement to encryption, not as a substitute for it. It will be a good idea to investigate how to apply the combined methodologies in future studies. Information security should be improved and made more effective with the use of the Crypto-Stego technique and the M-LSBR model, according to expectations. As a result, studies should explore going beyond steganography. Other types of cover files besides images can also be used in future studies to use the M-LSBR approach. It is also possible to utilize an audio, video, or text file as cover material. Increasing the suggested method's capacity while maintaining or increasing PSNR results in greater quality against incursion.

FUNDING

No funding received for this work

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their efforts.

CONFLICTS OF INTEREST

The authors declare no conflict of interest

REFERENCES

[1] J. Korothan, S. Kishor, and P. Butey, “De-Noise Steganography by Enhancing the Cover Image: A Multi-Level Security Approach,” International Arab Journal of Information Technology, vol. 13, no. 6A, pp.851-857, 2016.

[2] S. A. Laskar and K. Hemachandran, “Steganography Based on Random Pixel Selection for Efficient Data Hiding,” International Journal of Computer Engineering and Technology, vol.4, no.2, pp.31-44, 2013.

[3] T. Manikandan, A. Muruganandham, R. Babuji, V. Nandalal, and J. M. Iqbal, "Secure E-Health using Images Steganography," Journal of Physics: Conference Series, vol.1917, no. 012016, pp.1-8,2021.

[4] P. Mortazavian, M. Jahangiri, and E. Fatemizadeh, "Low degradation steganography model for data hiding in medical images," International Journal of Computer Science and Information Technology, vol.4, no.2, pp.234-240, 2016.

[5] S. C. Salluri, "Image Steganography and Sending Private Data through Email Using Cloud Computing," International Journal of Progressive Research in Science and Engineering, vol.1, no.8, pp.92-94, 2020.

[6] A. Devi, and K. B. ShivaKumar, "A Novel Image Steganography Technique for Secured Online Transaction Using DWT and Visual Cryptography," IOP Conference Series: Materials Science and Engineering, vol.255, no.012070, pp.1-9, 2017.

[7] M. Goel and N. Jain, "A RSA- DWT Based Visual Cryptographic Steganogrphy Technique," International Journal of Advanced Research in Computer Science and Electronics Engineering, vol.1, no.2, pp.40-45, 2012.

[8] R. Gupta, A. Jain, and G. Singh, "Combine use of Steganography and Visual Cryptography for Secured Data hiding in Computer Forensics," International Journal of Computer Science and Information Technologies, vol.3, no.3, pp.4366 - 4370, 2012.

[9] M. Juneja, P. S. Sandhu, and E. Walia, "Application of LSB based steganographic technique for 8-bit color images," International Journal of Computer and Information Engineering, vol.3, no.2, pp.297-299, 2009.

[10] Champakamala, Padmini, and Radhika (2009) are characterized by a low PSNR value (9.9987 dB only) and poor image quality.

[11] M. Usman and B. Yusuf, "A Modified Least Significant Bit Replacement: Steganography Method for Handling Random Noise Effect," IPS Journal of Physical Sciences, vol.1, no.1, pp.1-5, 2022.

[12] A. A. Ali, and A. S. Saad, "New Image Steganography Method By Matching Secret Message with Pixels of Cover Image (SMM)," International Journal of Computer Science Engineering and Information Technology Research, vol.3, no.2, pp.1-10, 2013.

[13] A. S. Ansari, M. S. Mohammadi, and M. T. Parvez, "A Multiple-format steganography algorithm for color images," IEEE Access, vol.8, no.3, pp. 83926- 83939, 2020.

[14] M. M. Mijwil, M. Aljanabi, and ChatGPT, "Towards Artificial Intelligence-Based Cybersecurity: The Practices and ChatGPT Generated Ways to Combat Cybercrime," Iraqi Journal For Computer Science and Mathematics, vol.4, no.1, pp.65-70, 2023.

[15] M. M. Mijwil, E. Sadıkoğlu, E. Cengiz, and H. Candan, "Siber Güvenlikte Yapay Zekanın Rolü ve Önemi: Bir Derleme," Veri Bilimi, vol.5, no.2 pp.97-105, 2022.

[16] M. M. Mijwil, O. J. Unogwu, Y. Filali Y., I. Bala, and H. Al-Shahwani, "Exploring the Top Five Evolving Threats in Cybersecurity: An In-Depth Overview," Mesopotamian journal of cybersecurity, vol.2023, pp.57-63, 2023.

[17] M. M. Mijwil, M. Aljanabi, and A. H. Ali, "ChatGPT: Exploring the Role of Cybersecurity in the Protection of Medical Information," Mesopotamian journal of cybersecurity, vol.2023, pp.18-21, 2023.

[18] M. M. Mijwil, I. E. Salem, and M. M. Ismaeel, "The Significance of Machine Learning and Deep Learning Techniques in Cybersecurity: A Comprehensive Review," Iraqi Journal For Computer Science and Mathematics, vol.4 no.1, pp.87-101, 2023.

[19] M. M. Mijwil, R. Doshi, K. K. Hiran, AH. Al-Mistarehi, and M. Gök, "Cybersecurity Challenges in Smart Cities: An Overview and Future Prospects," Mesopotamian journal of cybersecurity, vol.2022, pp.1-4, 2022.