

# Steganography Technique for Binary Text Image

Zainalabideen Abdual Samad Rasheed

Assistant lecturer, Computer Science, Education College, Kufa University, Iraq

**Abstract:** In last decades different techniques have been created for hiding information. The role of hiding information is to create a secure communication between authorize parties in terms of exchange their vital data in safe channel. Steganography is a method of hiding data in such an approach no one can feel there is a hidden data or found it. I.e. Only the authorize party can extract these data because he know the achievement of hiding information. In this paper, an efficient method of hiding binary text image via an optimal using of Fibonacci sequence. Digital image is considered to be the cover image. A binary text image is used to represent the secret message instead of normal text. This method is used for the purpose of having a chance to read the binary text image if should any modification happen along the communication channel. (E.g. JPEG compression). A binary text image is an image which includes a text and can be construct by using any software for image processing like paint.

**Keywords:** Steganography, image quality, Traditional Fibonacci, Bit Plan

## 1. Introduction

Allocation of digital media is one of the main causes for the internet successful. That is lead to exchanging different information between people. In terms of sharing information through an open sources like the internet the needing of create a secure communication is highly important requirement. Steganography is one of the most important technologies which are used in terms of hiding information. Steganography is a method for hiding data or information within a digital cover media without leaving evidence. Leaving evidence means the modification to the cover media could be noticed by an attacker and that means the system is broken. There are two domains for applying image steganography methods, image steganography in spatial domain and frequency domain. In frequency domain, the domains careful are Discrete Cosine Transform, Discrete Fourier Transform and Discrete Wavelet Transform. There are three requirements of visibility, robustness and capacity should be achieved by techniques in two domains spatial and frequency. Visibility means the quality of stego-image, i.e. hiding process in cover media should not make the cover media sense by the human observation. While capacity mean the capacity of cover file must be preserved in attention through embedding process. However, robustness means the impossibility of delete or modification secret message. All these requirements are rest on each other and stability need be reference between them. [1][2][3]

Lately, steganography has come to be commonly used in the improvement of information security. Digital media can be used to hide secret information such as image, video, audio, text etc.

## 2. Literature View

### 2.1 Image Compression

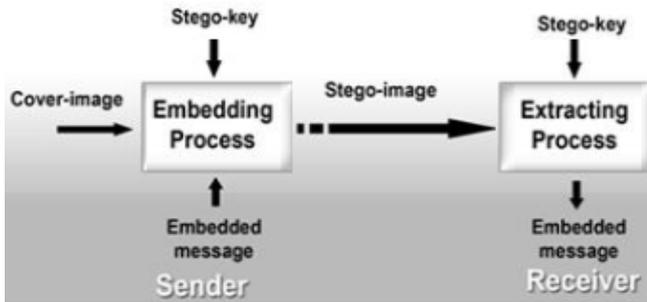
Working with images requires focusing on issues like the large sizes of images that take long time to transmit via internet as well as storage space the images requires. To reduce the time of transition and the speed display, image compression technology is used to reduce the image file size. These technologies use mathematical equations to analyses and summarize image information. There are two

types of image compression techniques; lossless and loss. Both these methods result in the file size being reduced. The resulting image file size of the loss compression process is a small image file size, and the decompressed image output approximately is close to the original image but not identical. Because this technology ignores unnecessary information from the original image, it deletes the very small details which are barely seen by the human eye. JPEG image format is an example of loss compression. However, the lossless compression technique keeps all information in the original image and does not ignore or remove any of the details in the original image. i.e. the decompressed image output is completely identical to the original image [13]. Compression technologies play an important role in the techniques of steganography. It is possible to lose the embedded message due to loss compression as it removes some details from the original image [14]. Image in spatial domain can be represented by stream of pixel such that each pixel represents the intensity value of the pixel. Each pixel value is represented by number of bits. For example, the pixel value in a grayscale image is represented by 8 bits with value range of (0 to 255). [4]

### 2.2 Steganography

According to Cachin [5], it can be explained a steganography algorithm by looking at the secure communication between sender and receiver such that this communication will be secured from the wardens. I.e. steganography allows sender to embed the secret message in a cover file and send it as stego-file to the receiver, and it allows receiver to extract the hidden message from the stego-file. From Figure 1 below us can define the general steps of steganography process.

- 1) choose the cover file (can be any type of format like image, audio, text... etc.)
- 2) determine the secret message or data (can be any type format too)
- 3) apply a steganography function for both cover and secret to hide the message in cover file.
- 4) Key is an optional and is approach depended Figure -1 shows the general process of the steganography.

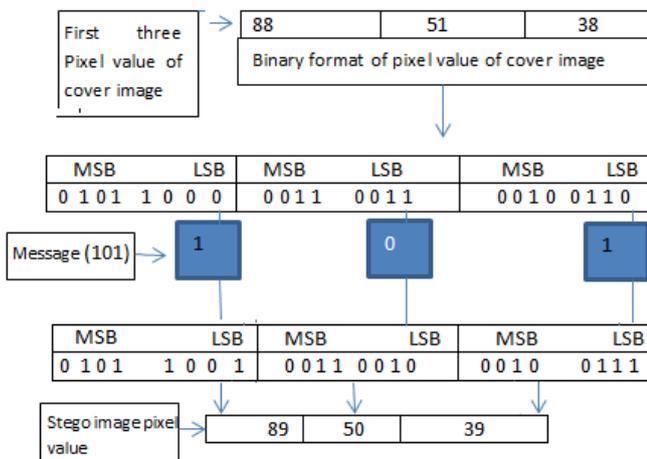


**Figure 1:** A steganography process (Cachin, 2005)

In image steganography there is variety of steganography methods. In the spatial domain the LSB method is commonly used to hide a secret message. The changes in the cover image will be less noticeable if the changes are happening in the 1<sup>st</sup> or 2<sup>nd</sup> LSB; the more you change above the 3<sup>rd</sup> and 4<sup>th</sup> bit the more effect will be noticeable on the cover image [6]. However, as long as choosing a high bit plan (MSB) your message will be robustness against modification. In This paper will focus on LSB replacement and Fibonacci sequence which give the mean of supporting highest bit plans (MSB) could be used for embedding process to choose the best bit plan. This choosing of best bit plan related to robustness against channel modification which mostly case by loss comparison algorithm.

**2.2.1 LSB Replacement (Least Significant Bit)**

In LSB replacement method, the bit of binary secret message is used to overwrite the LSB of the cover image pixel. I.e. firstly, the secret message is converted to binary bit stream. Secondly, the cover pixel value is converted to binary stream bit too. Finally, LSB of cover pixel is replaced by bit of secret message. As shown in Figure 2 the first three pixels value of cover image converted to binary bits (8 bits used to represent pixel value from 0 to 255) and secret message convert to bits too. The first bit from secret message replace with 1<sup>st</sup> LSB of first cover pixel ,second bit from secret message replace with 1<sup>st</sup> LSB of second pixel and The third bit from secret message replace with 1<sup>st</sup> LSB of third pixel [7][8].



**Figure -2:** LSB replacement process

**2.2.2 Basic Fibonacci**

“The classical Fibonacci numbers introduced in the 13th century by Leonardo of Pisa”. The sequence Fibonacci numbers is defined by the relation  $F(N) = F(N-1) + F(N-2)$ .

Where  $F(0) = 1$  and  $F(-) = 0$ .

Fibonacci sequence is [1,1,2,3,5,8,13,21,34,55,89,144,233,377,... etc. ]. It can be used to represent any numeric value like binary representation. If N is the number of bits allocated to represent any numeric value then it will be easy to distinguish between binary representation and Fibonacci representation. For example if  $N = 4$  then the maximum numeric range is from [0...15] according to binary representation (1, 2, 4, 8) for four bits. Using Fibonacci representation the maximum numeric range is from [0...7] according to Fibonacci representation (1, 1, 2, 3). For example, if  $N = 8$  then the maximum numeric range is from [0...255] according to binary representation (1, 2, 4, 8, 16, 32, 64,128) for four bits. Using Fibonacci representation the maximum numeric range is from [0...51] according to Fibonacci representation (1, 1, 2, 3, 5, 8, 13, 21). However, in binary representation there is a unique code representation for each number either for the values in [0..15] or for the values in [0..255]. Using Fibonacci, code representation could be more than one for the same number. Example, the number 51 in binary system, is represented by the unique code 110011 i.e. (1+2+16+32), whiles in Fibonacci representation, it can be represented by (34+13+3+1) or (21+13+8+5+3+1).i.e. there are redundant codes when Fibonacci representation is used to represent the numbers. Using more than one code to represent the same number makes the decomposition process incorrect in the case of Fibonacci representation; however a unique code in binary representation makes the decomposition process always correct. Hence, for correct Fibonacci decomposition a unique code is needed. Zeckendorf produce a theorem to represent a Fibonacci unique code for positive integer numbers. “Each positive integer m can be represented as the sum of distinct numbers in the sequence of Fibonacci numbers using no two consecutive Fibonacci numbers. [9][10][11]

**2.2.3 The Traditional Fibonacci Method**

The Fibonacci sequence can be used in data hiding in digital files (image) .i.e. Fibonacci representation will be used to represent a pixel value of image instead of binary representation. in traditional Fibonacci method, the 12 bits of Fibonacci sequence is used to cover the intensity values instead of using 8 bits in binary system to represent pixels value from (0,255).the hiding and extracting process can be explained as steps below:

- 1) the secret message is converted to binary system
- 2) the intensity values of cover image is converted to Fibonacci representation
- 3) the bit of secrete message is replaced with the bit of the Fibonacci representation
- 4) Decompose the Fibonacci representation into the intensity value of the cover image
- 5) keep track for which pixel is used for hiding process

For extracting the secret message this steps will be followed

- 1) The track will be used to determine the selected pixels
- 2) The value of the selected pixel is compose to Fibonacci representation
- 3) Secret message is extracting from the cross pounding bit plan of the Fibonacci representation. [12]

**2.2.3.1 Problem of using 12 bits in traditional Fibonacci**

In traditional Fibonacci LSB, 12 bits is used to represent a pixel value, the Fibonacci sequence will be [233,144,...,3,2,1], in this case we can find unique codes for all values in the range of [0..255] and all pixels value will be represented by a unique code. However, this representation has a problem after the embedding process due to getting same code appearing more than one for some values i.e. there is no guarantee to get unique code in conversion process. From table 5 we can see codes of some pixels value whose values are greater than 232 and how embedding operation will make the pixel value jump over 255. x in the table is suppose to be a bit plane which might be used for the hiding process. It is clear when x is set to one the pixel value will jump over 255 and will result in losing this bit in the retrieving process, because in the retrieving process 255 is usually represented by the code [1 0 0 0 1 0 0 0 0 0 1]. Therefore we will lose the bit plane x because every time it will be zero in the retrieving process i.e. these bit planes can not be used in embedding process due to the reason above. However, this representation allows usage of all pixels values in cover image for hiding but not all bit planes. Table 5 below shows some codes and the effect of these codes for some bits plane

**Table 1:** Some values and a fibonacci code of these values

233	144	89	55	34	21	13	8	5	3	2	1	12 bits fibonacci sequence
1	0	0	0	0	0	0	0	0	0	0	0	Unique Code of pixel value (233)
1	x	x	x	x	0	0	0	0	0	0	0	effect Code for some bit plan of pixel value(233)
1	0	0	0	0	1	0	0	0	0	0	1	Unique Code of pixel value (255)
1	x	x	x	x	1	0	0	0	0	0	1	effect Code for some bit plan of pixel value(255)
1	0	0	0	0	0	0	0	0	0	0	1	Unique Code of pixel value (234)
1	x	x	x	x	0	0	0	0	0	0	1	effect Code for some bit plan of pixel value(234)

**2.3 Proposal System**

**There are two proposals for the problem in traditional Fibonacci 12 bits:**

**2.3.1 First our Proposal**

We can use 11 bits to represent Fibonacci code of pixel value but skip the pixels value when it is greater than 232 to avoid the problem of representing the values (233,255) i.e. all pixels value from (233,255) will not be used in the hiding process. In this case using 11 bits guarantees the getting of unique code for all pixels values from (0,232) in the retrieving process. However, the capacity of hiding will be limited i.e. all values greater than 232 will not be used.

**2.3.2 Second our Proposal**

We can use 11 bits to represent Fibonacci code of pixel value and use all pixels available in cover image for the hiding process. This solution gives a full capacity of cover image for the hiding process and also gives guarantee to retrieve all secret messages in the retrieving process because it keeps the unique code for each pixel values. However, this solution makes more changes to the pixel value than the previous solution because in the conversion process the maximum value will be 232. If pixel value is greater than 232 the change to these pixel values will be higher than the change to all pixels value that are less than 232 although both use the same bit plane. From table 6 below we can see that if the pixel value is 243 and we embed zero to the 9<sup>th</sup> bit the value of the pixel will be 177. However, if the pixel value is 232 and we embed zero to the 9<sup>th</sup> bit the value of the pixel will be 177. That tells us the change that happen to the first pixel value is 65 greater than change to the second pixel value which is 55 although they use the same bit plane bit. Result in less quality steego image than the first proposal for the same bit plan.

**Table 2:** Change to the pixels value a using 11 bits Fibonacci representation

144	89	55	34	21	13	8	5	3	2	1	11 bits Fibonacci sequence
11 <sup>th</sup>	10 <sup>th</sup>	9 <sup>th</sup>	8 <sup>th</sup>	7 <sup>th</sup>	6 <sup>th</sup>	5 <sup>th</sup>	4 <sup>th</sup>	3 <sup>rd</sup>	2 <sup>nd</sup>	1 <sup>st</sup>	Bit plan
1	0	1	0	1	0	1	0	1	0	1	Pixelvalue(243) before embedding
1	0	0	0	1	0	1	0	1	0	1	Pixelvalue(177) after embedding 9 <sup>th</sup> bit plan
1	0	1	0	1	0	1	0	1	0	1	Pixelvalue(232) before embedding
1	0	0	0	1	0	1	0	1	0	1	Pixelvalue(177) after embedding 9 <sup>th</sup> bit plan

**3. Result and Discussion**

In our testing, different type of cover image format with different sizes have been tested. For example, same size for binary text image and cover image were tested. Also, smaller binary text image was tested. The binary text image has been tested with different font sizes. As long as the size of font increases the capacity of binary text image decrease (I.e. the capacity of text in a binary text image increase with decrease the size of font). Increase in the font size of binary text image increases readability of binary text image after JPEG compression. The binary text image is generated using paint software. In this project, (400\*400\*3 byte) leena400.jpeg cover image and (400\*400\*1 logical) TESTIMAGEBINARY48FONT.bmp binary text image are used to test for all methods below. The font size of the binary text image ‘‘ TESTIMAGEBINARY48FONT.bmp’’ is 48.

The size of cover image and font size is selected randomly. For all methods peak signal-to-noise ratio (PSNR) is used to measure the quality of an image as stated in literature review. Test results for all methods are discussed below. All methods were implemented in matlab 2010 on an HP computer with Intel i3 processor and 6G Ram. Below are two images a cover image (leena400.jpeg) and binary text image (TESTIMAGEBINARY48FONT.bmp’’)

leena400.jpeg    TESTIMAGEBINARY48FONT.bmp



Quality of stego image peak signal-to-noise ratio (PSNR)'' It is the measure of quality of the image by comparing the cover image with the stego image, i.e., it measures the statistical difference between the cover and stego image, '' [15]. The formula of PSNR is given by:

$$PSNR(dB) = 10 * \log\left(\frac{255^2}{MSE}\right)$$

The formula of MSE is given by:

$$MSE = \frac{1}{mn} \sum_0^{m-1} \sum_0^{n-1} \|f(i,j) - g(i,j)\|^2$$

Where:

f (i ,j): is pixel value of cover image . g(i,j): is the pixel value of Stego image. mn: is the size of an image. [13]

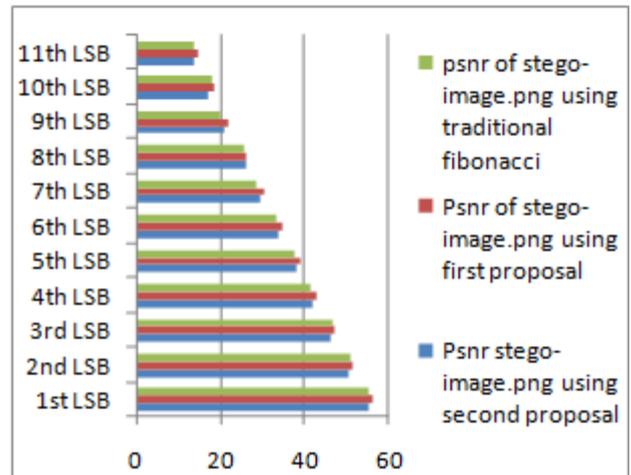
### 3.1 Previous Result

The prevoise result is made for testing the three methods below :

- 1) First our propsal (using 11 bits of fibonacci sequece skipping pixel value greater than 232)
- 2) Second our proposal (using 11 bits of fibonacci sequece without skipping pixel value greater than 232)
- 3) Traditonal fibonacci sequece using 12 bits .

acompatrive have been applaid on the fibonacci sequece for all bit plans for the three methods; first our proposal,second our proposal and traditional fibonacci to choose the best method in terms quality stego-image only within fibonacci sequece.and it is clear from Figure the quality of stego-image for the first our proposal is the the best from the rest.this prevouis compartive has been applaied to choose a method which will be compare later with LSB replacent in terms of roubstness and qaulity of stego-image.

Below Figure 3 shows PSNR of Stego\_image.png for all bit plans using three methods : first our proposal,second our proposal and traditional fibonacci



**Figure 3:** psnr of stego-imge.png for all bit plans using three methods,first our proposal , second our proposal and traditional fibonacci

### 3.2. Final Result

In final result a testing have been applied on LSB replacement and first our proposal only for all bit plans.

#### 3.2.1. The First our Proposal Result

As we can see from Figure 4 by using level bit planes from the 7<sup>th</sup> bit plan to 11<sup>th</sup> bit plan the binary text image could be redable after jpeg compression (Stego\_image.jpg) but at the expense of quality of stego image. However,by using 6<sup>th</sup> bit plane to 1<sup>st</sup> bit plane the quality of stego image is good but at the expense of reading binary text image using first our proposed method. Below Figure 4 shows binary text images and Stego\_images.jpg with all levels of LSB from 1<sup>st</sup> ,2<sup>nd</sup> ...11<sup>th</sup> using first our proposed method

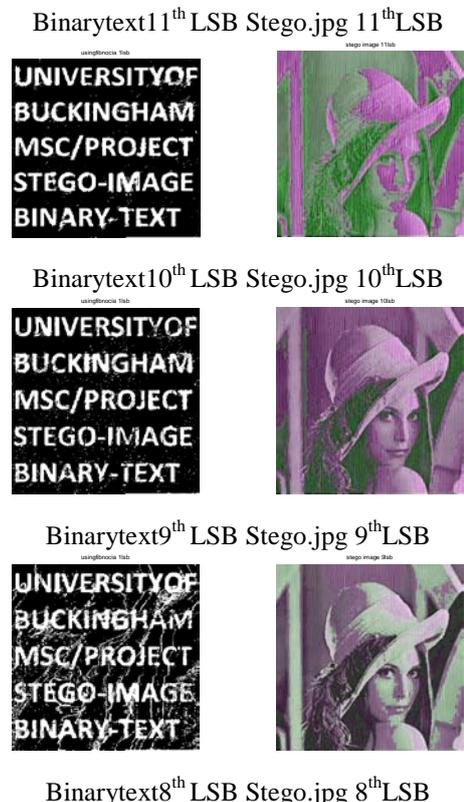
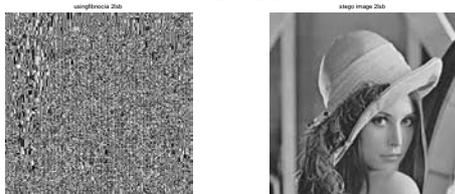
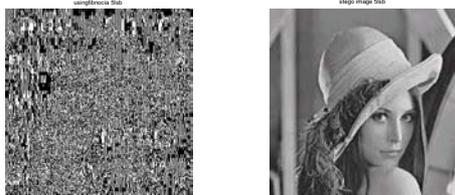
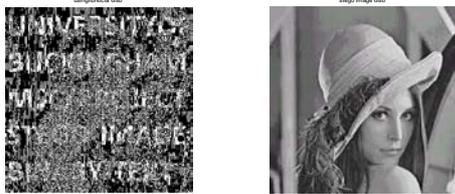
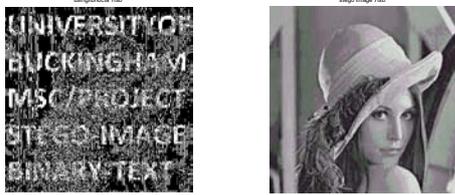


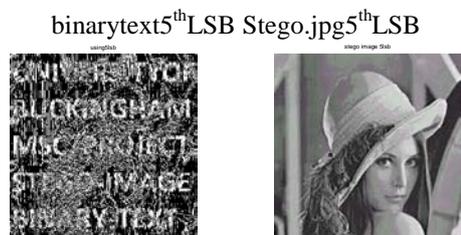


Figure 4: the binary text images and Stego\_image.jpg with all levels of LSB using first our proposed method



### 3.2.2. LSB replacement

For LSB replacement method we can see from the table 8 as long as one increases the level of LSB embedding the quality of Stego image decrease (i.e. PSNR of Stego image decrease). However, the effect of jpeg compression on retrieving binary text image decrease with increase in the level of LSB embedding. As shown in Figure 5 the binary text images could be readable if we choose 6<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> bit planes. However, the quality of stego image decreases with these levels of bit planes. Figure 5 below shows the binary text images and Stego\_image.jpg with all levels of bit planes using LSB replacement.



Binarytext



**Figure 5:**Stego\_image.jpg and binary text image for all level of LSB using LSB replacement

Moreover, a useless compression has applied on (Stego\_image.png) using first our proposed method and LSB replacement and the binary text image for all level of LSB is fully readable. This type of image compression will keep the information of image after compression without any losing.

#### 4. Conclusion

1. It is clear from Figure 3, the quality of Stego\_image (PSNR) of first our proposed method (using 11 bits, skipping the pixel value greater than 232) is better than both methods; second our proposal (using 11 bits without skipping pixel value 232 method) and traditional fibonacci (using 12 bits method).
2. Traditional fibonacci using 12 bit it is not possible to choose high bit planes 7<sup>th</sup> or more bit plane for embedding process because the problem which is explained in previous part (2.2.3.1). that is mean the level of bit plane from 7<sup>th</sup> to 11<sup>th</sup> which be applied in first our method and is justified terms of robustness against lossy compression is not allowed to use in traditional fibonacci due to problem in part (2.2.3.1)
3. First our proposal better than LSB replacement in terms of quality image. In binary bit representation each byte can be represented by 8 bits for the value from 0 to 255 (1,2,4,8,16,32,64,128) while in Fibonacci each byte can be represented by 11 bits using Fibonacci series 1,2,3,5,8,13,21,34,55,89,144. For example if we aim to

use the 7<sup>th</sup> LSB of pixel value then the change in pixel value using binary bitwise will be to add or subtract 64 from the pixel value. However, using Fibonacci series 21 will be added or subtracted according to Fibonacci representation. Less change to the pixel value means less distortion to the cover image resulting in a high quality stego image.

4. In terms of jpeg compression, (the effect of jpeg compression will be affecting the possibility of reading the retrieved binary text image after applying jpeg compression to stego image). As shown in Figure 4 and 5, first our proposed method using 7<sup>th</sup> LSB bit plane and above results in a readable binary text image with better quality of stego-image (PSNR) compared with LSB replacement. Better quality of first our proposal is discussed in the previous point in conclusion. Hence, for robust application against JPEG compression first our proposal is considered the best. due to using a high bit planes with less change to cover image compare with lsb replacement.

#### References

- [1] Mammi, E., et al. "A novel spatial data hiding scheme based on generalized Fibonacci sequences." *Proc. SPIE6982 C 69820* (2008): 1-7.
- [2] Bhattacharyya, S., Kshitij, A., & Sanyal, G. (2010). A Novel Approach to Develop a Secure Image based Steganographic Model using Integer Wavelet Transform. *IEEE*, (pp. 173-178).
- [3] Sarkar, D., Nag, A., Biswas, S., & Sarkar, P. (2011). A Novel Technique for Image Steganography Based on DWT and Huffman Encoding. *International Journal of Computer Science and Security (IJCSS)*, 4(6), 561.
- [4] Owens, M. (2002). A discussion of covert channels and steganography. *SANS Institute*.
- [5] Cachin, C. (2005). Digital steganography. *Encyclopedia of Cryptography and Security*, 129-168.
- [6] Silman, J. (2001). Steganography and steganalysis: an overview. *SANS Institute*.
- [7] Ker, A. (2005). Steganalysis of LSB matching in grayscale images. *Signal Processing Letters, IEEE*, 12(6), 441-444.
- [8] Chan, C., & Cheng, L. (2004). Hiding data in images by simple LSB substitution. *Pattern Recognition*, 37(3), 469-474.
- [9] Jr, V. H., Cox, N., & Bicknell, M. (1973). A Primer for the Fibonacci Numbers: Part XII. *The Fib. Quart*, 11, 317-331.
- [10] Picione, D., Battisti, F., Carli, M., Astola, J., & Egiazarian, K. (2006). A Fibonacci LSB data hiding technique.
- [11] Aroukatos, Nikolaos, et al. "Data hiding techniques in steganography using fibonacci and catalan numbers." *Information Technology: New Generations (ITNG), 2012 Ninth International Conference on*. IEEE, 2012.
- [12] Borgohain, R. (2012). Data Hiding Techniques using number decompositions. *Arxiv preprint arXiv:1206.4155*.
- [13] Bhaskaran, V., & Konstantinides, K. (1997). *Image and video compression standards: algorithms and architectures*. Springer.

- [14] Das, S., Das, S., Bandyopadhyay, B., & Sanyal, S. (2011). Steganography and Steganalysis: different approaches. *Arxiv preprint arXiv:1111.3758*.
- [15] Kumar, K., Raja, K., Chhotaray, R., & Pattanaik, S. (2010). Bit Length Replacement Steganography based on DCT Coefficients. *International Journal of Engineering Science and Technology*, 2(8), 3561-3570.

### Author Profile



**Zainalabideen Abdullasamd Rasheed** University of kufa, Education of College, Najaf /Iraq. Has a BSc (Baghdad University, Iraq), and MSc (Buckingham university, UK).he has a long experience in teaching various computing practical courses at Baghdad university .at AL Kufa university, Mr. Rasheed teaches different courses like computer organization, operating system and computer architecture .he supervises undergraduate projects. He interest in data security (steganography and cryptography) and digital image processing (biomedical image and edge detection, de-noising images).E-mail:zain9999@live.com,Tel:009647711131246 .Iraq.