International Journal of Research in Advent Technology, Vol.4, Issue 7, July 2016
E-ISSN: 2321-9637
Available online at www.ijrat.org

Textured Based Steganography using LSB

Smt.Renuka Mathapati[1],Dr Jagadeesh Pujari[2],Prof Shrivatsa Koulgi[3]
Department of Computer Science[1],Department of Information Science[2],Department of Information Science[3]
E-mail : ren_friends@rediff.com[1], jaggudp@yahoo.co.in[2],shrivastakoulgi@gmail.com[3]

Abstract – This paper discusses a technique for implementing Steganography in a bitmap color image in RGB plane *using* LSB technique. The three planes are used for embedding text and also provide additional authentication for reconstruction.

Keywords–Steganography, RGB planes, Texture, Encoding, Decoding, LSB.

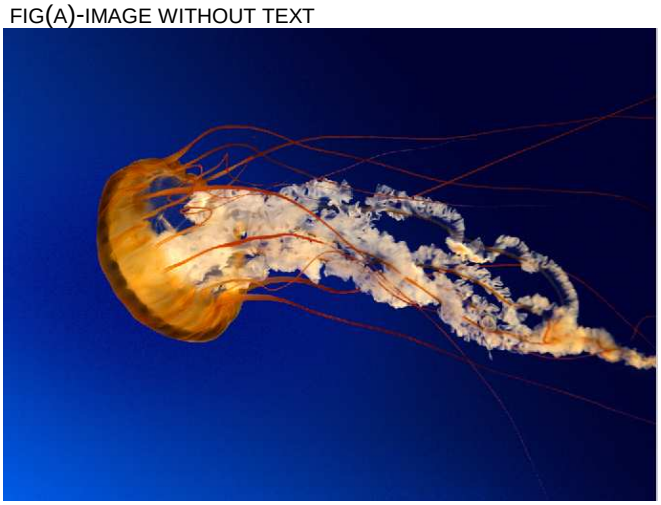
1. INTRODUCTION

In today's world communication has become an important aspect in the same way it is also necessary to maintain some secrecy for the message to be conveyed. In this paper Steganography [1] is concentrated to have secret message from unauthorized access that is why a method called texture is been implemented using LSB technique [2]. Basically in this method the image is read which is an RGB image converted to texture [3] where the pixel values are detected in a random fashion then the characters are hidden [4] from the last bit in the form of 3 bits in one plane, 1 bit in other plane and 1 more bit in third plane respectively. To be clearer the pixel value is separated into 2 nibbles were first four bits are used to hold the character without creating the disturbance in the image [5].

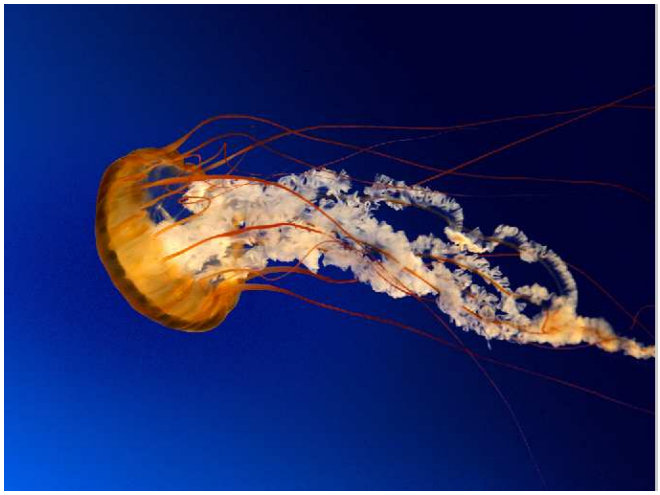
2. PROPOSED WORK

This work involves encryption and decryption of secret message which is been embedded within an RGB [6] colored image in which all the three planes are used to hide the text on the bases of texture in respective planes i.e R,G, B planes then the characters are embedded in any of the triplets by finding the length of the characters that are stored in the first 2 bit array in the form of 4 bit then the character is hidden into it using LSB technique. The character can he hidden in a pixel which can be a part any three planes either R or G or B pattern. Considering a color image [7] for embedding the characters within it based on nibble This work is done using LSB technique.
Following are the images i.e FIG (A) which shows image without text and FIG (B) which shows image with text or hidden text under image.

FIG(A)-IMAGE WITHOUT TEXT



FIG(B)-IMAGE WITH TEXT



3. ALGORITHM

1. Encoding

Step 1: Read color image

Step 2: Read text file

Step 3: Create a Filtered Image

Step 4: Extract pixel values of binary image where the pixel is white(1)

Step 5: Convert string to array of ASCII values

Step 6: Store 8 bits in bit array by shifting and anding

Step 7: If value exists then

Step 8: Put Length indicator in the beginning

Step9: Extract MSB Byte of size

Step10: Right shift 8 times, to get a number

Step11: Extract LSB byte of size

Step12:Store 8 bits in bit_array by shifting and anding

Step13: if value exists

Step14: Repeat two times, as one pixel will contain only one nibble of character

Step 15: First time,offset=0, take lower nibble

Step 16: Second time offset=4, take higher nibble

Step17: Clearing the last 2 bits

Step 18: Clearing the last 1 bit

Step19: Clearing the last 1 bit

Step 20: Store in first position

Step 21: Store in second position by shifting one time

Step 22: Display the Image

2. Decoding

Step 1: Read the color image

Step 2: Create a filtered Image

Step 3: Applying Threshold for the image to get the binary image. This threshold value can be varied

Step 4: Extract pixel values of binary image where the pixel is White(1)

Step 5: Fetch the last 2 bits

Step 6: Fetch the last 1 bit

Step 7: Fetch the last 1 bit

Step 8: One pixel stores one nibble data

Step 9: Get first nibble

Step 10: Get second nibble

Step11: Shift second nibble by 4 positions

Step12: Combine the two to get one byte of data

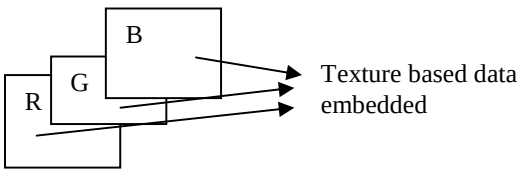
Step13: Extract length value

Step14: Else print the text in file

Step 15: Repeat till it reaches length

Step 16: If all conditions are true, obtain the embedded bit

Step 17: Separate image file and text file



3. Figures

4. RESULTS

During our implementation phase, we have tested our algorithm for different sets of images as well as text messages. For each and every normal bitmap images the proposed technique is working fine. Different parameters are been

applied to test the quality of the image like Mean Square Error (MSE),Peak Signal to Noise Ratio (PSNR) and Root Mean Square Error (RMSE) with their results when applied to the image. As the length of the text increases the results of these parameters also varies. Thus more number of characters can be embedded into the image depending upon its size.

IV a. Tabular representation of different parameters applied to image using LSB

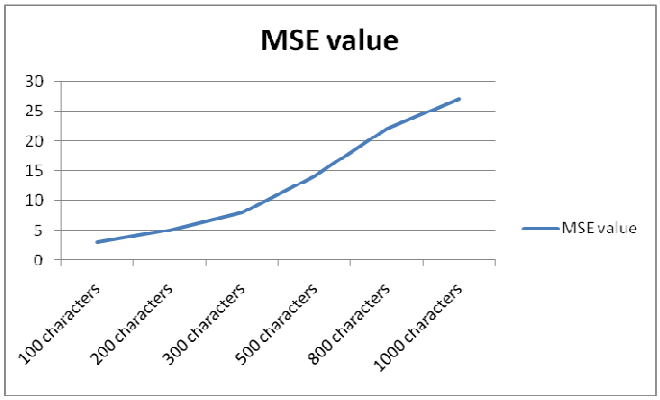
No.of Characters	MSE value
100 characters	3
200 characters	5
300 characters	8
500 characters	14
800 characters	22
1000 characters	27

No.of Characters	PSNR value
100 characters	43.39358676
200 characters	41.17509926
300 characters	39.13389944
500 characters	36.70351895
800 characters	34.7405725
1000 characters	33.85116166

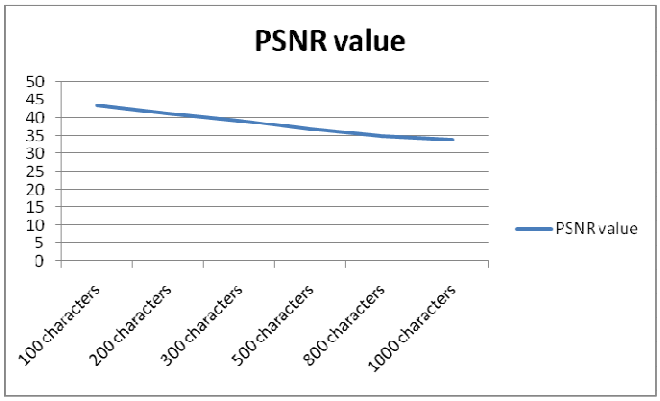
No.of Characters	RMSE value
100 characters	1.732050808
200 characters	2.236067977
300 characters	2.828427125
500 characters	3.741657387
800 characters	4.69041576
1000 characters	5.196152423

IV b. Graphical representation of different parameters based on results Using LSB

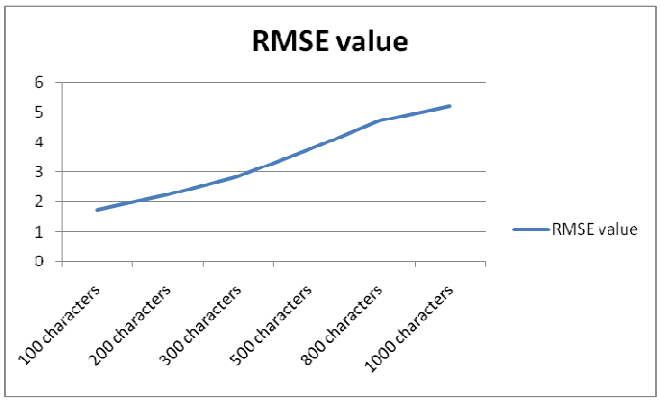
Mean Square Error



Peak Signal to Noise Ratio



Root Mean Square Error



5. CONCLUSION

In this paper, the major importance is given on the secrecy as well as the privacy of information. So, to obtain privacy we have used the concept of steganography. This algorithm is supposed to be more efficient as here from the resultant image it is difficult to guess the actual data that is hidden behind it. The triplets play an important role for hiding the text in either of one triplet based on the texture randomly. The technique LSB is compared with different parameters like MSE, PSNR and RMSE which gives the analysis of good quality of image that provide a clear idea about the best parameter in this technique. Since it is texture based whenever there is a corruption during transmission even then the data cannot be tracked.

REFERENCES

[1] S. Channalli and A. Jadhav, "Steganography an Art of Hiding Data", International Journal on Computer Science and Engineering, IJCSE, vol. 1, no. 3, (2009)

[2] K.H. Jung, K.J. Ha and K.Y. Yoo, "Image data hiding method based on multi-pixel differencing and LSB substitution methods", Proc. 2008 International Conference on Convergence and Hybrid Information Technology (ICHIT '08), Daejeon (Korea), (2008) August 28-30, pp. 355-358.

[3] A Texture-Based Method for Modeling the Background and Detecting Moving Objects Issue No.04 - April (2006 vol.28) pp: 657-662 Marko Heikkil? Matti Pietik?inen , IEEE

[4] S. C. Katzenbeisser. Principles of Steganography. in Information Hiding Techniques for Steganography and Digital Watermarking", S. Katzenbeisser and F. Petitcolas, Ed. London: Artech House, (2000), pp. 43-78

[5] P. Kruus, C. Scace, M. Heyman, and M. Mundy., A survey of steganography techniques for image files .Advanced Security Research Journal. [On line], 5(1),
(2003), pp. 41-52.

[6] Adnan Gutub, Ayed Al-Qahtani, Abdulaziz Tabakh, "Triple-A: Secure RGB Image Steganography Based on Randomization",International Conference on Computer Systems and Applications (AICCSA-2009), pp: 400-403, 10-13 May 2009.

[7] R.Amirtharajan, Sandeep Kumar Behera, Motamarri Abhilash Swarup, Mohamed Ashfaaq and John Bosco Balaguru Rayappan , "Colour Guided Colour Image Steganography " Universal Journal of Computer Science and Engineering Technology , 16-23, Oct. 2010, pp. 2219-2158.