 ISSN: 2321-8169
2020 – 2026

Embedding Approach of Audio Data in RGB Images Using Circle Equation

Devendra Singh Rao
Department of Compute Science Engineering
Pacific Institute of Technology, Udaipur
raodeven@gmail.com

Pankaj Singh Parihar
Department of Compute Science Engineering
Institute of Technology & Management, Bhilwara
pankajsinghparihar2002@gmail.com

Abstract: - Steganography is a process of hiding secret message into lager sized covers. Secret data and covers can be text, images, audio, video and protocols. Image steganography have image as covers to embed secret information ensuring that there no much difference in size and appearance in original cover and newly generated image. In this paper we have used JPEG, PNG, BMP and TIFF images are used as covers. Secret data is taken audio in WAV data format. An approach is suggested to embedding audio data into RGB components of cover image. RGB data are extracted from image using circle equation and substitute LSB's of cover from audio data.

Keyword: *Image Steganography; RGB; LSB; Audio Embedding; Circle Equation; Pixel Data*.

*****

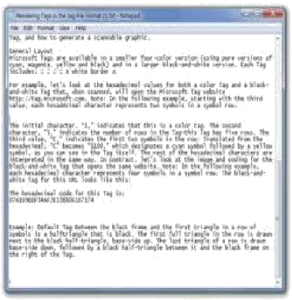
I. INTRODUCTION

In present scenario there is a high amount of sensitive data travel over high speed delivery channels. Steganography plays role of hiding this digital data. Steganography is a different technique 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 [1]. Steganography and cryptography are both methods to secure information from unwanted users but neither technology alone are efficient and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of Steganography is partly defeated. The strength of Steganography can thus be amplified by combining it with cryptography [4].

A. Cover File

Digital formats with high redundant bits can be used to hide secret messages, so the implications of replaced bits are unnoticeable. Steganography can be done trough Text, Image, Audio-Video and Protocols as Shown in Figure 1.

Using Text file as a cover medium is quite small in size and can have small amount of message and audio-video is very large to transmit over the network. So the Image has medium size and sufficient enough to hide secret messages, so images are quite popular in steganography.



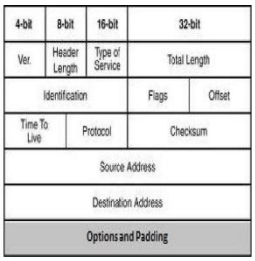


Figure 1: Steganography file formats: Text, Image, Audio-Video and Protocol

Images are presented as grid, where each individual point is called a 'pixel'. Pixels are displayed horizontally, row by row. In a color scheme, the number of bits is known as the 'bit depth' and this basically refers to the number of bits assigned to each pixel. The smallest bit depth is represented by 8 bits. Thus, in any given pixel, the number of different shades of red, green, and blue can reach 256 (2^8) that adding up to more than 16 million (256^3) combinations that finally result in more than 16 million colors.

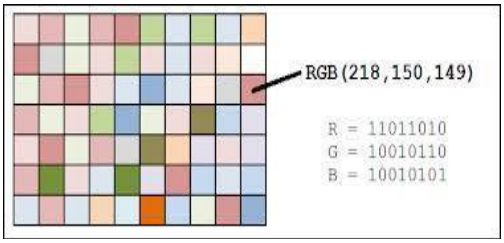


Figure 2: Matrix representation of RGB image

Various Image formats experimented are: [2]

- **BMP:** Capable of storing 2D images of arbitrary width, height and resolution.
- **JPEG:** The degree of compression can be adjusted, allowing tradeoff between size and quality.
- **PNG:** Supports images of 24 bit RGB, employs lossless data compression, specially designed for transferring images over internet.

 ISSN: 2321-8169
2020 – 2026

- **TIFF:** Widely supported by image manipulation applications.

B. ***Message File***

There are three major groups of audio file formats [3]

- **Uncompressed:** The most actual representation of sound wave. Eg. WAV, AIFF.
- **Lossless Compression:** Stores data in less space by elimination unnecessary data. Eg. MPEG4 SLS, MPEG4 ALS, MPEG4 DST.
- **Lossless with Lossy:** Enables even greater reduction of file size by removing some of the data. Eg. MP3, Vorbis, Lossy Windows Media Audio.

C. Substitution Techniques and RGB Color Model

Substitution technique is a process of replacing LSB's of cover image data by message data bits.When we are using 24-bit image then we can embed 8-bits of audio message into each pixel data. Replacements of bits take in place 3+3+2 bits in Red, Green and Blue component of a pixel respectively. For example given vector shows a pixel of 24 bit color image, using 3 bytes of memory.
(10100**111** 10101**001** 11011**00**)
When an 8 bit binary (10010011) inserted, resultant will be
(10100**100** 10101**100** 11011**11**)

The following diagram shows, the LSBs modified in proposed algorithm. 3 bits of each Red and Green and 2 bits of Blue are modified. The secret data would be put in these LSBs only.

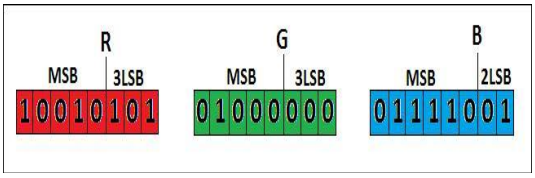


Figure 3: LSBs of Red, Green and Blue component

II. PROPOSED APPROACH AND ALGORITHM

A. Circle Equation

Pixels used in substitution are selected using mathematical equation of circle. Where x and y are length of axis and r is radius.

$$(x-a)^2 + (y-b)^2 = r^2$$

Centre of the image (a,b) can be found by dividing the length and breadth of image by 2. That is, $(a,b) = (X_{max}/2, Y_{max}/2)$

Figure 4: Understanding circle equation

The first key in encryption (and decryption) process is the radius of circle, *r*. And the second key is the pixel position, over the circle traced with this radius, which acts as a starting point for writing (and reading) on the circle. After the whole circle is covered, next radius is selected and a new pixel position.

B. Algorithm Steps for Encoding

1. Read the audio file.
2. Convert the audio file into binary format and determine its length (size).
3. Read the image file and trace its centre.
4. Choose a radius length, not already used; this is the first secret key for this round.
5. From the circle covered by above radius, consider a pixel position as a second secret key for this round.
6. Starting from this pixel position, store positions of all pixels of this circle into a matrix.
7. Retrieve Red, Green, Blue color values in binary format at these pixels into a matrix.
8. Substitute the last bits (3+3+2) of R, G and B with audio binary data.
9. Replace the old R, G, B values with new R', G', B' values in the image.
10. Repeat steps 4 to 8 till last audio bit is not reached.

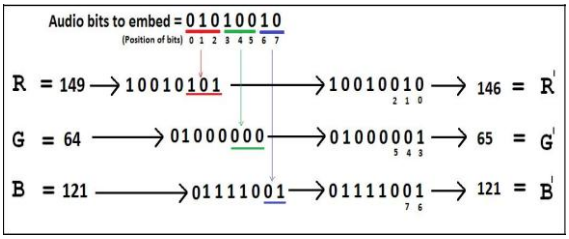


Figure 5: LSB substitution of pixel value by audio data

C. Algorithm Steps for Decoding

1. Read the image file and trace its centre.
2. Use the first secret key (radius) to trace the circle which is hiding the message.
3. Starting from the first pixel position (second secret key), retrieve Red, Green, Blue color values in binary format of all the pixels of the circle into a matrix.
4. Extract the 3 LSBs of Red and Green and 2 of Blue into a matrix, each row giving one byte of data.
5. Append the bits rows obtained above, one after another.
6. Repeat steps 2 to 5 till all circle are read.
7. Process the above bits to generate the audio message.

III. IMPLEMENTATION

As of now, the tools and process to be used are decided. The implementation activity includes a prototype implementation of the proposed work. It is applied to test the hiding approach that whether the algorithm works effectively or not.

The test cover image and audio file for this evaluation

 ISSN: 2321-8169
2020 – 2026

experiment are randomly selected from internet. The dimension of image considered is 1920*1080 pixels, in different file formats. An image specially having circles and hazy color distribution is selected for high performance of algorithm. The wav audio file is 20KB in size. Matlab 7.5.0 software platform is used to perform the experiment.

1. Read a sample *png* cover image into a variable.

   PICpng = imread('pic1.png');

2. Read the secret message Wav file

   [y,Fs,FORMAT] = wavread ('sample.wav');

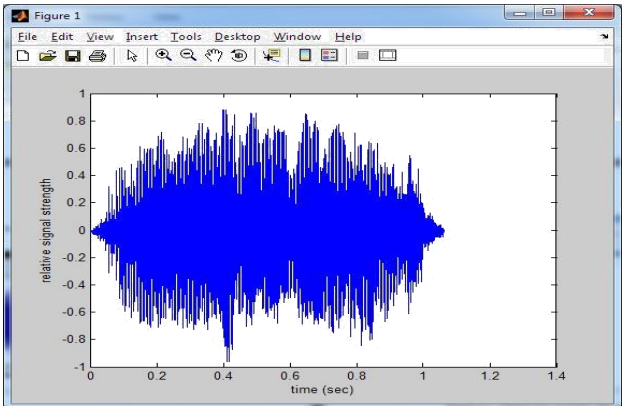


Figure 6: Histogram of Audio file

3. Covert WAV data into binary (0,1) form and store in a variable *wavbinary*, and its length in *wavbinary_size.*

   wavdata = wavread('sample.wav');
   wavbinary = dec2bin(typecast(single(wavdata(:)), 'uint8'), 8) - '0';
   orig_size_audio =size(wavdata);
   wavbinary_size = size(wavbinary);

Figure 7: Sample bits of Audio data in binary format with its length

4. Trace centre*(centx, centy)* of the cover image, as all circles would originate from centre. *y* and *x* are the length and width of image.

   [y x numberOfColorChannels] =size(PICpng);
   centx = x / 2;
   centy = y / 2;

5. Define a random radius *r (key 1)* and store positions of all pixels over that circle in a matrix *YX*

   r = 500;
   *theta = 0 : (2 * pi / 2000) : (2 * pi);*
   *pline_x = round(r * cos(theta) + centx);*
   *pline_y = round(r * sin(theta) + centy);*
   YX = vertcat(pline_y, pline_x);

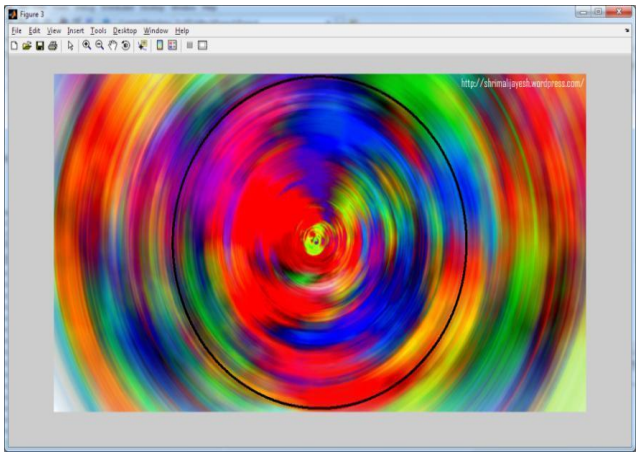


Figure 8: Screenshot of the image with circle on which audio data would be placed

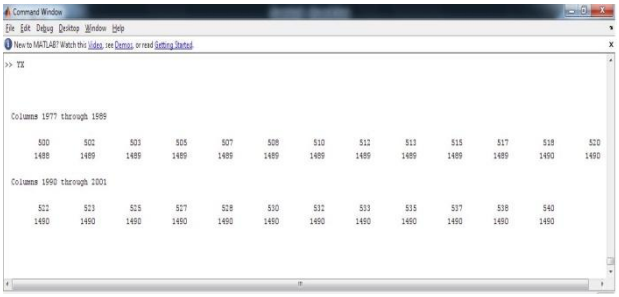


Figure 9: Screenshot of pixel positions obtained YX

6. Calculate space available in a circle and number of circles, *n*, required to hide the whole secret message. Every circle would have different radius, thus different storing capacity. At each pixel, 3+3+2 (=8) LSB are substituted in R,G,B respectively.

   *spaceInCircle = size(pline_x)*8*
   No of circle required = n;
   while(wavbinary_size) >= totalOfAllCircles(spaceInCircle)
   n++;

7. Start reading from a specified pixel position (key 2), and store the R,G,B values of all pixels of the circle in a matrix, *mat*, in three columns.
   for k = 1:2:size_of_matrix(YX),
   R = PICpng(YX(k), YX(k+1),1);
   G = PICpng(YX(k), YX(k+1),2);
   B = PICpng(YX(k), YX(k+1),3);
   new_row = [R , G , B];
   end

 ISSN: 2321-8169
2020 – 2026

mat_binary = de2bi(mat,8,'left-msb');

8. Replace LSBs of each R, G, B present in matrix with Audio binary data (*w*). For substituting last three bits of Red component:

for k = 1:size(mat)
nR = new_mat_binary(k,1:8);
for a = 1:3,
newR = de2bi (bitset (bi2de (nR,'left-msb'), a, substring (wavbinary,w,w)),
8,'left-msb');
nR= newR;
a=a+1;
w=w+1;
end
mat_Red = [mat_Red; nR];
k= k+1;
end

Similarly *mat_Green* (3 LSBs) and *mat_Blue* (2 LSBs) would be generated.

9. Combine all the three color matrix in a single matrix and store it in decimal format

mat_final = [mat_Red; mat_Green;mat_Blue];
mat_final_dec = bi2de(mat_final,'left-msb');

10. Replace old R, G, B color values of image with newly developed matrix.

l = start_position_of_Red;
m = start_ position_of_Green;
n = start_ position_of_Blue;
for k=1:2: size_of_matrix(YX),
PICpng(YX(k), YX(k+1),1) = mat_final_dec(l);
PICpng(YX(k), YX(k+1),2) = mat_final_dec(m);
PICpng(YX(k), YX(k+1),3) = mat_final_dec(n);
l = l+1;
m = m+1;
n = n+1;
end

11. Testing the algorithm using the PSNR (Peak signal-to-noise ratio). PSNR is a standard measurement used in steganographic technique in order to test the quality of the stego images. The higher the value of PSNR, the more quality the stego image will have [21].

Calculate the mean square error, *mse* and Peak Signal to noise ratio *PSNR_Value* using following formulae.

*mse = 1/yx * (sum((histo of new image – histo of original image).^2)) ;*

*PSNR_Value = 10 * log10(MAX^2 / mse);*

The pixels are represented using 8 bits per sample, so MAX = 255.

IV. RESULT ANALYSIS

Sample pixel positions are selected to analyze the change in image. Following table shows the values of Red, Green and Blue component at various pixel positions over the circle selected on cover image. These values are later converted into binary format for processing.

Table 1: Pixel positions, showing RGB values

Pixel Position (x,y)	Red	Green	Blue
430,540	124	22	98
887,15	124	15	144
960,1070	242	63	22
1287,123	36	136	4
1490,540	149	64	121

Table 2 shows the change in Red, Green and Blue component values after LSB substitution. And then, table 3 compares the change in old binary values of color component with new values.

Table 2: Decimal values of RGB pixel positions
(Changed Values are Underlined)

Pixel Position	Red	Green	Blue
430,540	120	18	98
887,15	124	8	146
960,1070	240	60	22
1287,123	36	137	4
1490,540	146	65	121

Table 3: Audio Data Embedded at pixel positions
(Change bits are underlined)

Pixel Position	Audio Data Embedded	Red Old / New	Green Old / New	Blue Old / New
430,540	000 010 01	1111100 / 1111000	1010 / 1000	1100010 / 1100010
887,15	001 000 01	1111100 / 1111001	1111 / 1000	10010000 / 10010001
960,1070	000 001 01	11110010 / 11110000	111111 / 111001	10110 / 10101
1287,123	001 001 00	100100 / 100100	10001000 / 10001001	100 / 100
1490,540	010 001 01	10010101 / 10010010	1000000 / 1000001	1111001 / 1111001

Each pixel position would store one byte of secret data. As shown above, for embedding 5 bytes (that is 40 bits) of data, only 14 bits are modified, that is a 35% change.

Table 4 show the peak signal to noise ratio of performance of the proposed method of hiding audio file (mono.wav) into different formats of cover image (pic1.*).

ISSN: 2321-8169
2020 – 2026

Table 4: Experiment Results

Cover File	Original Image Size (Kb)	MSE	PSNR (dB)
pic1.png	5852	2.9885e-009	133.3763
pic1.jpg	1525	1.4679e-015	133.2966
pic1.bmp	6076	2.9885e-009	133.3763
pic1.tiff	4575	Not Supported	Not Supported

The algorithm best suits for JPEG image, as it is small in size and gives PSNR equivalent to other formats. The change in image for storing secret message on one circle is shown through histograms.

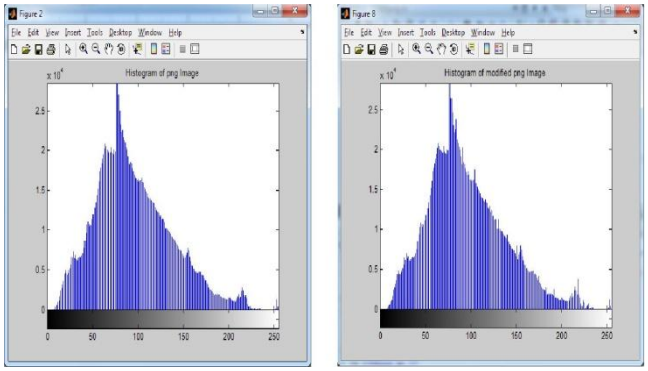


Figure 10: Histogram of Cover Image and Stego image

Figure 11: Comparing the 24 bit JPG, Original Cover Image and Stego Image

As proved that the approach is most suitable for JPG images, the popular JPG images are compared as follows.

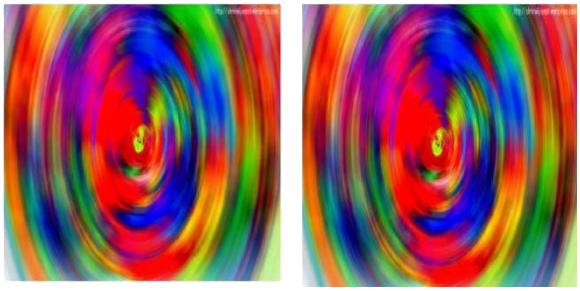


Figure 12: Comparing the popular Lena.jpg (i) Original Cover Image (ii) Stego Image

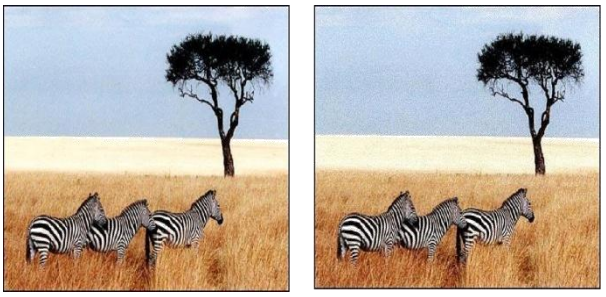


Figure 13: Comparing the popular Zebra.jpg Original Cover Image and Stego Image

Table 5:Comparing the result obtained by putting data in JPG images

Image	Image Dimension	Radius Considered	No of pixels used	Audio bytes Hidden in one Circle
Pic1.jpg	1920x1080	530	2000	2000
Lena.jpg	225x225	110	700	700
Zebra.jpg	400x300	148	1000	1000

Table 6: Comparing the change after altering pixels of one circle

Image	MSE	PNSR
Pic1.jpg	3.0439 e-009	133.2966
Lena.jpg	4.7681 e-007	111.3474
Zebra.jpg	1.1611 e-007	117.4821

Pic1.jpg has greater dimensional area compared to others. The largest circle on this image can store 2KB data. The next circle selected on same image would have lesser radius, thus lesser storage capacity. The above results show that larger the image dimension, larger is the radius. And larger radius means more number of pixels available for putting the secret data. The dimension of the cover image should be large, if large circles are to be used. The dimension doesn't matter much when smaller circles are used. The preferred approach would be using circles of random radius, sometimes outer most, sometimes inner most, rather than smallest to largest or largest to smallest. This would make intruder difficult to understand the sequence of circles, which comes after which one. Thus, efficiency would increase at no extra cost.

Analysis

- In our base paper "Hide and Seek" [15] by Dr. Emad S. Othman, suggested an algorithm in which only 3-bit LSB replaced by audio bits in color component of one pixel while in our approach 8 bits can be embedded in RGB components of one pixel.
- Base paper has no comparison with various file formats of same image as cover, we have compared efficiency with .bmp, .jpg, .png and .tiff file formats and we found that JPEG is best suited image as cover.
- Embedding 3 bits in one pixel data may lead 30-50%

changes in pixel data while in our approach where 8 bits embedded in one pixel data also probability of 35-50% changes in pixel data, but we can hide more audio data in cover image. By this approach we can increase the capacity of embedding audio by 267%.

V. CONCLUSION

The work presents an approach to store audio data in image file by exploiting the concept of pixels over a circle. The secrecy is increased manifold by random selection of circle radius and starting point of the circle, thus making it difficult to track the presence and sequence of data. Even if distortion in image is found, intruder cannot find which data circle comes first. The secrecy can be increased by adding encryption of audio binary data before embedding in cover image. The limitation found is that a lot of area near the corners of image is not utilized, which is less prone to human vision. Few advantages reached by the proposed work are:

- More than one secret message can be embedded in a cover image. Every different message will have different key (radius). This feature could be very useful when there are more than one receivers of audio message, and each of them need to receive their own audio message.
- Random selection of radius (key 1) makes the sequence of data random. So the intruder cannot find which data comes first.
- The use of key 2 makes it even difficult to find, from where the data starts over a circle, as circle can not otherwise have a starting or an ending point.
- Being based on the LSB method, it is simple, neat and easy to implement.
- Using a hazy image prevents the carrier from visuals attacks. Changes are undetectable with the naked eye.
- As there is least change in color value, it is immune to attacks by comparing histograms. The frequency of appearance of colors in the steganographic image is very similar to that of the cover image.
- In experiment, only 35% LSBs were changed when substituted with data.
- Using arrays (matrix) for storing and modifying data, makes the structure of algorithm simple.
- Stego image is more secured from cutting and cropping, as corners of the image do not have any data.

VI. REFERENCES

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

[2] U. Rakesh Rao, Jayashree C. Nidagundi, “An Encrypted Image Audio Steganography”, International Journal of Research in Image, Video and Signal Processing –IJRIVSP Vol 01, Issue 01, April 2012

[3] Sara Natanj, Seyed Reza Taghizadeh, Kahje-Nassir Toosi, “Current Steganography Approaches: A survey”, International journal of advance research in Computer Science and Software Engineering, December 2011

[4] ZaidoonKh. AL-Ani, A.A.Zaidan, B.B.Zaidan, Hamdan.O.Alanazi, “Overview: Main Fundamentals for Steganography”, Journal of Computing, Volume 2, Issue 3, March 2010

[5] Budda Lavanya1, VittapuSravankumar; “Combination of Cyphertext and Audio Steganography Technique for Secrete Communication”, International Journal of Emerging Technology and Advanced Engineering, Volume 2, December 2012

[6] BhavsarJaimin H, Imran Khan, “Techniques Of Steganography And Steganalysis”, Electronic copy available at: http://ssrn.com/abstract=2029407

[7] Nagham Hamid, AbidYahya, R. Badlishah Ahmad &Osamah M. Al-Qershi, “Image Steganography Techniques: An Overview”, International Journal of Computer Science and Security (IJCSS), Volume (6) : Issue (3) : 2012

[8] Suman Chaudhry, Kulwinder Singh, “A Brief Introduction of Digital Image Processing“, International Journal of Advances in Computing and Information Technology, February 2012, http://www.ijacit.com/articles/twelve/vol1issue1/vol1issue1/EIJACIT120010.pdf

[9] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn, “Information Hiding - A Survey”, Proceedings of the IEEE, special issue on protection of multimedia content, 87(7):1062{1078}, July 1999

[10] Juan Jose Roque, JesúsMaríaMinguet, “SLSB: Improving the Steganographic Algorithm LSB”, Universidad Nacional de Educación a Distancia (Spain), http://www.fing.edu.uy/inco/eventos/cibsi09/docs/Papers/CIBSI-Dia3-Sesion9(1).pdf

[11] Image File Formats, scanning tips of JPG, PNG, BMP, TIFF : Website: http://www.scantips.com/basics09.html

[12] Dr. Emad S. Othman, “Hide and Seek: Embedding Audio into RGB 24-bit Color Image Sporadically Using Linked List Concepts”, IOSR Journal of Computer Engineering (IOSRJCE) Volume 4, Issue 1 (Sep-Oct. 2012)

[13] M.I.Khalil, “Image Steganography: Hiding short Audio message within Digital Images”, JCS&T Vol. 11 No. 2, October 2011

[14] A. E.Mustafa, A.M.F.ElGamal, M.E.ElAlmi, Ahmed.BD, “A Proposed Algorithm For Steganography In Digital Image Based on Least Significant Bit”, Research Journal Specific Education, Faculty of Specific Education, Mansoura University, Issue No. 21, April. 2011

[15] Bin Li, Junhui He, Jiwu Huang, Yun Qing Shi, “A Survey on Image Steganography and Steganalysis”, Journal of Information Hiding and Multimedia Signal Processing, Ubiquitous International Volume 2, April 2011

[16] Rosziati Ibrahim, Teoh Suk Kuan, “Seganography Algorithm to Hide Secret Message inside an Image”, David publishing, Computer Technology and Application 2 (2011) 102-108