Stretching the Limits of Image Steganography

Anshit Agarwal

Abstract-- *Sending encrypted messages has always been an interesting problem for the humankind. In a digital world, image steganography is used to hide secret messages inside an unsuspicious image. Steganography is often used together with cryptography to further strengthen the encryption. Steganography is a Greek word meaning "Covered Writing". This paper presents an overview of image steganography, the procedure of encrypting a graphic message in an image, the extent of information hiding possible and the error percentage related to the process. A new approach is proposed to maximize the amount of information that can be hidden. This paper also analyzed the significant drawbacks of existing methods and how our new approach extends the limits of image steganography.*

Index Terms-- Data Hiding, Encryption, Image Processing, Steganography, Bit Manipulation, Image within Image

1 Introduction

THE word steganography is derived from the Greek words "*stegos*" meaning "cover" and "*grafia*" meaning "writing" [1] defining it as "covered writing". The goal of image steganography is to transmit a message through an image over a communication channel where the existence of the message is hidden. Image files are ideal for steganographic transmission because of their large size. Minor Iterations in the pixels of an image go unnoticed to human eye and this becomes the basis of image steganography. Steganography has been a widely used technique in recent historical times and the present day.

An image is a collection of numbers that constitute different light intensities in different areas of the image [2]. This numeric representation forms a grid and the individual points are referred to as pixels. Most images on the Internet consists of a rectangular map of the image's pixels (represented as bits) where each pixel is located and its colour [3]. The smallest bit depth in current colour schemes is 8, meaning that there are 8 bits used to describe the colour of each pixel. Monochrome and greyscale images use 8 bits for each pixel and are able to display 256 different colours or shades of grey [4]. Digital colour images are typically stored in 24-bit files and use the RGB colour model, also known as true colour. All colour variations for the pixels of a 24-bit image are derived from three primary colours: red, green and blue, and each primary colour is represented by 8 bits. Thus in one given pixel, there can be 256 different quantities of red, green and blue, adding up to more than 16-million combinations, resulting in more than 16-million colours. Not surprisingly the larger amount of colours that can be displayed, the larger the file size [3].

This paper is organized as follows. Section 2 presents a brief description of image formation and information hiding in pixels. Section 3 describes the proposed technique for steganography. Section 4 gives an overview of the limits of the proposed technique. Finally, Section 5 highlights and discusses the arrived at conclusions.

2 Image Steganography

A 24-bit bitmap has 8 bits, representing each of the three: colour values (red, green, and blue) at each pixel. Each bit is represented either as a 0 or 1. The difference between 11111111 and 11111110 in the value for blue intensity is likely to be undetectable by the human eye. Hence, if the terminal recipient of the data is nothing but human visual system (HVS) then the Least Significant Bit (LSB) can be used for something else other than colour information. This technique can be directly applied on digital image in bitmap format as well as for the compressed image format like JPEG. In JPEG format, each pixel of the image is digitally coded using discrete cosine transformation (DCT)[5].

Example:

We take three consecutive pixels from top left corner of an image and extract the RGB values from each pixel.

Red Value	Green Value	Blue Value
124	124	127
65	124	68
255	73	67

The equivalent binary bit pattern of those pixels is: -

Red Value	Green Value	Blue Value
01111100	01111100	01111111
01000001	01111100	01000100
11111111	01001001	01000011

The ASCII equivalence of letter 'A' is 65, which further converted into binary gives us "01000001**.** Now, each bit of the binary number is copied serially (from the left hand side) to the LSB's of equivalent binary pattern of pixels, resulting in the bit pattern as:

Red Value	Green Value	Blue Value
0111110**0**	0111110**1**	0111111**0**
0100000**0**	0111110**0**	0100010**0**
1111111**0**	0100100**1**	01000011

The equivalent decimal pattern of those pixels is: -

Red Value	Green Value	Blue Value
124	125	126
64	124	68
254	73	67

The maximum error possible using the technique of LSB steganography is ±1 in the values of each component RGB. The maximum error percentage possible in each colour value is 0.392%. This is likely to be undetectable by the human eye.

3 The Proposed Method, 4-bit Image Steganography

Text in Image: This text can also be encrypted in a single pixel itself. Each bit of the letter 'A' can be copied to the last 4 bits of any two colour components of a pixel, resulting in the bit pattern as:

Red Value	Green Value	Blue Value
0111**0100**	0111**0001**	01111110

The equivalent decimal pattern of those pixels is: -

Red Value	Green Value	Blue Value
116	113	126

The maximum error possible using the technique of last 4 bits steganography is ±15 in the values of each component RGB. The maximum error percentage possible in each component value is 5.88%. Since, this is the maximum error possible, if the 4th last bit remains unchanged, the error will reduce to ±7 and if the 3rd last bit also remains unchanged, the error will further reduce to ±3.This is likely to be imperceptible or faintly perceptible by the human eye.

Image in Image: We consider a carrier image of **M × N** resolution and a the message image of **P × Q** resolution with the same aspect ratio. The numbers of bits in the message image are '**B'.** The maximum numbers of bits that can be encrypted using the 4-bit steganography in the carrier image are '**C'.**

B= Resolution of Image × Number of Colour Components × Bit Depth of each Component (1)

C= Resolution of Image × Number of Colour Components × 4 (2)

Example:
For a High Definition 'message' bitmap,
B= 1280 * 720 * 3 * 8
B= 22118400

For a Full High Definition 'carrier' bitmap,
C= 1920 * 1080 * 3 * 4
C= 24883200

Since the value of C is greater than the value of B, the 'message' image can be encrypted. This example also shows that a High Definition image can be encrypted into a Full High Definition image using the proposed method.

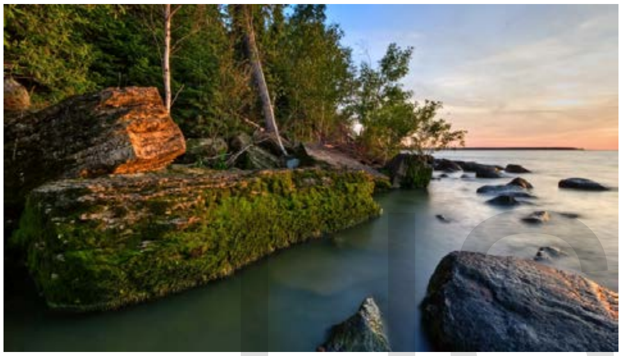

Figure 1: The Full High Definition Image used as carrier image

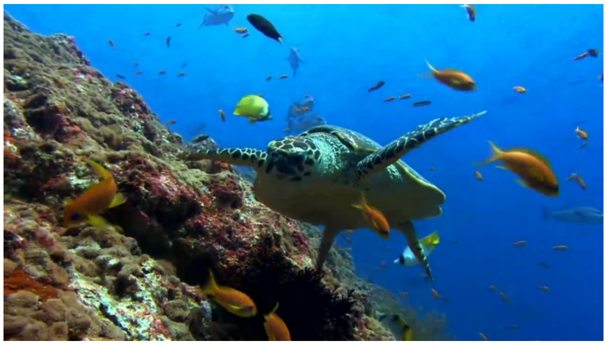

Figure 2: The High Definition Image used as message image

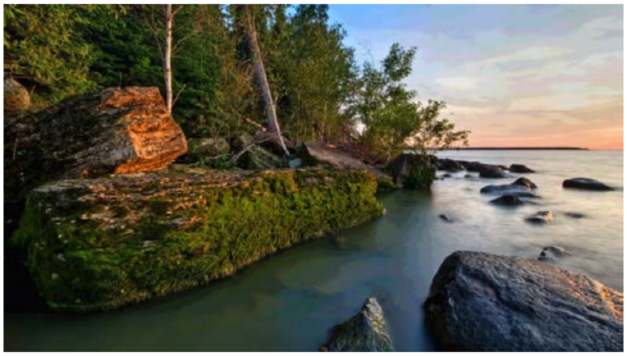

Figure 3: The Full High Definition Image that contains the message image in encrypted form

4 Limits of 4-bit Image Steganography

The limit of 4-bit stegnography can be determined with the following calculation. Consider both the images, carrier as well as the message image of same aspect ratio **A : R.**
Now,

$$B= A \div R \times Q \times Q \times 3 \times 8 \qquad (3)$$
$$C= A \div R \times N \times N \times 3 \times 4 \qquad (4)$$

The ratio **C ÷ B** should either equate or exceed 1 to make the encryption possible.

$$C \div B= N^2 \div (2 \times Q^2) \qquad (5)$$

Where N is the width of the carrier image and Q is the width of the message image. Below presented are two graphs that trace the increasing value of **C** with the width of the image for two common aspect ratios, 16:9 and 4:3.

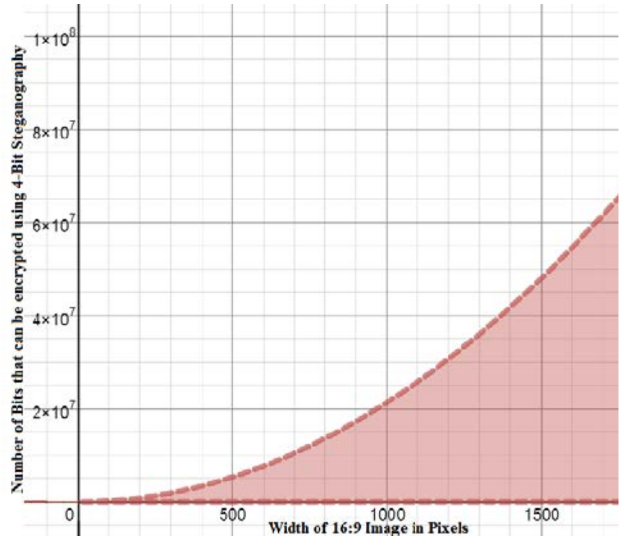

Figure 4: The Number of Bits that can be encrypted in 16:9 Image

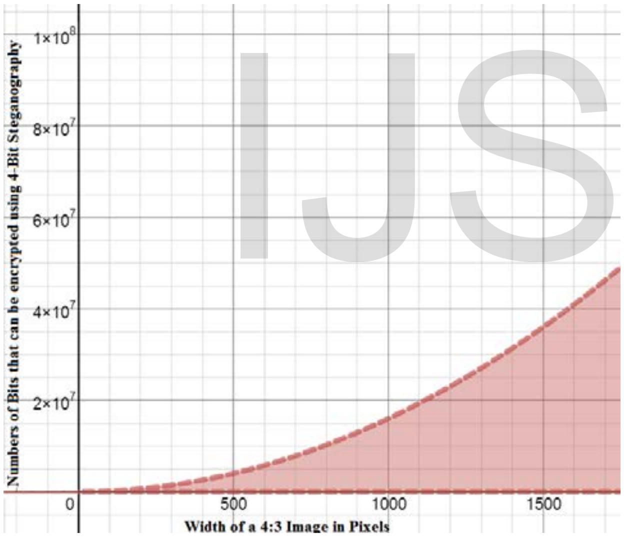

Figure 4: The Number of Bits that can be encrypted in 4:3 Image

5 Conclusions

4-bit stegnography is a process with both advantages and disadvantages. It allows hiding of high quality graphic content into another high quality graphic content. Although, the maximum error possible can deform the image through 5.88%, but in practical cases, this number rarely crosses 3.388%. The difference between the original image and the carrier image is minor and close to unnoticeable.

Steganography is a fascinating technique of hiding data. 4-bit stegnography further increases the applications by stretching the size of hidden messages that can be successfully encrypted. It will allow a much broader spectrum of uses; to encode .exe, .doc, .pdf, .mp3, etc.

References

[1]Research Scholar, Department of Information Technology, Singhania University, Jaipur, Rajasthan, INDIA.

[2] Johnson, N.F. & Jajodia, S., "Exploring Steganography: Seeing the Unseen", *Computer Journal*, February 1998

[3]"Reference guide: Graphics Technical Options and Decisions", http://www.devx.com/projectcool/Article/19997

[4]Owens, M., "A discussion of covert channels and steganography", *SANS Institute*, 2002

[5]Atallah M. Al-Shatnawi, "A New Method in Image Steganography with Improved Image Quality", *Applied Mathematical Sciences, Vol.6, 2012*, no. 79, 3907 - 3915

[6]Reddy V.L., Subramanyam A., Reddy P.C., "Implementation of LSB Steganography and its Evaluation for Various File Formats", Int. J. Advanced Networking and Applications Volume: 02, Issue: 05, Pages: 868-872 (2011)

The Author

Anshit Agarwal is a 2nd year undergraduate student in department of Computer Science and Engineering from Jaypee Institute of Information Technology, Noida. He is an avid programmer. His fields of interest are Image processing, Computer Vision and Artificial Intelligence.

Email: a.anshit@gmail.com