



ISSN: 0067-2904

Steganography Technique using Genetic Algorithm

Reyam Jassim Essa*, Nada A.Z. Abdullah, Rawaa Dawoud AL-Dabbagh
Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq

Abstract

Steganography is a useful technique that helps in securing data in communication using different data carriers like audio, video, image and text. The most popular type of steganography is image steganography. It mostly uses least significant bit (LSB) technique to hide the data but the probability of detecting the hidden data using this technique is high. RGB is a color model which uses LSB to hide the data in three color channels, where each pixel is represented by three bytes to indicate the intensity of red, green and blue in that pixel. In this paper, steganography based RGB image is proposed which depends on genetic algorithm (GA). GA is used to generate random key that represents the best ordering of secret (image/text) blocks to be hiding in the cover image. Experimental results, obtained from different experimental tests, show that our proposed system proves to be more efficient than another steganography technique presented in literature and in comparison with the original cover image in terms of fidelity criteria and degradation quality of the *stego_image*.

Keywords: steganography; security; evolutionary algorithms (EAs)

تقنية إخفاء المعلومات باستخدام الخوارزمية الجينية

ريام جاسم*، ندا عبد الزهرة، رواء داوود
قسم الحاسبات، كلية العلوم، جامعة بغداد، بغداد، العراق

الخلاصة

يستخدم الإخفاء لنقل بيانات سرية باستخدام ناقلات تسمى الغطاء مثل الصوت و الفيديو والصورة والنص. يعد الإخفاء باستخدام الصور من أكثر الأنواع إنتشاراً وذلك باستخدام تقنية الإخفاء في البت الأقل أهمية LSB، ولكن في هذه الطريقة تكون نسبة احتمال الكشف عن البيانات المخفية عالية. في النظام الملون للصور RGB يتم استخدام تقنية الإخفاء LSB لإخفاء المعلومات في قنوات الالوان الثلاثة (الاحمر، الاخضر، الازرق)، حيث يتم تمثيل كل بيكسل باستخدام ثلاثة بت تشير الى كثافة الالوان الثلاثة. في هذا البحث، تم اقتراح نظام إخفاء صورة باستخدام قناة RGB التي تعتمد على الخوارزمية الجينية. يتكون النظام من الصورة الغطاء، والبيانات السرية، الخوارزمية الجينية التي تستخدم لتوليد مفتاح عشوائي يمثل افضل ترتيب لمقاطع البيانات السرية، و تم تطبيق عدد من الخطوات لإخفاء رسالة سرية في صورة الغلاف. تظهر نتائج اختبار معايير الدقة جيدة ونسبة تشوه الغطاء قليلة خاصة في الفرق بين صورة الغلاف والصورة بعد الإخفاء.

1. Introduction

Internet nowadays is the most popular, effective and faster media for data transmission. Thus, the data senders and data receivers via the internet need a highly secure method of data protection in order to avoid a variety of problems such as hacking and eavesdropping. Cryptography and steganography are two fields for data security and protection [1]. Cryptography converts the secret message into some

*Email: reyamjassim@yahoo.com

other forms, such that it is not understandable to anyone; however, this technique has a limitation that the encrypted message is visible to everyone. In this way over the internet, intruders may try to apply head and tail method to get the secret message. Steganography differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret [2]. Steganography is the process of hiding a secret message within a large medium such as, text, audio, video and image. This is implemented in such a way that information is concealed to everyone except for the intended sender and receiver. Hiding information inside image with a secret message is popular technique nowadays and can easily be spread over the World Wide Web [3].

Practically, all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. The redundant bits of an object are those bits that can be altered without the alteration being detected easily [4]. Image is the most common cover media in steganography in which it is used to hide information. One type of images is the digital colour image that is defined as a collection of pixels which usually stored in 24-bit files and uses RGB colour model with each primary colour represented as 8-bit each.

There are some proposed methods in literature that used RGB colour channel for steganography such as hide a large amount of data (image, audio, text) file into colour BMP image which is proposed in [3]. The authors used adaptive image filtering and adaptive image segmentation with bits replacement on the appropriate pixels. These pixels were selected randomly rather than sequentially by using new concept defined by main cases with their sub cases for each byte in one pixel. Their results show that the algorithm can embed efficiently a large amount of data that has been reached to 75% of the image size with high quality of the output. Another proposed method in [5] used to perform variable length bits embedding in RGB coloured channel of colour image. Two types of channels were used. One is called indicator channel which indicates how many data bits are hidden in the data channel, the other is data channel which is used for embedding data bits. The secret message is converted using two kinds of plain text RSA plaintext and IDEA plain text. The authors in [6] proposed a technique which is a combination of both steganography and cryptography for better security of secret of information. In RGB image each pixel (24-bits) has R channels 8-bits; G channel 8-bits and B channel 8-bits. One of the channel is used as indicator channel and remaining two channel are used for hiding secret message. The indicator channel is chosen based on the sum of color values and embedding is, as 4 bit in each selected channel satisfying some conditions. Another stego method in [7] is proposed to hide secret data in different position of RGB image, which require element like cover image, secret message, two secret keys (key1 is a circular ID array with only 0 or 1 value is allowed and key2 is a ID array with 8-digits). LSB of red value of pixel is XOR with key1 bits then result is used for taking decision that secret information bit will be placed in blue or green color. Key2 is used to describe the position where secret information placed. This process is carried out repeatedly until all secret information bits are placed. It is concluded that this approach was very beneficial and secure against attacks and very effective way of hiding information without any visible distortion in the carrier image. In [8] a genetic algorithm (GA) based steganographic technique in frequency domain using discrete cosine transform has been proposed. A 2×2 sub mask of the source image was taken in row major order and Discrete Cosine Transformation was applied on it to generate four frequency components. Two bits of the authenticating image were embedded into each transformed coefficients except the first one. In each coefficient second and third positions from LSB were chosen for embedding in the transform domain. Stego sub intermediate image was generated through reverse transform. Sub mask from this intermediate image was taken as initial population. New generation followed by crossover was applied on initial population to enhance a layer of security. Rightmost three bits of each byte were taken; a consecutive bitwise XOR was applied on it in three steps which generated a triangular form. The first bit of each intermediate step was taken as the output and crossover was performed on two consecutive pixels where two LSB bits of two consecutive bytes were swapped. The dimension of the hidden image is embedded followed by the content. Reverse process was followed during decoding. Finally, in [9] a new technique was proposed based on optimization in steganography. The technique proposed was a combination of GA implemented on image steganography. This may however increase the time complexity of the algorithm but it will result in increasing robustness. This paper hide text in image using GA which guided the steganography process to the best position for data hiding in randomized LSB; where the next bit to be

used for hiding message bits was selected based on some pseudo random technique. Since the ordering in which the target bit for substitution was selected was not obvious or known earlier, it is difficult for the attacker to reveal the hidden message until the pseudo random code is known.

The aim of this work is to implement an effective stego algorithm for hiding data of type images or text in color image based on GA. In this work, GA is used to find the best random ordering of secret (image/text) blocks to increase the security level.

The rest of this paper is organised as follows. Section 2 is devoted to present a preliminary concept of standard GA. In Section 3, the proposed stego algorithm based GA is described. The experimental results of various steganography tests have been presented in Section 4. Section 5 concludes the paper.

2. Genetic algorithm

Genetic algorithm (GA) is one of the most common search algorithms that are used to find optimal or near-optimal solution to difficult and complex problems. It generally mimics the principles of Genetics and Natural Selection. The basic idea behind GAs begins with a set of candidate solutions (chromosomes) called population \mathbb{P} . The initial population $P^{G=0}$ is created from random generated solutions then undergoes the operations of crossover and mutation to produce new children. These two operations followed the selection operation that is with the help of the fitness function decides which solution will survive to the next population. The more suitable the solutions are the bigger chances they have to reproduce. This process is repeated over a certain number of generations, $MAXG$. Genetic algorithm has been widely used in many fields of science and engineering; this is due to its effectiveness and efficient performance in solving different optimization problems [10]. The general GA procedure can be stated as shown in Algorithm 1.

Algorithm 1: Genetic algorithm standard procedure

Step1. [Initialization] Generate $P^{G=0}$ which involves a set of Np random candidate solutions.

Step2. [Evaluation] Use the fitness function $f(x)$ to evaluate each candidate solution.

Step3. [New Generation] Repeat the following steps until $MAXG$ is reached.

Step3.1 [Selection] Choose randomly two chromosomes to compete, the better will survive to the mating pool.

Step3.2 [Crossover] With respect to a crossover probability P_c , select randomly two chromosomes to cross over; otherwise, the offspring is an exact copy of these two chromosomes.

Step3.3 [Mutation] With respect to a mutation probability P_m , a position in the chromosome is selected to be mutated.

Step3.4 [Evaluation] Use the fitness function $f(x)$ to evaluate each offspring.

Step3.5 [Replacement] Replace the old population with the new generated offspring.

Step3.6 $G = G + 1$

Step4. [Test] If the stopping condition is satisfied, i.e. $P^{G=MAXG}$ then stop; otherwise go to **Step3**.

3. The proposed steganography technique

Sequential steps have been adopted for implementing the proposed steganography algorithm by identifying the image (cover) with the secret message. First, the cover is divided and the secret message into blocks, the best locations for hiding message within the cover is determined, then the secret message is included in the best location to get the desired stego_image. These general steps of the proposed steganography algorithm are depicted in Figure 1.

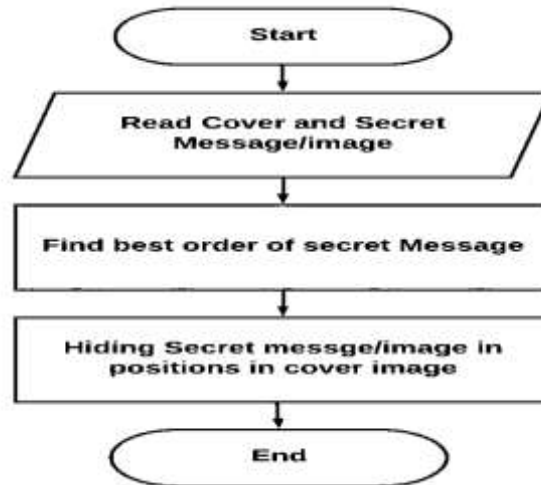


Figure 1-General steps of the proposed steganography algorithm

In this work, secret message is divide it into fixed number of blocks. Finding the best combination of n blocks is defined as a combinatorial optimization which can also be defined as NP-hard problem especially if the size of the image is large and the number of blocks is big. In this case, using an advanced search algorithm such as GA is the solution. In this proposed study, GA is used to generate a key of sequence of blocks that minimizes the fitness function in which it is defined as the MSE between the original hidden text/image and the covered image. Figure-2 depicts in detail the proposed steganography system based on genetic algorithm.

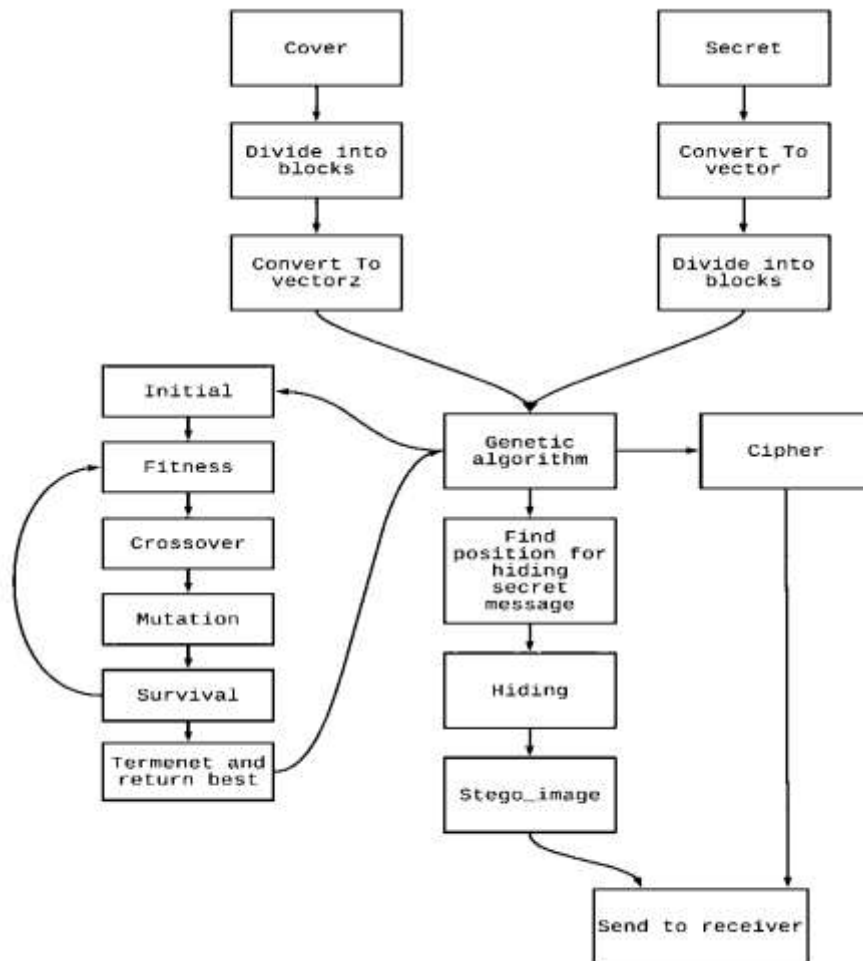


Figure 2-Steganography system based on genetic algorithm

3.1 Steganography algorithm at sender stage

This subsection summarizes the steganography process at sender stage as shown in Algorithm 2. In this algorithm, blk which is the total number of blocks that the cover image is divided to must be within the representation 2^p in which p is an integer value agreed between the two parties (sender and receiver) and $p \in \{3,4,5, \dots\}$. Thus, the size/length of secret image/text is $[2^{p-2} \times 2^{p-2}]$. In this way, the secret image/message can be divided into blocks with equal size without losing information during the hiding and the extraction process. Then, p value is used to encrypt the fit chromosome that represents the best order ordering of secret (image/text) blocks before sending it to receiver.

Algorithm 2: Steganography procedure based GA at the sender stage

Input: Load the cover image I of size $N_1 \times M_1$; Load the secret message/image (S) of size $N_2 \times M_2$.

Output: The *stego_image* of the secret message or image S hidden in I with the best blocks permutation.

Begin

Step 1: Calculate $[N_1, M_1] = size(I)$ and $[N_2, M_2] = size(S)$.

Step 2: Convert secret message / image into vector of length $L = length(S)$.

Step 3: Determine the number of cover blocks that are needed to hide S as,

$$no_blk_sec = L/blk \quad (1)$$

where blk is the total number of blocks.

Step 4: Determine the new size of the cover image which is $n \times M_1$ as,

$$n = \text{floor}(N1/\text{no_blk_sec}) \quad (2)$$

Step 5: Initialize the population P of size = 30 by rearranging the order of the blocks of the secret message using uniform random number generator. Each gene in a chromosome contains index of image pixel.

Step 6: Use genetic algorithm to find the optimal distribution of secret message blocks in cover as described in **Algorithm 3**. For each chromosome i , the best position of each block j (gene) is determined by converting each block of cover image to vector then compare all pixels of this vector with one pixel of blocks of secret message then choose minimum different of Eq. (4) to generate *stego_image* as,

$$f_{i,j} = \text{seg}_i - \text{txt}_j \quad (3)$$

where $f_{i,j}$ is the difference value between (*seg*, *txt*); seg_i are pixels of I [vector of first row]; txt_j are pixels of S .

Step 8: Hide the secret message/image within cover to create *stego_image*.

Step 7: Cipher the fit chromosome obtained from GA by adapting the BITXOR (p , fit_individual) function to increase security.

Step 8: Calculate PSNR for *stego_image* that have minimum MSE

Return The *stego_image* of the secret message or image S hidden in I with the best blocks permutation.

End.

Then, GA is used to find the optimal distribution of secret message blocks in the cover, where the search space of this problem can be defined as,

$$\text{searchspace} = (\text{no_blk_sec} - 1)! \quad (4)$$

where no_blk_sec is the number of blocks in S . In this case, the search space has been reduced as we considered to always starting with the same block index.

Algorithm 3: GA based best blocks permutation

Input: no_blk_sec ; S ; I ; $\text{MAXG} = 10$; P

Output: *fit_individual*

Begin

Step 1: Evaluate P by calculating the MSE (mean square error) for each *stego_image*:

$$\text{MSE} = \frac{1}{M_1 * N_1} \sum_{x=0}^{M_1-1} \sum_{y=0}^{N_1-1} I(x,y) - I'(x,y) \quad (1)$$

$G = 0$

Step 2: While ($G <> \text{MAXG}$)

Step 2.1 Select the *min_fit* individual for reproduction by using tournament selection

Step 2.2 Apply the partially mapping crossover (PMX) and swap mutation to generate new individuals P' .

Step 2.3 Evaluate the new individuals P' .

Step 2.4 Replace least fit of P with new individuals from P' .

$G = G + 1$

End While

Return *fit_individual*

End.

4. Experimental Results

In this section, two different experiments have been carried out to prove the effectiveness of the proposed technique. In these two experiments, the chromosome represents the genotype of the secret message/image S which is the blocks' permutation.

4.1 Experiment 1: Text in Image

In this experiment, the following set up have been considered,

- agreed fixed value ($blk = 128 = 2^7$),
- length of text = 1024,
- size of cover image = $[384 \times 384]$,
- number of secret blocks = 8,
- search space = $(8 - 1)! = 5040$,
- population size = 30

Figure-3 depicts the cover image and the secret text to be embedded. In this experiment, the MSE represents the fitness function value of each chromosome in the evolution process. Tournament selection of size 2 is applied to generate the mating pool, and then PMX crossover (with two cross point) and swap mutation are used to generate the next population. After 10 generations the best *stego_image* with the minimum MSE (that includes the best chromosome) is reserved.



(a)

(b)

Figure 0- (a) cover image and (b) secret text

Figure-4 depicts the *stego_image* obtained after 10 generations with MSE that equal to (0.73853) and PSNR that has the higher value of (49.481). This information is then will be sent to the receiver that represents the key to where this chromosome (message) is hidden.



Figure 0- *stego_image* and extract secret message with the best MSE and PSNR value

As shown in the



In image steganography, image is used to hide the information. Image is the most common cover objects in steganography. The digital color image is the collection of pixels which usually stored in 24 bit files and uses RGB color model with each primary color representing 8 bit each. Here we are reviewing some method according to the use of RGB color channel. In [1] study deals with constructing and implementing new algorithm based on hiding a large amount of data(image, audio, text)file into color BMP image. They have been used a daptive image filtering and a daptive image segmentation with bits replacement on the appropriate pixels. These pixels were selected randomly rather than sequentially by using new concept defined by main cases with their sub cases for each bit in one pixel. This concept based on both visual and statistical. According to the steps of design, they concluded 16 main cases with their sub cases that cover all aspects of the input data into color bitmap image. High security layers have been b

Figure 0, the image quality of cover image is not much affected. It is also worth mentioning that **Table 2** the results are different when using another format of cover image and secret messages of different lengths. It depends on the nature of the cover image and its contents.

Table shows results of our proposed system using four cover images of two formats (BMP, JPG) of the same size and secret message of two lengths (2048 and 1024). The MSE and PSNR for each case is calculated and show that the cover image quality is not much affected compared with the results obtained from the steganography method proposed in [1].

Table 1-Comparative results for text in image steganography

Cover	Secret	Proposed method		Steganography method in [1]	
		MSE	PSNR	MSE	PSNR
<p>London.jpg</p> 	<p>Txt2048.txt</p> 	0.4386	51.7444	24.218	34.32
<p>London.bmp</p> 	<p>Txt2048.txt</p> 	0.4286	51.8440	24.73	34.23
<p>London.jpg</p> 	<p>Txt1024.txt</p> 	0.2382	54.3958	11.23	37.63
<p>London.bmp</p> 	<p>Txt1024.txt</p> 	0.2359	54.437	12.263	37.278








<p>Flower.jpg</p> 	<p>Txt2048.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.4385</p>	<p>51.745</p>	<p>31.400</p>	<p>33.195</p>
<p>Flower.bmp</p> 	<p>Txt2048.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.4221</p>	<p>51.9111</p>	<p>31.336</p>	<p>33.204</p>
<p>Flower.jpg</p> 	<p>Txt1024.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.3614</p>	<p>52.5843</p>	<p>14.980</p>	<p>36.153</p>
<p>Flower.bmp</p> 	<p>Txt1024.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.3456</p>	<p>52.7785</p>	<p>15.889</p>	<p>36.153</p>
<p>Car.jpg</p> 	<p>Txt2048.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.0374</p>	<p>62.4399</p>	<p>16.491</p>	<p>35.992</p>
<p>Car.bmp</p> 	<p>Txt2048.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.0390</p>	<p>62.2500</p>	<p>16.729</p>	<p>35.929</p>
<p>Car.jpg</p> 	<p>Txt1024.txt</p> <p>as a result of the... (repeated text)</p>	<p>0.0235</p>	<p>64.4519</p>	<p>8.071</p>	<p>39.095</p>



Figure 5-(a) cover image and (b) secret image





Figure-6 depicts the *stego_image* obtained after 10 generations with MSE that equal to (0.04305) and PSNR that has the higher value of (61.825). This information is then will be sent to the receiver that represents the key to where this chromosome (image) is hidden. Fig shows the *stego_image* of the best chromosome and the retrieved secret image.




















Figure 6-*stego_image* and extracted secret image with the best MSE and PSNR value

The results are different when using different cover and secret image formats. Table 2 shows the results of our proposed system using four cover images of two formats (BMP, JPG) of the same size and secret image of two formats (BMP, JPG) of the same size as well. The MSE and PSNR for each case are calculated and show that the quality of cover image is not much affected compared with the results of the steganography method proposed in [1].

Table 2-Comparative results for image in image steganography

Cover	Secret	Prposed method		Steganography method in [1]	
		MSE	PSNR	MSE	PSNR
London.jpg 	redflower.jpg 	0.4072	52.0669	49.054	31.257
London.bmp 	redflower.jpg 	0.3946	52.2034	47.968	31.364

London.jpg 	Tree_cars .bmp 	2.6384	43.9514	57.751	30.549
London.bmp 	Tree_cars .bmp 	2.6083	44.0012	60.683	30.334
Flower.jpg 	redflower.jpg 	0.8186	49.0340	49.73	31.19
Flower.bmp 	redflower.jpg 	0.9241	48.5075	50.041	31.176
Flower.jpg 	Tree_cars .bmp 	2.9425	43.4776	47.571	31.390
Flower.bmp 	Tree_cars .bmp 	2.6105	43.9976	46.539	31.488
Car.jpg 	redflower.jpg 	0.4404	51.7265	36.016	32.599
Car.bmp 	redflower.jpg 	0.6141	50.2822	36.434	32.549

Car.jpg 	Tree_cars .bmp 	2.1436	44.8533	48.689	31.290
Car.bmp 	Tree_cars .bmp 	2.1778	44.7847	48.053	31.347
Trees.jpg 	redflower.jpg 	0.0144	66.5915	55.426	30.727
Trees.bmp 	redflower.jpg 	0.0053	70.9379	57.344	30.579
Trees.jpg 	Tree_cars .bmp 	0.0595	60.4206	69.150	29.830
Trees.bmp 	Tree_cars .bmp 	0.0412	62.0193	69.099	29.770

From Table-(1, 2) it can be concluded that the BMP images can have larger embedded data than JPEG images. Moreover, the image of type BMP is not distorted because of the ability of this kind of images to carry amount of data without notice, where JPEG image can have less data. In addition, the process of steganography can be affected by the details of the cover image, i.e. the more the cover image has details the better it will be to hide information.

5. Conclusions

In this study, the use of genetic algorithms has helped to increase the security through configuring a random secret key that represent the order of message / image blocks. The results represented by the value of PSNR proved the efficiency of the algorithm, as the distortion information ratio of the cover files is small and the entire text/ image has been retrieved. However, there are some problems that might be arisen when using GA is that this algorithm may require more execution time when the size of the hidden data is large.

References

1. Alqadi, Z.A.A., Zalata, M.K.A. and G. M. Qaryouti, G.M. **2016**. "Comparative Analysis of Color Image Steganography," *International Journal of Computer Science and Mobile Computing*, 5(11): 37 – 43, November.
2. Wang, H. and S. Wang, S. **2004**. "Cyber Warfare: Steganography vs. Steganalysis," *Communications of the ACM-Voting system*, 47(10): 76-82.
3. El-Emam, N.N., **2007**. "Hiding a Large Amount of Data with High Security using Steganography Algorithm," *Journal of Computer Science*, 3(4): 223 - 232.
4. Anderson, R.J. and Petitcolas, F.A.P. **1998**. "On the Limits of Steganography," *IEEE Journal of selected Areas in Communications*, 16(4): 474-481, May 1998.
5. Upreti, K. Verma and Sahoo, A. **2010**. "Variable bits secure system for color images," in 2010 Second International Conference On Advances in Computing, Control and Telecommunication Technologies (ACT) , IEEE, pp. 105-107.
6. Swain, G. and Lenka, S.K. **2012**. "A Better RGB Channel Based Image Steganography Technique," in Global Trends in Information Systems and Software Applications, vol. 270, Springer-Verlag Berlin Heidelberg, pp. 470-478.
7. Dagar, S. **2013**. "RGB based dual key image steganography," The Next Generation Information Technology Summit (4th International Conference), IEEE, pp. 316-320.
8. Khamrui, A. and Mandal, J.K. **2013**. "A genetic algorithm based steganography using discrete cosine transformation (GASDCT)," Elsevier Procedia Technology, vol. 10 pp.105 – 111.
9. Gangeshawar, J.A. **2015**. "Optimizing Image Steganography using Genetic Algorithm," *International Journal of Engineering Trends and Technology (IJETT)*, 24(1): 32-38.
10. Goldberg, D.E. **1989**. "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley Longman Publishing Co.