International Journal of Research in Advanced Engineering and Technology
ISSN: 2455-0876; Impact Factor: RJIF 5.44
Received: 06-06-2019; Accepted: 09-07-2019
www.newengineeringjournal.in
Volume 5; Issue 3; July 2019; Page No. 82-87

A highly secure method of secret message encoding

Ziad Alqadi[1], Ahmad Sharadqh[2], Naseem Asad[3], Ismail Shayeb[4], Jamil Al-Azzeh[5], Belal Ayyoub[6]
[1, 2, 5, 6] Department of Computer Engineering, Al Balqa Applied University, Amman, Jordan
[3, 4] Princess Alia University College; Al-Balqa Applied University Amman, Jordan

Abstract
Secret, private and confidential message usually sent by embedding them in a covering media such as digital color image. Most steganographic methods used are suffering from the low level of security even if they used a private key for message hiding. In this paper we will introduce a method of message encoding-decoding, this method will use a digital color image as a source key to encode-decode secret messages, this method will provide a high level of security because of many reasons such as the huge size of the image which is used as a private key, this image is to be saved, and selected by the sender and the receiver without transmitting it over the internet. The same source image can be used to encode-decode various messages with any size even bigger than the image size.

Keywords: steganography, encoding, decoding, secret message, security

1. Introduction

1.1 Digital color image background

Digital color images (DCI) [1, 2] are the most widely used data types on the Internet and are widely used in social media, they are usually represented by 3D matrix [3, 4], and in this matrix one dimension (channel) is reserved for each color, the first channel is reserved for the red color, the second for the green color, while the third channel is reserved for the blue color as shown in figure 1.

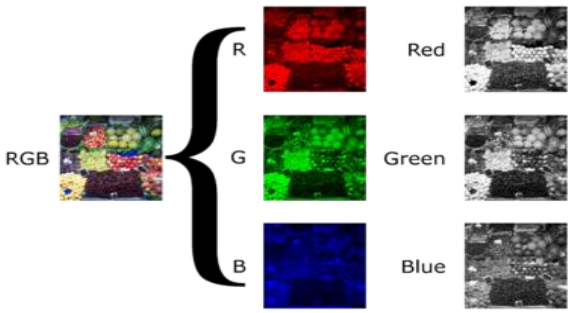

Fig 1: Color image channels

In RGB color model [5, 6, 7] red, green and blue are mixed and added together to get the resultant color which ranges from black(red=0, green=0, blue=0) to white(red=255, green=255, blue=255), figure 2 shows the color space of DCI, while figure 3 shows some samples of mixing red, green and blue color in DCI.

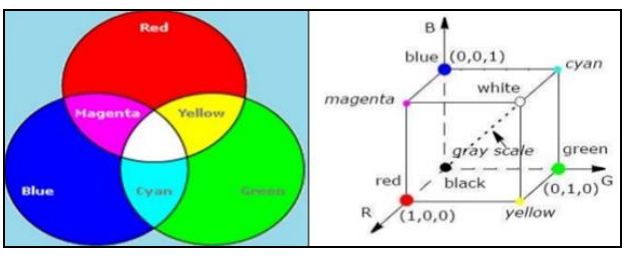

Fig 2: DCI color space

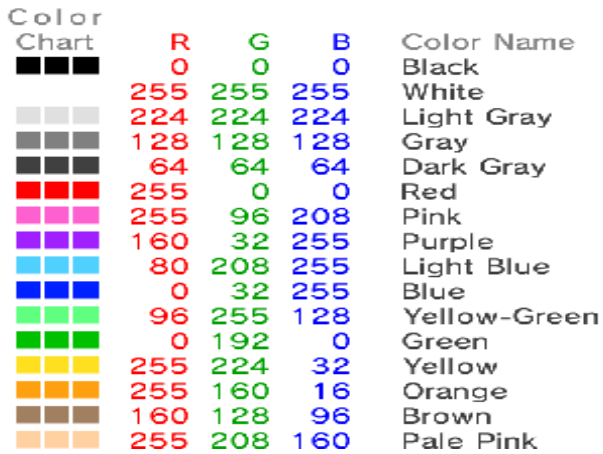

Fig 3: Samples of mixing red, green and blue colors.

Each color in DCI is represented by a 2D matrix, and each color can be extracted and treated alone as shown in figure 4:

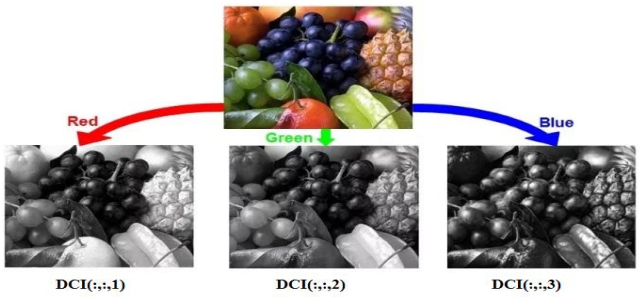

Fig 4: Extracting colors from DCI

Good DCI usually has a high resolution, and here we can lock to DCI as a 2D matrix, the intersection of the row and column is called a pixel as shown in figure 5, each pixel can be considered as a set of 3 numbers, these numbers together represent a particular color as shown in figure 6.

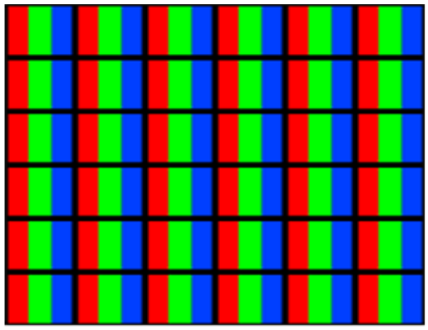

Fig 5: DCI pixels

Fig 6: Pixels sets

DCI usually has a huge size, and in order to eliminate the efforts of dealing with a huge size data structure, we can represent DCI with a histogram [1, 2].
DCI histogram is a 3 columns matrix, each column has 256 entries, and each entry represents the repetition of the associated color intensity, thus we can use the histogram as pixels intensities distribution matrix. Using histogram of DCI is very useful as it gives an intuition regarding some properties of the image such as the tonal range, the contrast and the brightness as it shown in figures 7, 8, and 9.

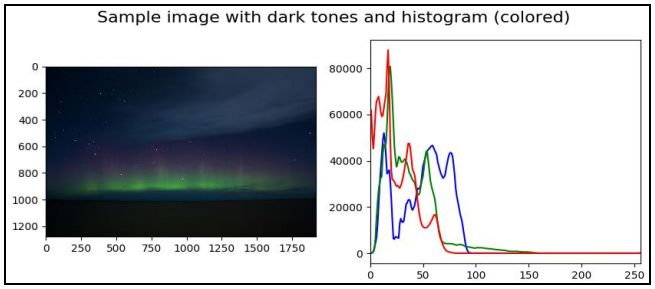

Fig 7: Low entensity image

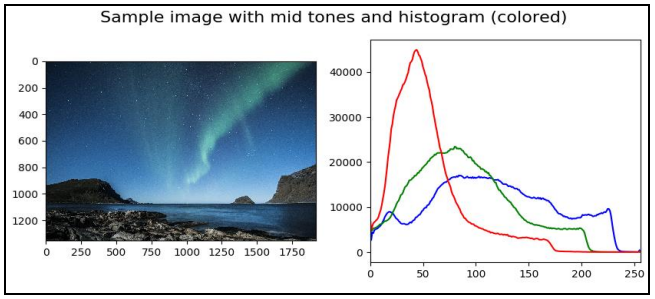

Fig 8: Mid intensity image

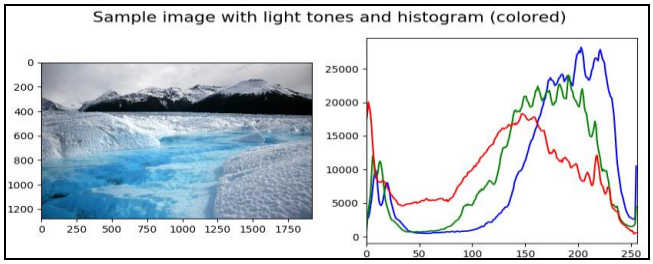

Fig 9: Light intensity image

The histogram can be calculated for each color, and using this histogram we can judge about the color properties as shown in figures 10 and 11:

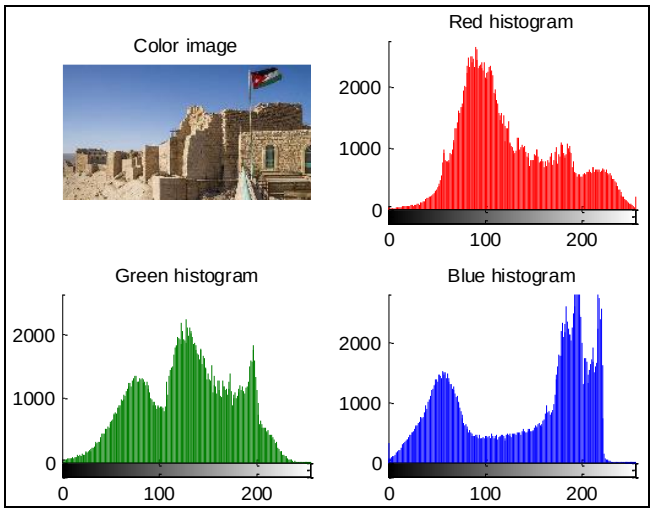

Fig 10: Red, Green and blue histograms

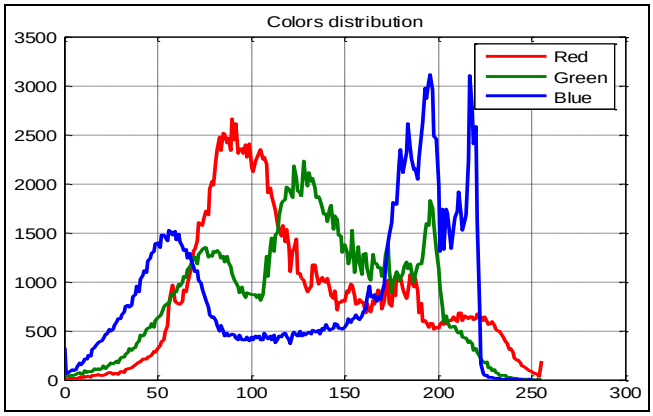

Fig 11: Colors distribution

From figure 11 we can see that all the color values (0 to 255) are covered, so all the ASCII values are covered by DCI, this feature will be used in the proposed method of message encoding (encryption).

1.2 Data steganography

Steganography is the art of hiding secrete and important data using a digital media cover in such a way that prevents attackers and Intruders from revealing data and sharing it with others. DCIs are the most popular media for data steganography [8, 9].

Steganographic process usually hides the existence of the message inside the covering image without harming it and keeping it in good quality so as not to be noticed by the human eyes [10, 11] as shown in figures 12 and 13, color image holds the message:

"*Al-Karak, also known as just Karak or Kerak, is a city in Jordan known for its Crusader castle, the Kerak Castle. The castle is one of the three largest castles in the region, the other two being in Syria. Al-Karak is the capital city of the Karak Governorate.*" (Message length=261 characters):

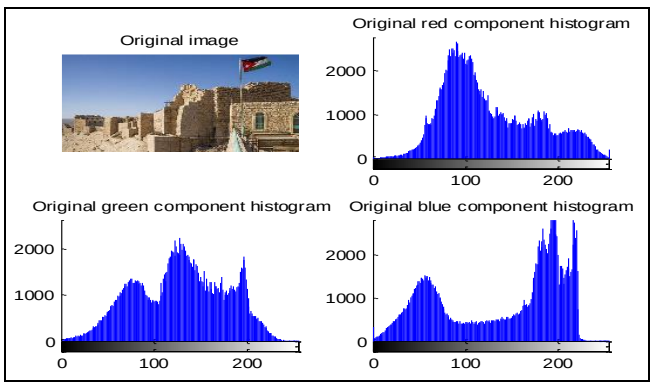

Fig 12: Original image

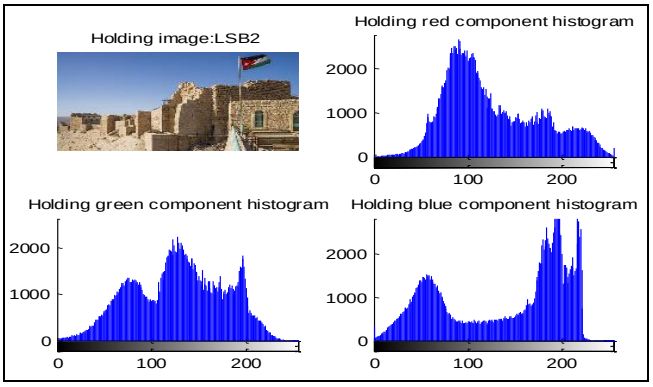

Fig 13: Holding covering image

The process of data steganography uses the following scenario [12, 13, 14] as illustrated in figure 14:

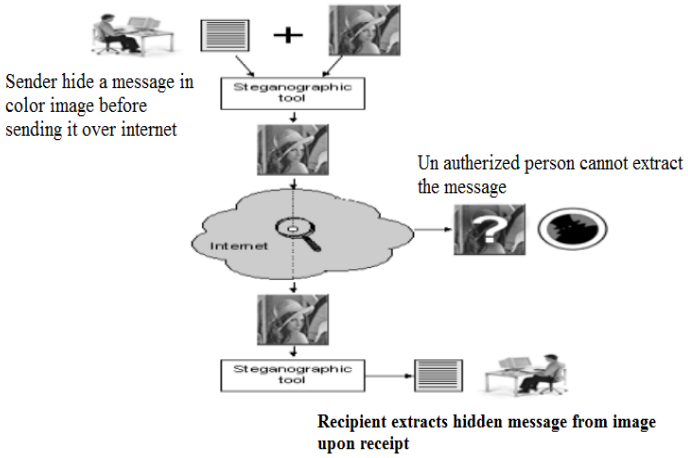

Fig 14: Steganography scenario

- The sender must select the secret message and the covering color image.
- The sender must select the method of steganography to be used as a tool to hide the message in the covering image; here the sender must select a key if needed.
- The sender must apply the tool to hide a message, and then transmit the covering image.
- The receiver must use the same tool to extract the message from the image.

Based on the above mentioned scenario we can construct the basic model of data steganography [15, 16, 17] which is illustrated in figure 15:

First we need to understand the three blocks in the left of the image:

- Cover File, 'X': This is the file that we will use for hiding the information.
- Message, 'M': This is the secret information that we want to hide into 'X'.
- Stego-Key, 'K': Some steganographic methods need to use specific keys, or data, for hiding and recovering 'M' from 'X'.

Once we have this information, we can apply the steganographic method, 'f(X, M, K)'. The output after applying the method is called "Stego-File", denoted with 'Z'. For recovering the message, we will apply the inverse process using the same Stego-Key used for hiding the message. It is important to mention that the Cover File is not important after obtaining the secret message, so it does not matter if we cannot recover the data we modified for embedding the Message.

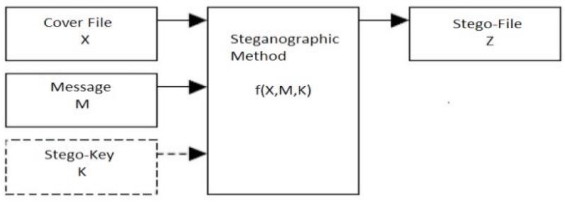

Fig 15: Basic Steganographic Model

During the secret data communication the sender and the receiver must agree on a steganographic tool used to hide-extract the data and the key used to increase the security

level of message exchange as shown in figure 16:

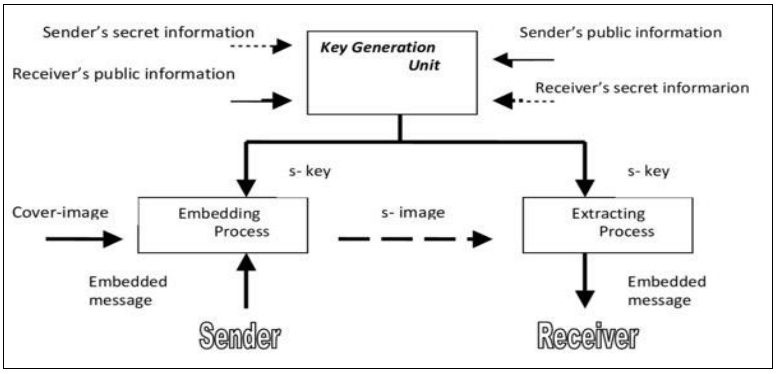

Fig 16: Secret message communication using steganography

When designing a steganographic algorithm we have to consider the following important features to be achieved by the algorithm:

Hiding Capacity

This feature deals with the maximum size of the secret data that can be embedded inside the covering DCI. A larger hiding capacity allows us to use a small covering image, and thus reduces the band-width required to transmit the stego–media. For example, if we have an RGB image with a size of 400 x 400 pixels, that means that we have 480,000 color values to be used as covering values for the secret message (400: width x 400: height x 3: R, G, B), then if we use only one bit per color channel for hiding the message we have a hiding capacity of 480,000 bits or 60,000 bytes, if we use 2 bits per color channel for hiding the message we have 120,000 bytes.

Perceptual Transparency

Perceptual transparency is an important feature of steganography. Each cover-media has certain information hiding capacity. If more information or data is hidden inside the cover, then it results in degradation of the cover–media. As a result, the stego–media and cover–media will appear to be different. If the attacker notices this distortion, then our steganographic technique fails and there is the possibility that our original message can be extracted or damaged by the attacker.

Robustness

Robustness is the ability of the hidden message to remain undamaged even if the stego–media undergoes transformation, sharpening, linear and non-linear filtering, scaling and blurring, cropping and various other techniques.

Tamper–resistance

Of all the features, this feature is very important. This is because, if the attacker is successful in destroying the steganographic technique then the tamper–resistance property makes it difficult for the attacker or pirates to alter or damage the original data.

2. The proposed method

The proposed method of message encoding (encryption) uses a color image as a key to extract the character position and uses it as a code to encode the character. Here we use a source color image to generate the encoded message, and then send the encoded message to the receiver as shown in figure 17, here the receiver must use the same image to decode the secret message as shown in figure 18.

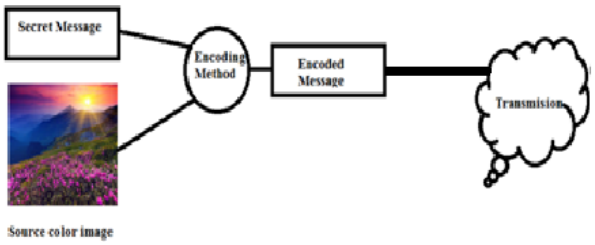

Fig 17: Message encoding

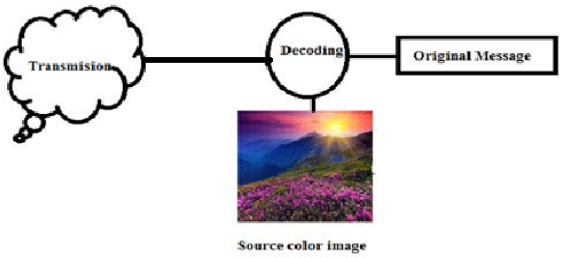

Fig 18: message decoding

The encoding process can be implemented applying the following steps:

1. Select the original source color image.
2. Retrieve the image size.
3. Reshape the color image from 3D matrix to 1D matrix.
4. Get the message
5. For each character in the message find the first position in the image which contains the ASCII value of the character.
6. Save the positions as an encoded message.
7. Send the encoded message.

The decoding process can be implemented applying the following steps

1. Get the encoded message.
2. Select the source color image.
3. Retrieve the image size.
4. Reshape the color image from 3D matrix to 1D matrix.
5. Use the encoded message values as positions to retrieve characters from the image.

3. Implementation and experimental results

The proposed method was implemented using matlab, the images shown in figure 19 were used for various

implementation. The message "ziad alqadi" was encoded using different color images, the results of implementation are shown in table 1.

From the obtained results shown in table 1 we can see the following:

1. Using various images leads to generate various coded for the same image.
2. The message character will have a unique code.
3. To avoid similarity in the encoded message we can encrypt the encoded message before sending and decrypt it when receiving.
4. The same image can be used to encode different messages as shown in table 2, here we use the following messages:

Message 1: ziad alqadi

Message 2: Albalqa applied university

Message 3: Faculty of engineering technology

Message 4: Jordan-Amman

Message 5: Computer and networks engineering

And the source image is shown in figure 20:

Fig 19: Various source digital images

Table 1: Message encoding using various images

Message = ziad alqadi									
ASCII = 122 105 97 100 32 97 108 113 97 100 105									
Image 1		Image 2		Image 3		Image 4		Image 5	
Encoded message	Encrypted message	Encoded message	Encrypted message	Encoded message	Encrypted message	Encoded message	Encrypted message	Encoded message	Encrypted message
1958	3204	133	2983	159	3005	403	2737	132	2982
1348	2033	243	582	260	945	487	850	79	762
1009	1261	222	1986	311	1579	384	1692	73	1877
702	1807	224	1361	270	1215	1413	52	75	1530
1656	3081	24	2665	9946	11435	704	2225	298	2907
1009	2847	222	2096	311	2521	384	2414	73	2215
908	1749	115	1322	214	1423	505	1184	252	1445
2115	2164	494	473	316	267	420	403	88	111
1009	2641	222	2430	311	2199	384	2080	73	2537
702	1928	224	1494	270	1080	1413	179	75	1405
1348	626	243	1989	260	1586	487	1745	79	1913

Fig 20: Source image

Table 2: Encoding various messages using the same source image

Message	Encoded message
1	132 79 73 75 298 73 252 88 73 75 79
2	22 252 240 73 252 88 73 298 73 253 253 252 79 76 75 298 89 87 79 254 76 592 112 79 594 915
3	31 73 241 89 252 594 915 298 257 244 298 76 87 77 79 87 76 76 592 79 87 77 298 594 76 241 78 87 257 252 257 77 915
4	36 257 592 75 73 87 301 22 86 86 73 87
5	28 257 86 253 89 594 76 592 298 73 87 75 298 87 76 594 585 257 592 85 112 298 76 87 77 79 87 76 76 592 79 87 77

4. Conclusions

A new method of confidential data encoding-decoding was presented, implemented and tested. The obtained results showed the following facts

- Any digital image after reshaping will cover all the ASCII characters.
- Any image can be used as a key to encode-decode secret message.
- The same digital color image can be used to encode-decode various messages.
- The transmission process requires only the encoded message.
- The security level of encoding-decoding is very high because the used source image is to be known only by the sender and receiver and the size of this image-key is very high making the process of guessing the image impossible.

5. References

1. Ziad Alqadi A, Majed Al-Dwairi O, Amjad Abu Jazar A, Rushdi Abu Zneit. Optimized True-RGB color Image Processing, World Applied Sciences Journal. 2010; 8(10):1175-1182, ISSN 1818-4952.
2. Moustafa AA, Alqadi ZA. Color Image Reconstruction Using A New R'G'I Model, journal of Computer Science. 2009; 5(4):250-254.
3. Jamil Al Azzeh, Hussein Alhatamleh, Ziad Alqadi A, Mohammad Khalil Abuzalata. Creating a Color Map to be used to Convert a Gray Image to Color Image; International Journal of Computer Applications. 2016; 153:2.
4. Jamil Al-Azzeh, Ziad Alqadi, Mohammed Abuzalata. Performance Analysis of Artificial Neural Networks used for Color Image Recognition and Retrieving, International Journal of Computer Science and Mobile Computing. 2019; 8:2.
5. Jamil AL-Azzeh, Bilal Zahran, Ziad Alqadi, Belal Ayyoub, Mazen Abu-Zaher. A Novel Zero-Error Method to Create a Secret Tag for an Image; Journal of Theoretical and Applied Information Technology, 2018.
6. Jamil AL-Azzeh, Bilal Zahran, Ziad Alqadi. Salt and Pepper Noise: Effects and Removal, International Journal on Informatics Visualization. 2018; 2:4.
7. Musbah Aqel J, Ziad Alqadi A, Ibraheim El Emary M. Analysis of Stream Cipher Security Algorithm, Journal of Information and Computing Science. 2007; 2(4):288-298.
8. Belal Ayyoub, Ashraf Abu-Ein, Ziad Alqadi. Suggested Method to Create Color Image Features Victor, Journal of Engineering and Applied Sciences. 2019; 14:7.
9. Matrouk K, Al-Hasanat A, Alasha'ary H, Al-Qadi Z, Al-Shalabi H. Speech fingerprint to identify isolated word person, World Applied Sciences Journal. 2014; 31(10):1767-1771.
10. Mohammed Abuzalata, Ziad Alqadi; Jamil Al-Azzeh; Qazem Jaber. Modified Inverse LSB Method for Highly Secure Message Hiding, IJCSMC. 2019; 8(2):93-103.
11. Mutaz Rasmi Abu Sara Rashad Rasras J, Ziad Al Qadi A. Engineering, A Methodology Based on Steganography and Cryptography to Protect Highly Secure Messages Technology & Applied Science Research. 2019; 9(1):3681-3684.
12. Ziad Alqadi, Bilal Zahran, Qazem Jaber, Belal Ayyoub, Jamil Al-Azzeh, Ahmad Sharadqh. proposed Implementation Method to Improve LSB Efficiency, International Journal of Computer Science and Mobile Computing. 2019; 8(3):306-319.
13. Deepak Garg, Gourav Sharma. Applications of Steganography in Information Hiding, international Journal of Advanced Research in Education & Technology (IJARET). 2016; 12(3):1.
14. Al-Azzeh J, Zahran B, Alqadi Z, Ayyoub B, Abu-Zaher M. A Novel zero-error method to create a secret tag for an image, Journal of Theoretical and Applied Information Technology. 2018; 96(13):4081-4091.
15. Ziad Alqadi AA, Mohammed Abu Zalata K, Ghazi Qaryouti M. Comparative Analysis of Color Image Steganography, JCSMC. 2016; 5(11):37-43.
16. Jose M. Hiding Image in Image Using LSB Insertion Method with Improved Security and Quality, International Journal of Science and Research. 2014; 3(9):2281-2284.
17. Ziad Alqadi, Bilal Zahran, Qazem Jaber, Belal Ayyoub, Jamil Al-Azzeh. Enhancing the Capacity of LSB Method by Introducing LSB2Z Method; International Journal of Computer Science and Mobile Computing. 2019; 8:3.
18. Naseem Asad, Ismail Shayeb. A Modification of Least Significant Digit (LSD) Digital Watermark Technique, International Journal of Computer Applications. 2018; 179:32.
19. Jamil Al Azzeh, Ziad Alqadi Qazem, Jabber M. Statistical Analysis of Methods Used to Enhanced Color Image; XX International Scientific and Technical Conference, 2016.
20. Mazen Abuzaher Jamil Al-Azzeh. JPEG Based Compression Algorithm; ternational Journal of Engineering and Applied Sciences. 2017; 4:4.