Secure RGB Image Steganography from Pixel Indicator to Triple Algorithm-An Incremental Growth

Namita Tiwari1[1], Madhu Shandilya[2]
[1] *Asst.Prof.,Deptt.of computer science,MITS,Gwalior,India*
namita_tiwari21@rediffmail.com
[2] *Professor, Deptt. Of Electronics, MANIT Bhopal,India*
madhu_shandilya@yahoo.in

Abstract

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information. In this paper we have investigate two methods of RGB image steganography one is pixel indicator technique and other is triple-A algorithem.They uses the same principle of LSB, where the secret is hidden in the least significant bits of the pixels, with more randomization in selection of the number of bits used and the color channels that are used. This randomization is expected to increase the security of the system and also increase the capacity. These techniques can be applied to RGB images where each pixel is represented by three bytes to indicate the intensity of red, green, and blue in that pixel. This work showed attractive results especially in the capacity of the data-bits to be hidden with relation to the RGB image pixels.

Keywords: *- Steganography, randomization, RGB Bitmap images, Triple-A Algorithm, Pixel Indicator Algorithm.Computer Security*

1. Introduction

Steganography is the art of hiding information into another covering media in a way that nobody except the receiver can detect the secret message and retrieve it.Steganography (which means "covered writing" in Greek) is an old art that has been used since the golden age of Greece where some practices were recorded like: writing a message on a wooden table then covering it with wax, and tattooing a messenger hair after shaving and then let his hair grow up before sending him to the receiver where his hair was shaved again. Other techniques use invisible ink, microdots, converting channels and character arrangement. Digital steganography[3] has many applications in today's life. It could be used as a digital watermarking to protect the copy-rights, or to tag notes to digital images (like post-it notes attached to paper files), or to maintain the confidentiality of valuable data from possible sabotage, theft, and unauthorized viewing.Image-based steganography techniques[4] need an image to hide the data in. This image is called a cover media. Digital images are stored in computer systems as an array of points (pixels) where each pixel has three color components: Red,Green, and Blue (RGB). Each pixel is represented with three bytes to indicate the intensity of these three colors (RGB).

Some techniques have been used for image steganography such as LSB, SCC and image intensity .In LSB, the least significant bit of each pixel for a specific color channel or for all color channels is replaced with a bit from the secret data. Although it is a simple techniques, but the probability of detecting the hidden data is high. SCC technique is an enhancement. The color channel, where the secret data will be hidden in, is cycling frequently for every bit according to a specific pattern. For example, the first bit of the secret data is stored in the LSB[6] of red channel, the second bit in the green channel, the third bit in the blue channel and so on. This technique is more secure than the LSB but still it is suffers detecting the cycling pattern that will reveal the secret data. Also it has less capacity than the LSB.

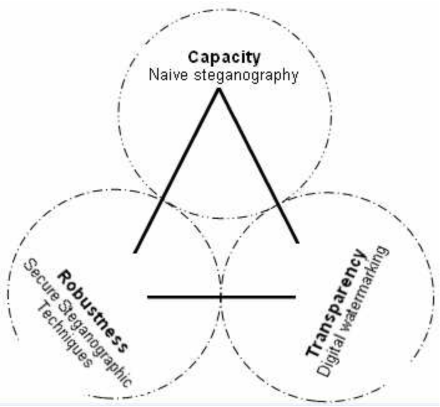


Figure 1. Steganography tradeoff parameters

Designing any stego algorithm should take into consideration the following three aspects (Figure 1):

- Capacity: The amount of data that can be hidden without significantly changing the cover medium.
- Robustness: the resistance for possible modification or destruction in unseen data.
- Invisibility (Security or Perceptual Transparency): The hiding process should be performing in a way that it does not raise any suspicion of eavesdroppers.

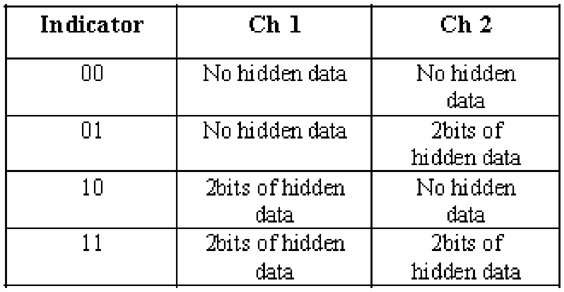
Figure 1, shows the relation between these three main parameters. If we increase the capacity of any cover to store more data than a practical possible threshold, then its transparency or robustness will be affect and vice versa. Similarly, transparency and robustness are related; if any of these two parameters are influenced, it can affect the performance in the other one. The capacity, robustness, and security parameters relation issues can be driven by the application need and its priorities.

The flow of the paper is as follows. Section 2 describes the pixel indicator method and Section 3 discusses triple algorithm. Implementation overview of pixel indicator method and triple algorithm provided in Section 4. Section 5 presents experimentations and comparisons of both methods. Finally section 6 gives the conclusion remarks of the work.

2. PIXEL INDICATOR TECHNIQUE

We analysis this pixel indicator technique[2] for RGB images steganography. The technique uses least two significant bits of one of the channels Red, Green or Blue as an indicator of data existence in the other two channels. The indicator bits are set randomly (based on the image nature) in the channel. Table 1 shows the relation between the indicator and the hidden data inside the other channels.To improve security, the indicator channel is not fixed. The indicators are chosen based on a sequence. In the first pixel Red is the indicator, while Green is channel 1 and Blue is the channel 2. In the second pixel,Green is the indicator, while Red is channel 1 and Blue is channel 2. In third pixel Blue is the indicator, while Red is channel 1 and Green is channel 2.

Table 1. Meaning of indicator values.



Indicator	Ch 1	Ch 2
00	No hidden data	No hidden data
01	No hidden data	2bits of hidden data
10	2bits of hidden data	No hidden data
11	2bits of hidden data	2bits of hidden data

3.TRIPLE-A: AN ALGORITHM

Figure 2 shows the Triple-A algorithm[1] taking the message (M), the carrier image (C), and the password based generated key (K) depending on password (P), as inputs and produces the message (M) hidden inside the carrier image (C).

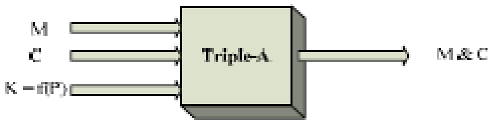
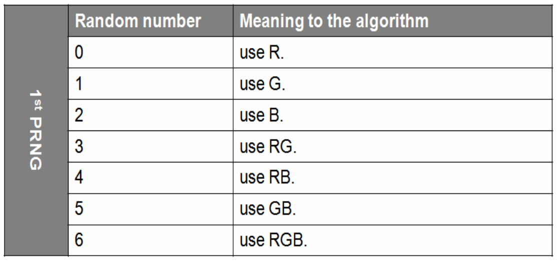


Figure 2. Inputs and outputs of triple-A an algorithm.

The RGB Image is used as a cover media. It utilizes the advantage of the Bmp images, where every pixel is independent from the reset of the image file. Enc (M, K) is hidden according to our triple-A algorithm of Fig. 2 which needs to have a pseudorandom number generator (PRNG).The assumption for PRNG is to give two new random numbers in every iteration. The seeds of these PRNGs namely Seed1 (S1) and Seed2 (S2) are formed as a function of the Key (K). S1 is restricted to generate numbers in [0, 6] while S2 is restricted to the interval [1, 3]. S1 random number is used to determine the component of the RGB image which is going to be used in hiding the encrypted data Enc (M, K).

Table 2 shows how (S1) random number selects the RGB components. On the other hand, (S2) random number determines the number of the component(s) least significant bits that is used to hide the secret data. On the same way Table 3 shows how (S2) random number determines the number of component bits. Bmp images are represented in computer systems as a two-dimensional array of X-position and Y-position. X-position and Y-position for the pixels with hidden data is distributed inside the image according to the size of the secret data Enc (M, K) and the carrier image (C) computed.

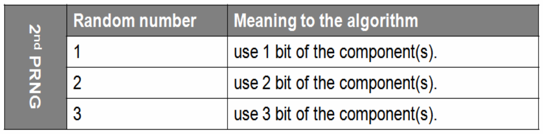
Table 2. Seed 1 random number usage.



	Random number	Meaning to the algorithm
1st PRNG	0	use R.
	1	use G.
	2	use B.
	3	use RG.
	4	use RB.
	5	use GB.
	6	use RGB.

Table 2 shows that the maximum number of bits used from a component are 3 bits. While table 3 shows that the maximum number of bits, which is used to determine the component bits, are 2 bits. This indicates that the algorithm may add up to a maximum of ± 7 to the value of the color component(s) in that pixel.

Table 3. Seed 2 random number usage



	Random number	Meaning to the algorithm
2nd PRNG	1	use 1 bit of the component(s).
	2	use 2 bit of the component(s).
	3	use 3 bit of the component(s).

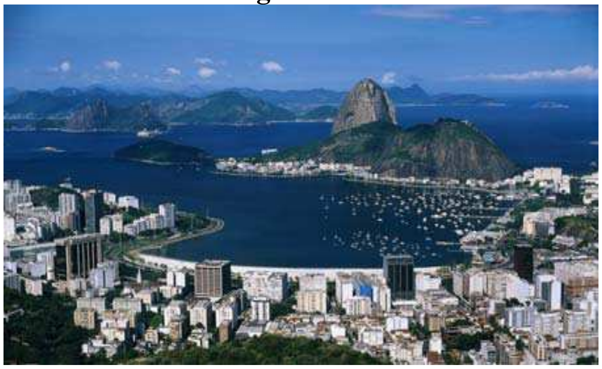
Also, by combining data from the previous tables, we can see that the minimum number of bits used in each pixel is 1 if we use only one bit of one chosen components of the RGB image. The maximum is 9 bits if we used all the three components with three bits.

4. IMPLEMENTATION OVERVIEW

Triple-A algorithm

The Triple-A algorithm is increased the capacity ratio and the security level of the concealment operation. Theoretically, the average number of bits used per pixel is equal to 3.428 where the maximum number of bits used in SCC could be 3 and for LSB is 1. This shows us that the capacity of the triple technique is higher than the previous techniques. By using this algorithm, the ratio between the number bits used inside a pixel to hide part of the secret message; and the number of bits in the pixels itself, which defined as the capacity factor can be in the range from 1/24 to 9/24 if we use a maximum of 3 bits. Moreover, if we extends the algorithm to hide 4 and even 5 bits the factor can be increased up to 15/24 which is above half of the pixel bits, but the down side is the additional noise introduced as the number of bits used to hide the secret data get higher

A: original carrier.



B: carrier with secrete using Triple-A algorithms.

Figure 3. Image steganography testing example.

Our algorithm has the same unpredictable message size as the pixel indictor scheme but the Triple has maximum capacity ratio better than the pixel Indicator. Also, the unpredictability in pixel indicator is function of the image carrier (C) which is usually has mega sizes. Triple-A in the other case depends on the key (K) which is of smaller size.

pixel indicator

The randomization is in the indicator channel content, which gave different results in every run based on the starting pixel to hide data. After many runs of the algorithm we came up with some results. Frequency analysis, histogram, was performed to check the change in the cover image. Also, the number of pixels used in each run to hide data was recorded. Note that the visual change between the original image and stego-image cannot be predicted. However, the differences between the images before and after hiding the data can be sensed through histograms. Histograms of the RGB channels: Red, Green, and Blue, are tested separately to study the method security. Figures 4 and 5 show the histogram of the Red channel pre and post the hiding process. Similarly,

Figures 6 and 7 show the histogram of the Green channel pre and post the hiding process. Also, Figures 8 and 9 show the histogram of the Blue channel pre and post the hiding process.

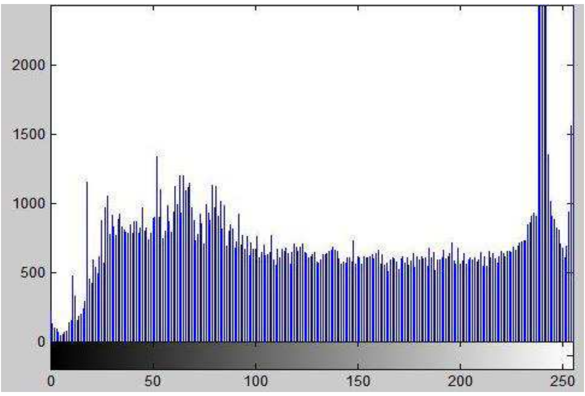


Figure 4. Original image histogram of the red channel

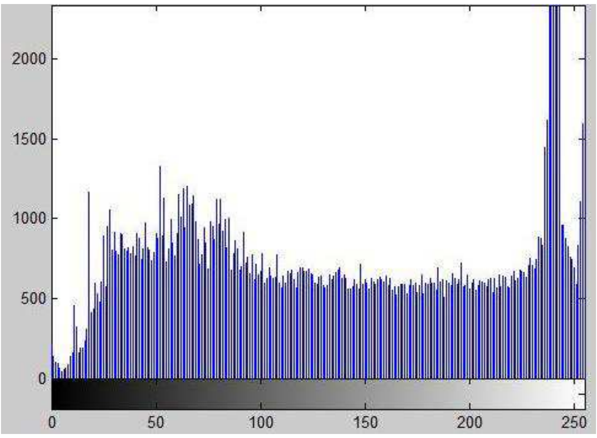


Figure 5. Modified image histogram of the red channel

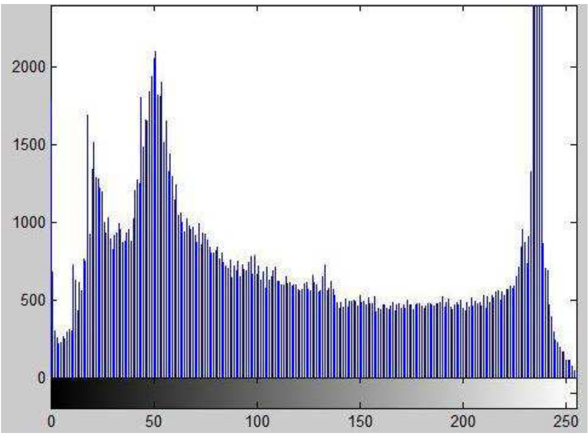


Figure 6. Original image histogram of the green channel

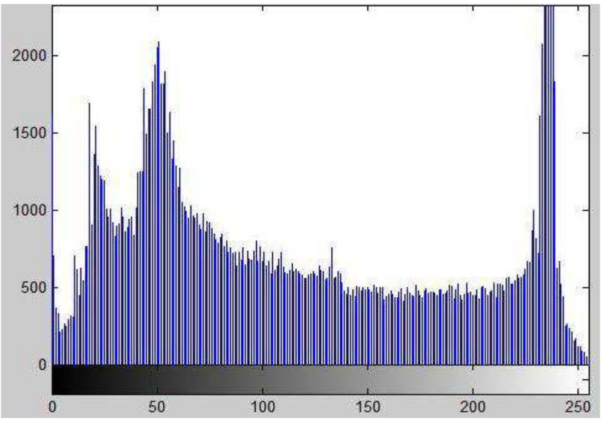


Figure 7. Modified image histogram of the green channel

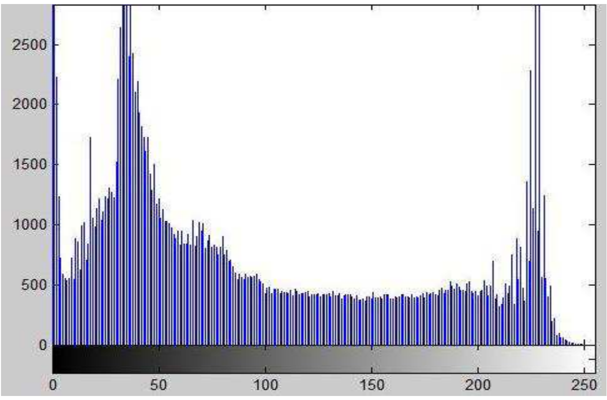


Figure 8. Original image histogram of blue channel

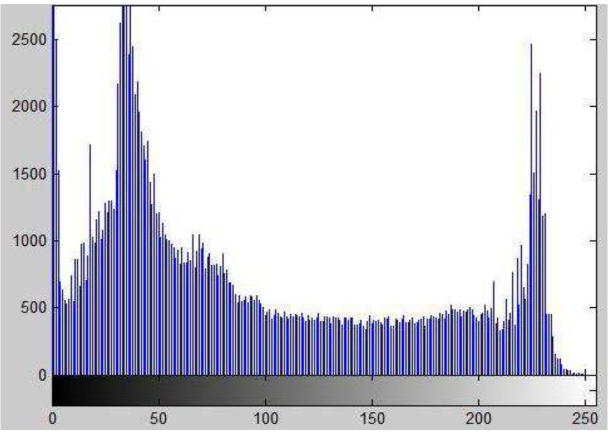


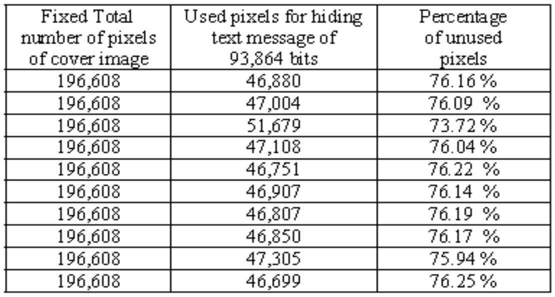
Figure 9. Modified image histogram of blue channel

By comparing the two histograms of the three RGB channels before and after hiding data, some security feedback can be concluded. Observe the red and green channel histograms before and after the modification (Figures 4 & 5, and Figures 6 & 7), the change cannot be detected; both, Red and Green, channels can be used to give some security promise. However, changes can be detected clearly in the histogram of the blue channel,which puts some future work to investigate the reasons and implications of this issue.

EXPERIMENTATIONS & COMPARISONS

In pixel indicator interestingly, from the different test runs, we got different distributions between the three channels, which continued varying between the channels with no apparently detected pattern. This undetectable pattern changing within RGB channels gave us the promise that this analysed technique may be considered random or pseudorandom based on the randomness of the indicator channel. The different testing results are listed in Table 4, which shows the randomness in the results. This sample study gave us hint of the capacity of this algorithm. Hiding the text message of 11,733 characters length (93,864 bits) within the image of 196,608 pixels utilized less than fourth of these pixels. It can be roughly concluded for the algorithm capacity that a pixel is needed to hide every two bits. The robustness of algorithm is not investigated thoroughly. The first impression about robustness is that it is achieved electronically, as long as the image is not modified or compressed. If this robustness issue is true, it can be considered as clear drawback of the method needing more future study. In general, this algorithm may open new directions in steganography research leading to interesting results.

Table 4. Capacity of cover image



Fixed Total number of pixels of cover image	Used pixels for hiding text message of 93,864 bits	Percentage of unused pixels
196,608	46,880	76.16 %
196,608	47,004	76.09 %
196,608	51,679	73.72 %
196,608	47,108	76.04 %
196,608	46,751	76.22 %
196,608	46,907	76.14 %
196,608	46,807	76.19 %
196,608	46,850	76.17 %
196,608	47,305	75.94 %
196,608	46,699	76.25 %

Triple-A algorithm is implemented using software package developed using C#. The resulting stego-images is tested and compared with the original images by using histograms generated by MATLAB to check the level of noise or distortion caused by the Triple-A algorithm. The results are compared with other stego-images generated using pixel algorithm. The level of distortion and the capacity issues are highlighted. Figure 3 shows an original carrier compared to the same carrier with secret using Triple-A algorithm. From the first moment, you cannot see difference within the images; but the histogram of the images shows a minor difference in the value of the components: R, G and B. It is shown that triple-A algorithm enhance the capacity ratio with a factor of around 4.

If we look at the image as a whole, since we used PRNG, we should average the number of bits used to hide the secrete data over the image carrier. Thus, the capacity ratio is = (Number of bits used each possible case)/ (Total number of cases * 24).

Since we have a total of 21 cases decomposed as:

Using One component case: here we have 3 ways to determine the bits * 3 ways to decide the component R, G or B. this results in 9 cases.

Using Two component case: here we have 3 ways to determine the bits * 3 ways to decide the component RG, RG or GB. This result in 9 cases.

Using Three component case: here we have 3 ways to determine the bits * one way to decide the component which is RGB. This result in 3 cases.

The average capacity ratio is around 1/7 or 14% of the original cover media size. The secret data is scattered throughout the whole image. Also, extracting the secret data without the Knowledge of seeds is almost impossible.

The secret message has retrieved correctly using both methods. As a stego-analysis procedure, Triple-A which is more difficult to guess.

CONCLUSION

Triple-A concealment technique is introduced as a new method to hide digital data inside image-based medium. The algorithm adds more randomization by using two different seeds generated from a user-chosen key in order to select the component(s) used to hide the secret bits as well as the number of the bits used inside the RGB image component. This randomization adds more security especially if an active encryption technique is used. The capacity ratio is increased above SCC and pixel indicator scheme. Triple-A has a capacity ratio of 14% and can be increased if more number of bit is used inside the component(s). Triple-A algorithm has the same unpredictable message size as the pixel indictor scheme but the Triple-A has maximum capacity ratio better than the Pixel indicator. Also, the unpredictability in pixel indicator is function of the image carrier (C) which is usually has mega sizes. Triple-A in the other case depends on the key (K) which is of smaller size.

References:

[1] Adnan Gutub, Ayed Al-Qahtani, Abdulaziz Tabakh,” Triple-A: Secure RGB Image Steganography Based on Randomization” IEEE,pp.400-403,2009

[2] Adnan Gutub, Mahmoud Ankeer, Muhammad Abu-Ghalioun, Abdulrahman Shaheen, and Aleem Alvi, Pixel indicator high capacity technique for RGB image based Steganography,WoSPA 2008 – 5th IEEE International Workshop on Signal Processing and its Applications,University of Sharjah, Sharjah, U.A.E. 18 – 20 March 2008.

[3] Mehdi Kharrazi, Husrev T. Sencar, and Nasir Memon” Image Steganography: Concepts and Practice” WSPC/Lecture Notes Series: 9in x 6in, Institute of mathematical sciences,Singapore April 22,2004

[4] Neil F Johnson&Sushil jajodia,“Exploring Steganography: Seeing the Unseen” IEEE computer,february 1998 pp26-34

[5] Lisa M. Marvel,“Image Steganography for Hidden Communication”Final rept.jan1997-may1999 2000.

[6] R. Chandramouli, Nasir Memon, “Analysis of LSB Based Image Steganography Techniques” Proc. IEEE ICIP pp1019-1022, 2001.

[7] Li Zhi,Sui Ai Fen, “Detection of Random LSB Image Steganography”IEEE pp2113-2117,2004

[8] Mehdi Kharrazi,Husrev T. Sencar,Nasir Memon”Performance study of common image steganography and steganalysis techniques” Journal of Electronic Imaging vol.15(4), pp 041104,Oct–Dec 2006

[9] Hedieh Sajedi, Mansour Jamzad" Cover Selection Steganography Method Based on Similarity of Image Blocks"IEEE 8th International Conference on Computer and Information Technology Workshops pp379-384,2008

[10] Ismail Avcıbas, Mehdi Kharrazi, NasirMemon, Bulent Sankur"Steganalysis with Binary Similarity Measures" EURASIP Journal on Applied Signal Processing 2005:17, 2749–2757,2005

[11] T Morkel, JHP Eloff and MS Olivier, "An Overview of Image Steganography," in Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005 (Published electronically)

[12] Sanjeev Manchanda, Mayank Dave and S. B. Singh," Customized and Secure Image Steganography Through Random Numbers Logic" Signal Processing: An International Journal, Volume 1: Issue (1)2007

[13] Yanming Di, Huan Liu, Avinash Ramineni, and Arunabha Sen," Detecting Hidden Information in Images:A Comparative Study", Department of Computer Science and Engineering Arizona State University, Tempe, AZ 85287,2005.

[14] Bret Dunbar," A detailed look at SteganographicTechniques and their usein an Open-SystemsEnvironment", As part of the Information Security Reading Room, SANS Institute 2002.

[15] J.Fridrich,M.Goljan and R.Du,"Reliable detection of LSB steganography in color and grayscale images"proc. ACM workshop multimedia security ,Ottava,ON Canada ,Oct 5,2001,pp27-30.

[16] K. Satish, T. Jayakar, Charles Tobin, K. Madhavi and K. Murali" Chaos Based Spread Spectrum Image Steganography" IEEE Transactions on Consumer Electronics, Vol. 50, No. 2,pp 587-590 MAY 2004

[17] Xiangwei Kong,Ziren Wang,Xingang You, "Steganalysis of palette images:Attack Optimal Parity Assignment Algorithm"

[18] Rafael Gonzalez & Richard woods, "Digital Image processing" prentice HALL 2002

[19] Kurak C., J. McHugh, "A Cautionary Note on Image Downgrading", IEEE Eight Annual computer security.Applications Conference, pp. 153-159, 1992.

[20] Mamta Juneja, Parvinder singh Sandhu,"Designing of Robust Image Steganography Technique Based on LSB Insertion and Encryption "IEEE,pp.302-305,2009

[21] Mohammad Tanvir Parvez and Adnan Gutub, "RGB Intensity Based Variable-Bits Image Steganography", APSCC 2008 – Proceedings of 3rd IEEE Asia-Pacific Services Computing Conference, Yilan, Taiwan, 9-12December 2008

[22] M Amin, M. Salleh, S . Ibrahim, M.R.K atmin, and M.Z.I. Shamsuddin," Information Hiding using Steganography,4th National Conference on Telecommunication Technology Proceedings, Shah Alam,Malaysia,pp.21-25,2003