Periodicals of Engineering and Natural Sciences ISSN 2303-4521
Vol. 8, No. 2, June 2020, pp.1000-1010

Improved technique for hiding data in a colored and a monochrome image

Baydaa Jaffer AlKhafaji[1], May A. Salih[2], Shaymaa AbdulHussein Shnain[3], Zahraa Modher Nabat[4]

[1]University of Baghdad

[2,3,4] Babylon University

ABSTRACT

Find a new way to hide the different types of confidential and important text files inside the images without noticing any change or distortion of the information in the images after the process of concealment or attempt to be detected by hackers. And then extract these texts and retrieved on demand without loss or loss or distortion of any of its content. The algorithm was used to hide the different text in different image formats using (256) elements, monochrome and color where about 4500 characters were hidden in a monochrome image and about 900 characters in a color image and with a hiding distance (S) where N = 3, key = 3. The overall error rate is low (0.05-0.17) and no distortions were observed on the resulting image.

Keywords: hidden text, monochrome image, color image, Encryption image

Corresponding Author:

Baydaa Jaffer AlKhafaji
Computer Science Department, College of Education for Pure Science/Ibn Al- Haitham, University of Baghdad, Iraq
bjkh68@yahoo.com

1. Introduction

Become use of computers in this time the most important means and widespread in of storing [1]. retrieving and circulating information on the local, international, e-mail and mobile phones through digital media such as text, audio, images and animation. It has become easy to intercept information sent across different Connection networks or access to those computers, whether independent or linked with the network, with intent See their content or steal or tamper with important information. In this light, protection. reliability, and information transmitted through these files, but looks like ordinary files where the general shape of the file is maintained [2]. the is Image Hiding, credibility of information must be secured and maintained. Various means of protection such as passwords, Cryptography, hiding techniques or information coverage have emerged. Encryption removes confidential messages and data into a non-readable form using the secret key and is retrieved using that key. The concealment is done by placing the data in the media files so that it cannot be observed or detected or the existence of which includes hiding as much of the important information (document, message, charts or images) in text files or images in a manner that does not arouse curiosity, but looks like images of a declaration or ordinary texts [3]. This encoding uses the numbers from 0 to 255, where each number is one byte and the encoding in this format facilitates the transfer of texts between the computers and the devices. Attached, being a standardized standard globally. The image consists of a matrix of elements (Pixels), each of which represents each of its elements according to the type of image, Binary Image consists of two colors, each element represents one house value either (0) for black or (1). The other type is monochrome, in which the image element represents a number of bits, through which the number of color gradients can be defined in this picture. For example, if the element is represented by three bits, this indicates that the image is at the most eight of the gradients. If, for example, eight bits (one pat), the image has at most 256 chromatic gradients, since the value (0) is not a color and the value (255) is the highest value for that color. The color image is

represented by three packets of monochrome image data, each package representing a different color from the remaining packages and most of the color images represented by the three blue, red, green, and each element is represented by three bits (24bits) Each of which carries the value of a color of the basic colors which can be obtained (16777216) different color, and the value of any chromatic gradient in most computer applications by the equation

Color = (Red *65536)+(Blue*256)+Green (1)

2. Difference between steganography and cryptography

Cryptography is the of hiding information, while Steganography deals with composing hidden messages so that only the receiver and the sender know that the message even exists. In Steganography only the receiver and the sender know the continuation of the message, cryptography Modern focuses on developing cryptographic where information needs to be protected from other third parties [4]. This is done through algorithms that are difficult to penetrate by the opponent due to computational rigidity and therefore cannot be easily broken, Thus the presence of the encrypted message is visible to the world. And for this Steganography removes excess attention that reaches the hidden message, Encryption methods attempt to defend message content capping three types of encryption algorithms used called hash functions, public key encryption, and symmetric key encryption. while Steganography uses methods that hide the message as well as content. by combining encryption and information concealment cryptography includes encryption methods where both the receiver and the sender share the same key used to encrypt the data. In Public-key cryptography, two different but mathematically related keys are used. Hide information is to create a hidden message so that only the recipient and sender know that the message exists hide information is used in ancient times and these methods are called the science of hiding information Examples of these methods are to hide texts in message these methods of hiding recent information and ensuring that there are random data inside the hidden message, annoying images inside the hidden message and merge the message with images inside video files.

3. The proposed system, and the algorithms used

The masking process is generally done by converting the hidden file into a series of bits hidden inside the image element bytes by substituting the number of bytes of the image element with the same number of bits of characters to be hidden from the Least Significant Bits (LSB). The manipulation of less important bit values does not significantly affect the value of color, and this slight effect cannot be observed because the human eye's ability to distinguish between convergent chromatic gradations is weak, as is its ability to distinguish between dark gradations [5]. In some previous concealment methods, binary text is hidden in the binary image in the monochromatic image, bits are hidden at most in the bits of the image element bytes. In this case, the effect of concealment on the image is very small [6]. and in other methods hide the bits within the elements of the image in different places and it requires the work of a map or table of distribution sites, and then requires the existence of the map or table or original image to extract bits hidden file, the hacker has thus increased the likelihood of text discovery. Another method is to convert text characters to a series of bits and then hide every 4 bits in the bytes of the image element, in this way you can hide a larger number of text characters [7]. By hiding file bits within the image elements sequentially (element after element) increases the likelihood of text discovery in image analysis methods or statistical processes, it is suggested that the following method be used. The proposed dispersion masking algorithm: The proposed method of masking depends on monochrome images or color images. When the monochrome image is selected, the bytes of each text are divided into two parts, each containing four bits. Each part of each image is hidden in one of the elements of the image. This requires two elements of the image to hide one character. In the color image, the character is divided into three parts, one containing two bits, the other two containing three bits, each part is hidden in bytes of each color of the three colors representing the color image element, and reference to the human vision system (HVS: Human Visualize System), which indicates that the human eye is more sensitive to the blue color than the other two colors (red and green) [8-10]. The formula for hiding the two colors in blue and other parts of the three other green and g-, R), and so one element of the color image is required to hide one character. To increase the accuracy of concealment and reduce the probability of detecting the text within the image, because the chance of detecting the hidden concealment is greater, the process of hidden hiding by hiding the parts of the text characters within the bytes of the elements of the image in a non-sequential way, the distance between them is irregular, and determine this distance (S) The value of the number of bits of the resulting image (N) after the current masking process and adding a value concerned as a key, this distance will be called the distance of the hash [11,12]. This distance will be added to the location of the current element (in the image matrix) Where the concealment process will take place [13,14]. For example, if the position of the current element E is (5, 25) and the value of the byte after the concealment is 10 (205) = (5, 25) E and the number of bits selected to add its value is 3 bits and the key value equals (7). The following is as follows:

$S=(N)_2+(Key)_{10}$ (2)

$(205)_{10} = (1100\ \underline{1101})_2$

$S= (\underline{101})_2 +(7)_{10} = (5+7)_{10} = (12)_{10}$

E_n= E (25, 5 + S) = E (25,17)

Adding a key is a safe way to avoid being hidden in the same location when the value of the selected bits is zero. The following two examples illustrate how to hide a character in a monochrome image element.

1. (K) in a single image element E1 where the position of the first element is (7, 2) and the color value of the element is (123) and the key value (10) and the number of bits selected to add value is (4) bits.

a. Convert the character value to binary.

$K = (107)_{10} = (0110\ 1011)_2$

b. Divide the byte of the character into two parts by performing an operation (AND) between the value of the item's bytes and the value

$(00001111)_2$ for the first part (P1) and then the process (AND) between the element byte value and the value $(11110000)_2$ and skew the output four steps to the right to obtain the second part (P2).

$P1= (1011)_2$

$P2= (0110)_2$

c. To convert the color value of the element to the binary.

$E1(2,7) = (132)_{10}= (1000\ 0100)_2$

d. Hide the first part in (LSBS) of the byte image element $E_{1new} = (10001011)_2= (139)_{10}$

e. Calculate the distance of concealment.

$S= (1011)_2+(10)_{10}=21$

f. Calculate the address of the subsequent element by adding the value of (S) to the value of the coordinate.

E_n (2,7+21) = (2,28)

g. Convert the color value of the subsequent element to the binary (assuming that the value is equal to (145))

$En\ (2,28) = (145)_{10} = (1001\ 001)_2$

h. Hide the second part in LSBS of the image element bytes.

$E_{n\ new\ =} (10010110)_2= (150)_{10}$

2. hidde (K) in the color image element E1 and the location of the first element is (2.7,) with the value of green (G = 210), red (R = 200), blue (B = 180) and Key value 9 The number of bits selected is (4) bits

a. Converts character value to binary character

$K= (107)_{10}= (0110\ 1011)_2$

b. Divide the character byte into three parts by doing an AND operation between the element byte value and the value $(0011\ 0000)_2$ for the first part and then the AND operation between the item byte value and the value $(0001\ 1100)_2$ and the output of the product is rotated to the right to obtain the second part. Then the process (AND) between the element byte value and the value of $(1110\ 0000)_2$ and the resulting five output scaling to the right to obtain the third part.

$P1= (11)_2$

$P2= (010)_2$

$P3= (011)_2$

c. To convert the color value of the element to the binary.

$R= (200)_{10}= (1100\ 1000)_2$

$G= (210)_{10}= (1101\ 0010)_2$

$B= (186)_{10}= (1011\ 1010)_2$

d. Hide the first part in (LSBS) of the green byte color.

$G_{naw}= (11010010)_2 \quad = (210)_{10}$

F- Hide the third part in LSBS () from the blue color byte.

$B_{new}= (1011\ 1011)_2 = (187)_{10}$

e. Hide the first second part in (LSBS) by the red color bytes.

$R_{new} = (1100\ 1011)_2 \quad = (203)_{10}$

f. Hide the second part in (LSBS) of the green byte color.

$G_{naw}= (11010010)_2 \quad = (210)_{10}$

g. Hide the third part in LSBS () from the blue color byte.

$B_{new}= (1011\ 1011)_2 = (187)_{10}$

h. Calculate the distance of concealment by taking (4) bits of green or any other color.

$S= (0010)_2+(9)_{10}=11$

i. Calculates the title of the subsequent item.

$E_n= (2, 7+11) = (2,18)$

It is necessary to estimate the size of the appropriate picture to hide the entire text, and can be calculated using the following equations, which were derived by the statistics conducted on a variety of images after the application of the process of concealment scatter them as follows:

Color Size = Number of characters * 3/4 Highest distance to hide (3)

Monochrome image size =Number of characters* 2* 3/4 The highest distance to hide

In the practical side of the search, we will display two image of the color image and a monochrome image according to the following algorithm.

Algorithm:

1- input: image and text.

2- output: steganography image.

3- process.

Step1: convert text to acull code for each character.

Step 2: convert each ascll code to binary number.

Step 3: extract RGB from each pixel.

Step 4: convert R to binary 8bits and convert G to binary 8 bits.

Step 5: take first bit frome text and put in the LSB of R then take second bit from text and put in the LBS of G and so on.

The masking algorithm can be summarized in monochrome images as in the following steps

1. Display the text file to be hidden by one application to display texts.
2. Add a special code at the end of the text, for the purpose of stopping when it appears in the process of decoding.
3. Choose the appropriate image size for the masking process.
4. Set the location of the image element that will hide the first letter of the text.
5. reading the letter from the text and find the corresponding ASCII format in bytes and then divide the byte into two parts each part consists of (4) bits.
6. Read the color value of the image element bytes.
7. Substituting the bit bits of the image element in the location of the less important bits in the bits of the first part to (Muhammad, *et al.*, 2015). configure the value of the new byte of the element and then convert it to the color value. Then locate the next element that will cover the masking process, by calculating the hiding distance and adding it to the location of the current element.
8. Repeat steps (7,6) to store the second part of the character bytes.
9. Refer to step (5) until the end of the characters of the text.
10. Store the image for later use or send it over the networks or by e-mail.

The retrieval of the hidden text within the monochrome image is done by the following steps:

1. Display a blank page for an application to display text.
2. Display the image that contains the text.
3. Locate the element of the image that holds the first part of the first letter of the hidden text, according

to the key agrkeed between the sender and the receiver.

4. Read the color value of the image element and converted to the byte formula and take the four bits replaced and stored in a specific location (where two sites are allocated in each location is stored one part of the character).

5. Select the next image element by calculating the masking distance and adding it to the location of the current element.

6. Repeat steps (5.4) to complete the storage of parts of the character and determine the picture to the right.

7. Compile the characters to form the character byte and then convert its value to ASCII and then to the shape of the character and place it in the text editor page.

8. Repeat the steps (7,6,5,4) until the symbol for the end of the text appears.

9. Store the text into the calculator for later use.

The algorithm of concealment in the color image is summarized in the following steps:

1. Display the text file to be hidden by one of the applications to display texts.

2. add a special code at the end of the text, to take advantage of when retrieving the concealment.

3. Choose the appropriate image size for the masking process.

4. Set the location of the image element to hide the first letter of text.

5. read the letter from the text and find the ASCII formula corresponding to him in bytes, and then divide the byte into three parts of the first part.

6. contains (2) the first two parts, but the second and third parts each contains (3) bits in succession.

7. Read the color value of the image element by extracting the values of the three bytes corresponding to the basic colors (red, green, blue) (R, G, B).

8. Replace bits of bytes of each color with three bits of one bit of the character in the least important bit position to create the value of the new three-byte bytes of the element.

9. Locate the element to the right that will be the process of concealment by calculating the distance of the hash (Scrap) and added to the location of the current element.

10. Repeat steps (8,7,6,5) until the end of the characters of the text.

11. Store the image for later use or send it by e-mail.

12.

The hidden text retrieval algorithm within the color image is as follows Display a blank page for an application to display text.

1. Display the image that contains the text.

2. Locate the image element with the first letter of the hidden text.

3. Reading the color value of the image element and extracting the values of the three (Lai, *et al.*, 2011). bytes representing the red, green and blue colors, then the replaced bits are taken in the masking process of each color and are grouped to be the value of the character bytes.

4. Convert the value of the character of the ASCII formula to the form it represents and placed on the text editor page.

5. Determine the location of the element to the right of the concealment process by calculating the distance and adding it to the current element location.

6. Repeat the steps (6,5,4) until the symbol for the end of the text appears.

7. Store the image for later use or send it over the networks or by e-mail.

To determine the resulting image quality after the masking process, (Root Mean Square error) equation is calculated between the original image and the resultant image as follows:

$$e_{RMS}s=\sqrt{\frac{1}{n^2}}\sum_{r=0}^{n-1}\sum_{c=0}^{n-1}[\,\bar{I}(r, c\,)-I(r, c)]^2 \quad (4)$$

whereas:

e_{RMS}: The average root of the error box for the image.

N: After the picture (by pixel).

(r, c) $\bar{I}$: represents the image element after the mask.

(r, c) I: represents the image before hiding.

In this example we opened the interface of the program and then we uploaded a color image for the purpose as shown below. in (Figure 1, 2,3,4,5,6 and7).

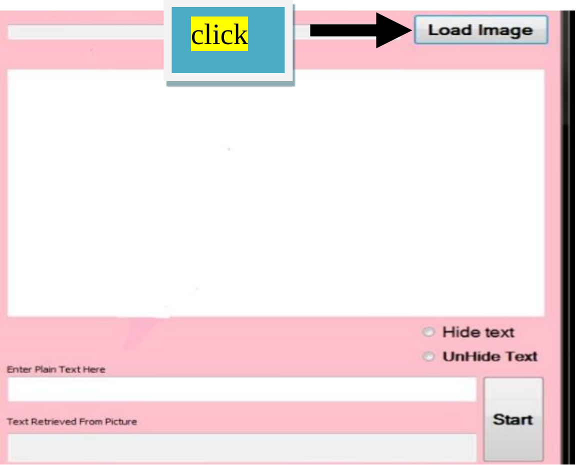


Figure 1. Open the interface to select the image

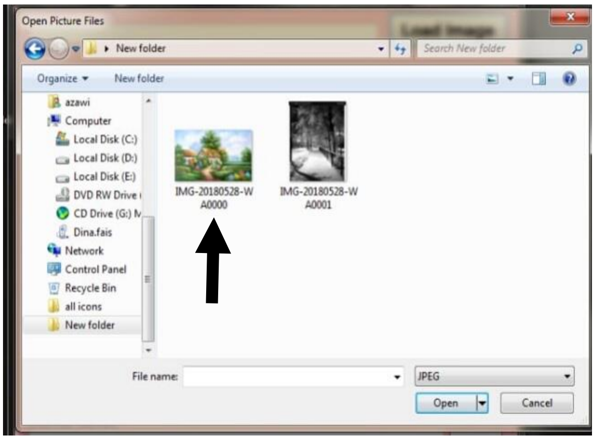
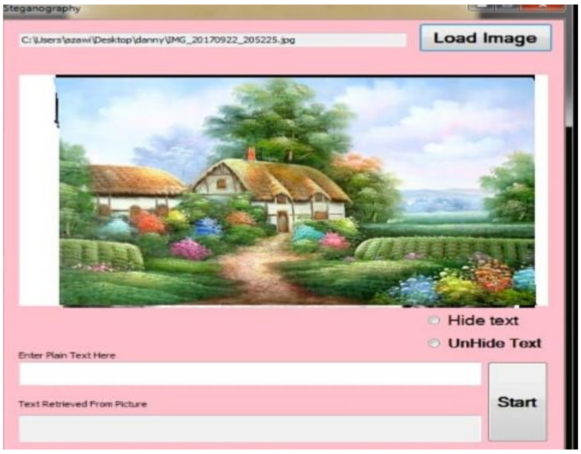
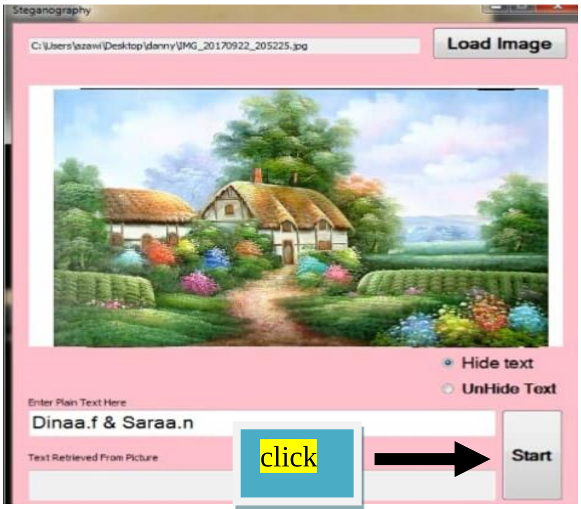


Figure 2. Open any file that contains image



C:\Users\azawi\Desktop\danny\IMG_20170922_205225.jpg

Figure 3. selected image

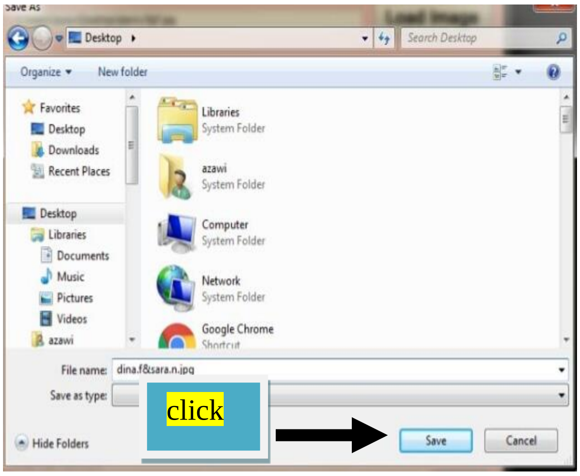


C:\Users\azawi\Desktop\danny\IMG_20170922_205225.jpg

Enter Plain Text Here: Dinaa.f & Saraa.n

Text Retrieved From Picture:

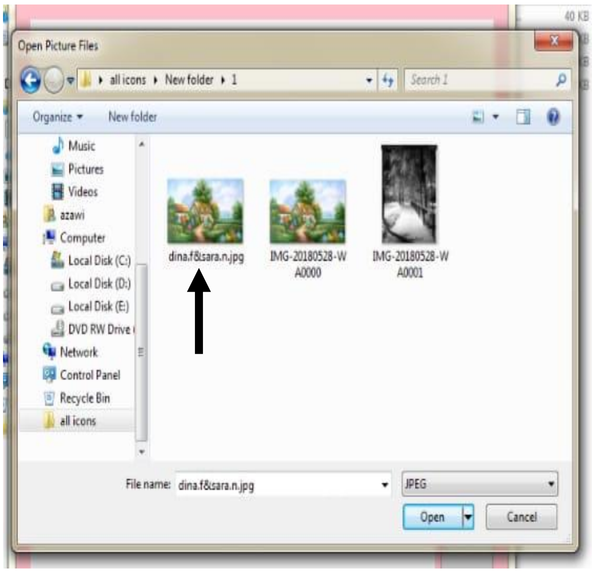
Figure 4. Insert text and press hide the text in the image



File name: dina.f&sara.n.jpg

Save as type:

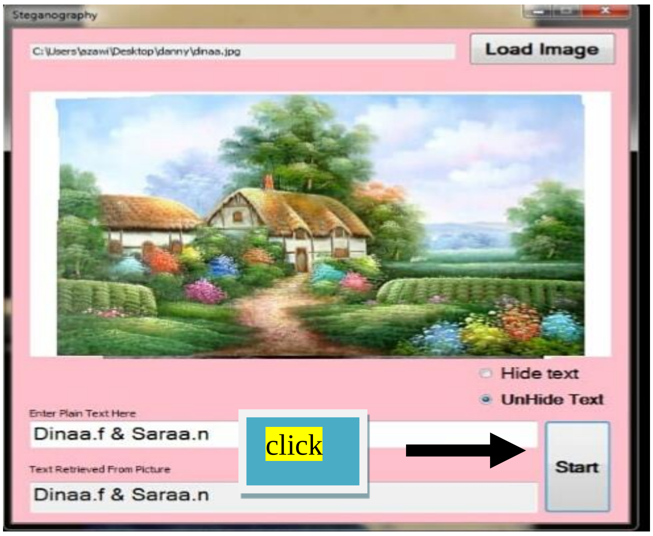
Figure 5. Save the new image with its written image with the jpg extension



File name: dina.f&sara.n.jpg

JPEG

Figure 6. We will select the new image in which we have hidden text



C:\Users\azawi\Desktop\danny\dinaa.jpg

Enter Plain Text Here: Dinaa.f & Saraa.n

Text Retrieved From Picture: Dinaa.f & Saraa.n

Figure 7. select the new image and then click on unhide to simulate hidden text inside the picture

In this example we opened the interface of the program and then we uploaded monochrome image for the purpose of hiding text inside in: (Figure 8,9,10and11).

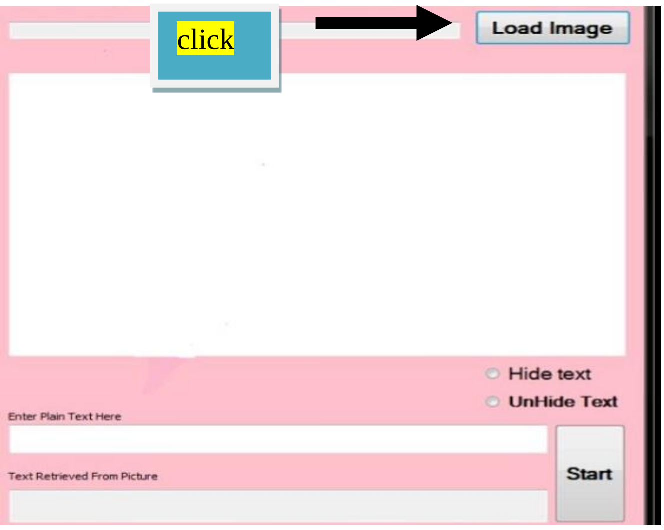


Figure 8. Open the interface to select the image

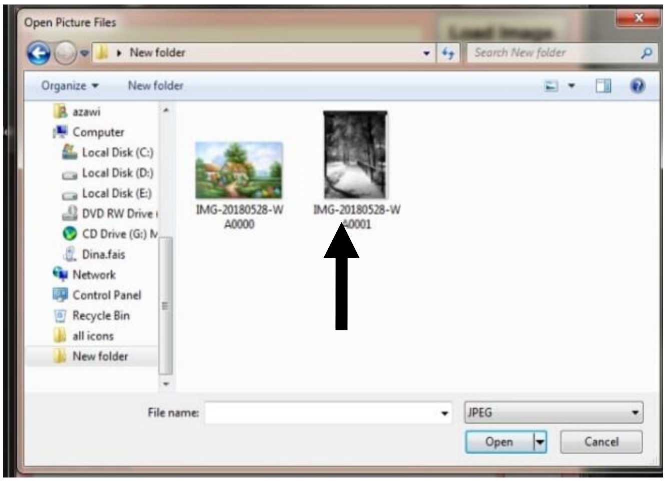
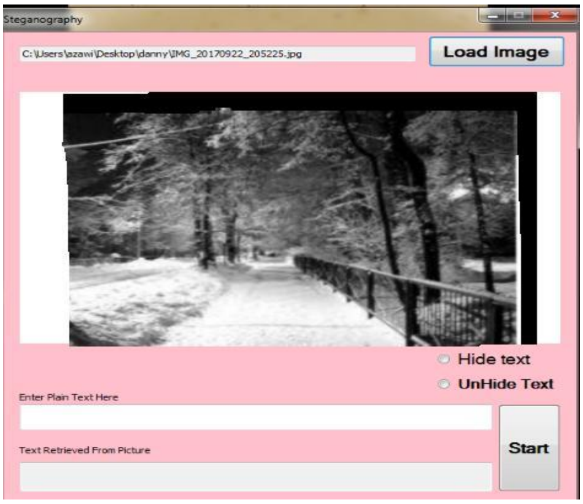
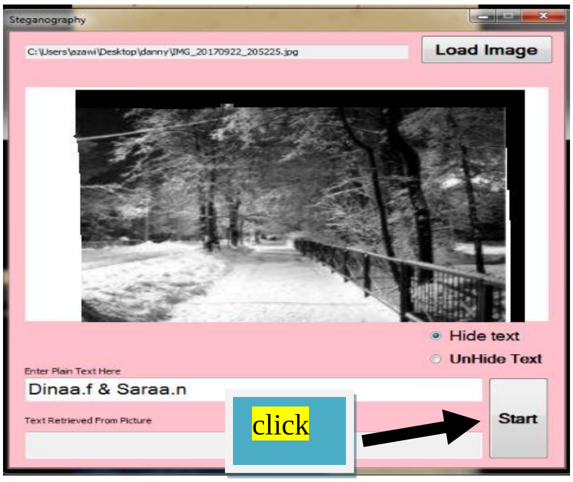


Figure 9. Open any file that contains images then we choose a picture



C:\Users\azawi\Desktop\danny\IMG_20170922_205225.jpg

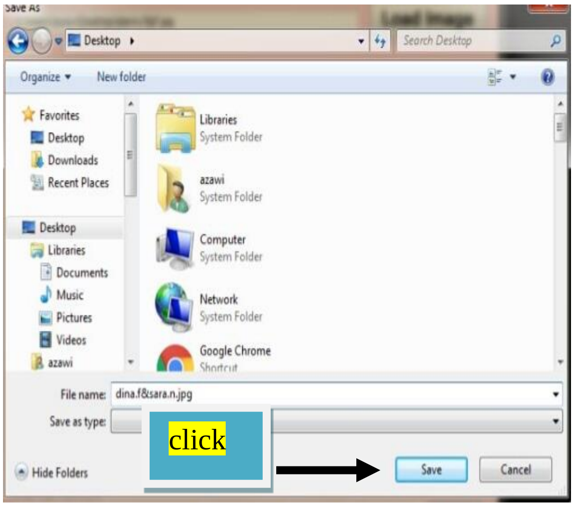
Figure 10. Selected image



C:\Users\azawi\Desktop\danny\IMG_20170922_205225.jpg

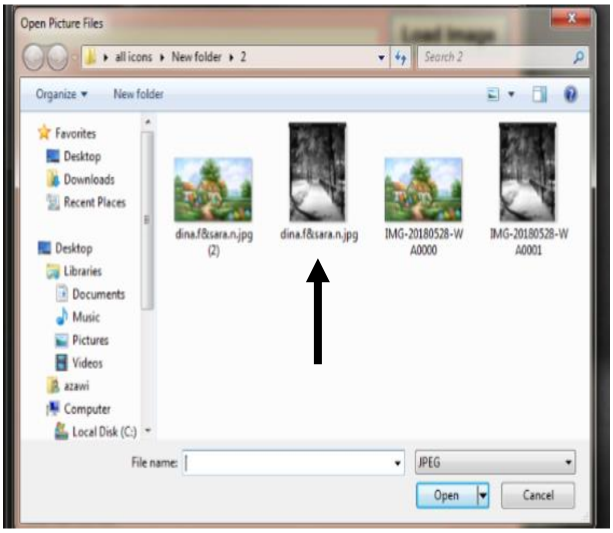
Enter Plain Text Here: Dinaa.f & Saraa.n

Figure 11. Insert text and press hide text to hide the text in the image



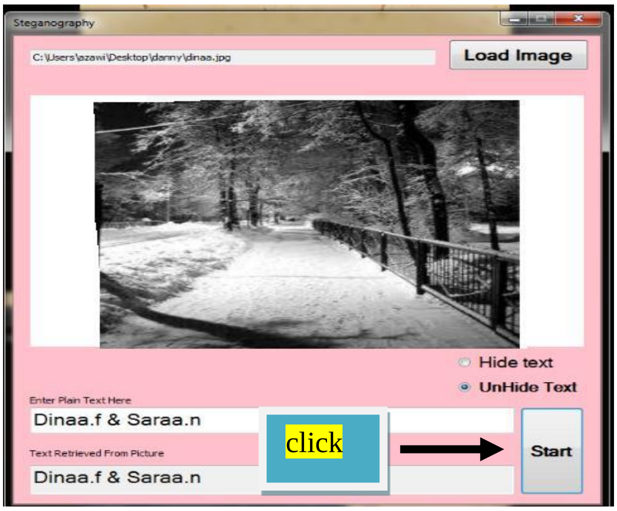
File name: dina.f&sara.n.jpg

Figure 12. Save the new image with its written image extension



JPEG

Figure 13. will select the new image in which we have hidden text



C:\Users\azawi\Desktop\danny\dinaa.jpg

○ Hide text

● UnHide Text

Enter Plain Text Here: Dinaa.f & Saraa.n

Text Retrieved From Picture: Dinaa.f & Saraa.n

Figure 14. select the new image and then click on unhide to simulate hidden text inside the picture

4. Results and discussion

The cryptographic masking algorithm was applied to hide various texts in different image formats with 256 (256) (elements, monochrome and color, and calculate the ratio of errors between the original image and the resulting image after the masking process using Equation (5) (Visual Basic) was prepared for this purpose, hiding approximately 4500 characters within a monochrome image and approximately 900 characters in a color image and with a concealment distance (S) where N = 3, Key = 3.The overall error rate was low (0.05-0.17) and no distortions were observed on the resulting image for both types of images (Figure 1 and 2

5. Conclusions

The method used has proved successful in the process of hiding various types of texts in monochrome and colored images. Hash distance between the elements of the image reduces the probability of detecting hidden text because the distribution depends on the (secret key) [13,14]. is agreed upon, as well as the distance of the skewing is not fixed, but methods that use sequential sequencing and at a steady pace, they are more vulnerable to discovery and Suspicion of the thief or hacker. The use of the value of the (key) with the value of part of the resulting image after the concealment can control the distance of the hash and thus (tradeoff) balance between the size of the text to hide and the size of the image cover. The percentage of concealment in this method is less compared to the traditional methods of the existence of space left without hiding because of the adoption of the mechanism of skimming in the process. It is preferable to use images with many details (ie high-texture image) in the masking process. Any process of compressing or improving the image bearing the hidden text or change the extension will lead to the loss of all or part of the hidden text and cannot be retrieved in full. To increase the efficiency of the concealment in the color image, it is possible to distribute the concealment of the three parts on the three colors, the amount of skimming for each color is calculated separately and so each part of the character to hide in a different element of the image as this reduces the possibility of detection.

References

[1] Behera, S.K. (2013). E-and M- Learning: A comparative study. International Journal on New Trends in Education and Their Implications 4(3): 65-78.

[2] Lupton, E., & Phillips, J.C. (2015). Graphic Design: The New Basics: Revised and Expanded. Chronicle Books.

[3] Burger, W., & Burge, M.J. (2016). Digital image processing: an algorithmic introduction using Java. Springer.

[4] AL-Khafaji, B.J. (2015). Proposed System Mix Cryptography and Steganography to Hide Information, 4 (3): 663 – 674.

[5] Amirtharajan, R., Qin, J., & Rayappan, J.B.B. (2012). Random image steganography and steganalysis: Present status and future directions. Inform. Technol. J, 11: 566-576.

[6] Cheddad, A., Condell, J., Curran, K., & Mc Kevitt, P. (2010). Digital image steganography: Survey and analysis of current methods. Signal processing 90(3): 727-752.

[7] Yahya, A. (2019). Steganography Techniques. In Steganography Techniques for Digital Images pp. 9-42.

[8] Kaur, J., & Singh, B. (2014). Comparison of LSB and Predictive Coding using PSNR and MSE.

[9] International Journal of Computer Applications 98(7).

[10] Muhammad, K., Ahmad, J., Farman, H., & Zubair, M. (2015). A novel image steganographic approach for hiding text in color images using HSI color model. arXiv preprint arXiv:1503.00388.

[11] Lai, I. J., & Tsai, W. H. (2011). Secret-fragment-visible mosaic image–a new computer art and its application to information hiding. IEEE transactions on information forensics and security 6(3):936-945.

[12] Alsaidi, B.K.; Al-Khafaji, B. J.; & Wahab, S.A.A. (2019). Content Based Image Clustering Technique Using Statistical Features and Genetic Algorithm. Engineering, Technology & Applied Science Research 9(2) 3892-3895.

[13] Al-Khafaji, B. J. (2014). Detect the Infected Medical Image Using Logic Gates. Ibn Al-Haitham Journal for Pure and Applied Science 27(2): 260-267.

[14] AL-Khafaji, B.J. (2010). Image Improvement Using the Combination of Wavelet and Multiwavelet Transform. Ibn Al-Haitham Journal for Pure and Applied Science. 23(3):275-282.