Comparison of LSB Steganography in BMP and JPEG Images

Eltyeb E. A bed Elgabar

Abstract-* *The literature on information security has to consider innovative and continuously developing ways of protecting data from infiltration taking into consideration the speed of the processors and the cryptanalysis. New steganographical technologies have been created to provide security with or without data encryption including data hiding. Etymologically, the term 'steganography', which means 'covered writing', originates from the Greek words "stegos" (cover), and "grafia" (writing). Steganography operates to conceal a secret message rooted in various forms of media including image, video, audio, and text. The domain of information hiding utilizes several algorithms. The easiest and widely known technique is Least Significant Bit (LSB).

This paper compares and analyses Least Significant Bit algorithm using the cover object as an image with a focus on two types: BMP and JPEG. The comparison and analysis are done with respect to a number of criteria to understand their strengths and weaknesses.

Index Terms—Robustness, Steganalysis, Steganography, Steganographic, Unsuspicious.

I. INTRODUCTION

Steganography (from Greek steganos, or "covered," and graphie, or "writing") is the hiding of a secret message within an ordinary message and the extraction of it at its destination. Steganography takes cryptography a step farther by hiding an encrypted message so that no one suspects it exists. Ideally, anyone scanning your data will fail to know it contains encrypted data. Also known as art and science of hiding information by embedding messages within other, seemingly harmless message. Steganography means "covered writing" in Greek [2]. The steganography goal is to hide the presence of a message within another message called cover message, so steganography can be seen as the complement of cryptography whose goal is to hide the content of a message.

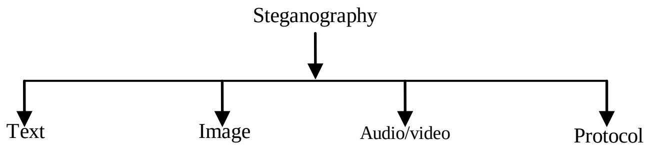

Fig.1 *Categories of Steganography*

A. Types of Steganography

Steganography can be classified into various types, depending upon the cover medium used. Cover medium may be text, image or audio or video file.Hence steganography can be said to occur in three major types [4]:

[1] Text Steganography.
[2] Image Steganography.
[3] Audio Steganography.
[4] Video Steganography.

B. Basic terms

- *Cover-object*, *c*: the original object where the message has to be embedded. Cover-text, cover-image,
- *Message*, *m*: the message that has to be embedded in the cover-object. It is also called stego-message or in the watermarking context mark or watermark.
- *Stego-object*, *s*: The cover object, once the message has been embedded.
- *Stego-key*, *k:* The secret shared between A and B to embed and retrieve the message

C. *The steganographic process*

- Embedding function, E: is a function that maps the tripled cover-object c, message m and stego-key k to a stego-object s. $E(c, m, k) = s$ (1)
- Retrieving function, D: is a mapping from s to m using the stego-key k. $D(s, k) = m$ (2)
- A secret key steganographic system [12] can be defined as the quintuple$\delta = < C, M, K, E, D >$ (3) where C is the set of possible cover-objects, M is the set of messages with $|C| \geq |M|$, K the set of secret keys,$E: C \times M \times K \rightarrow C$ and $D: C \times K \rightarrow M$ (4) with the property that $D(E(c, m, k), k) = m$ (5)
- for all $m \in M, c \in C$ and $k \in K$. (6)

II. IMAGE

A. Image definitions

To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image [5]. 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 color [9]. These pixels are displayed horizontally row by row. The number of bits in a color scheme, called the bit depth, refers to the number of bits used for each pixel [11]. The smallest bit depth in current color schemes is 8, meaning that there are 8 bits used to describe the color of each pixel [11]. Monochrome and grayscale images use 8 bits for each pixel and are able to display 256 different colors or shades of grey. Digital color images are typically stored in 24-bit files and use the RGB color model, also known as true color

Manuscript Received November, 2013.

Eltyeb E. Abed Elgabar, Information Technology, King Abdul Aziz University , Jeddah, Saudi Arabia.

[11]. All color variations for the pixels of a 24-bit image are derived from three primary colors: red, green and blue, and each primary color is represented by 8 bits [5]. 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 colors [11]. Not surprisingly the larger amount of colors that can be displayed, the larger the file size [9].

B. Image format

There are several types of image file formats that can be used for steganography such as BMP, JPEG, TIFF, GIF and PNG; each has certain advantages and disadvantages for hiding messages.

B.1 Bitmap images (BMP)

The letters “BMP” stand for “bitmap”, a file format best suited for “line art”. Bitmap images are introduced by Microsoft to be a standard image file format between users of their Windows operating system. The file format is now supported across multiple file systems and operating systems, but is being used less and less often. A key reason for this is the large file size, resulting from poor compression and verbose file format. This is, however, an advantage for hiding data without raising suspicion. To understand how bitmap images can be used to conceal data, the file format must first be explained. A bitmap file can be broken into two main blocks, the header and the data. The header, which consists of 54 bytes, can be broken into two sub-blocks. These are identified as the Bitmap Header, and the Bitmap Information. Images which are less than 16bit have an additional sub-block within the header labeled the Color Palette [12].

B.1.1 General BMP format properties

- A bitmap format that can be uncompressed, or compressed with RLE
- BMP files are.
  - In 1-bit black and white.
  - 8-bit greyscale.
  - 16-, 24- or 32-bit RGB color.
  - or 4- or 8-bit indexed color .
- BMP files don’t support CMYK color.
- Transparency is supported for individual pixels as in GIF files.
- Alpha channels are supported in new versions of BMP.

B.2 Joint Photographic Experts Group (JPEG)

The term actually stands for "Joint Photographic Experts Group," because that is the name of the committee that developed the format. But you don't have to remember that because even computer nerds will think you're weird if you mention what JPEG stands for. Instead, remember that a JPEG is a compressed image file format. JPEG images are not limited to a certain amount of color, like GIF images are. Therefore, the JPEG format is best for compressing photographic images. So if you see a large, colorful image on the Web, it is most likely a JPEG file. While JPEG images can contain colorful, high-resolution image data, it is a lossy format, which means some quality is lost when the image is compressed. If the image is compressed too much, the graphics become noticeably "blocky" and some of the detail is lost. Like GIFs, JPEGs are cross platform, meaning the same file will look the same on both a Mac and PC.

B.2.1 General JPEG format properties

- Are commonly used for photo.
- Can be compressed to a smaller size.
- JPEG files allow only 8 - 24-bit indexed color.
- JPEG files use lossy compression.

Table 1: Comparison of BMP & JPEG Images.

	BMP	JPEG
File types	Windows Bitmap or a map of bits	Joint Photographic Experts Group
File Suffix	.BMP	. JPG, JPEG
File Size	Larger	Small
Resolution	Medium	High
Support Color		16 Million Color
Complexity	Very Simplistic	Quite Complex
Ideal for	Icons and Small Images	Photo
Color Depth	1-32 bit color	8-24 bit color
Compression algorithms	Lossless	lossy

III. JPEG STEGANOGRAPHY

There are two broad categories of image-based steganography that exist today: frequency domain and spatial domain steganography. The first digital image steganography was done in the spatial domain using LSB coding (replacing the least significant bit or bits with embedded data bits) [30]. Since JPEG transforms spatial data into the frequency domain where it then employs lossy compression, embedding data in the spatial domain before JPEG compression is likely to introduce too much noise and result in too many errors during decoding of the embedded data when it is returned to the spatial domain. These would be hard to correct using error correction coding. Hence, it was thought that steganography would not be possible with JPEG images because of its lossy characteristics. However, JPEG encoding is divided into lossy and lossless stages [23]. DCT transformations to the frequency domain and quantization stages are lossy, whereas entropy encoding of the quantized DCT coefficients (which we will call the JPEG coefficients to distinguish them from the raw frequency domain coefficients) is lossless compression. Taking advantage of this, researchers have embedded data bits inside the JPEG coefficients before the entropy coding stage.

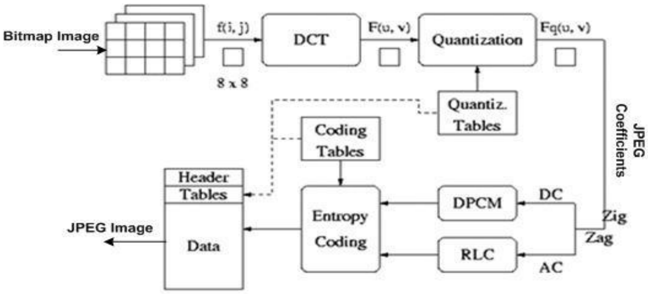

Fig.2 Most popular image format used [27].

IV. OVER VIEW OF LSB ALGORITHM

A digital image consists of a matrix of color and intensity values. In a typical gray scale image, 8 bits/pixel are used. In a typical full-color image, there are 24 bits/pixel, 8 bits assigned to each color components.

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image [5]. The least significant bit in other words, the 8th bit of some or all of the bytes inside an image is changed to a bit of the secret message. When using a 24-bit image, a bit of each of the red, green and blue color components can be used, since they are each represented by a byte. In other words, one can store 3 bits in each pixel. An 800 × 600 pixel image, can thus store a total amount of 1,440,000 bits or 180,000 bytes of embedded data [9]. For example a grid for 3 pixels of a 24-bit image can be as follows:

(00101101 00011100 11011100)
(10100110 11000100 00001100)
(11010010 10101101 01100011)

When the number 200, which binary representation is 11001000, is embedded into the least significant bits of this part of the image, the resulting grid is as follows:

(0010110**1** 0001110**1** 1101110**0**)
(1010011**0** 1100010**1** 0000110**0**)
(1101001**0** 1010110**0** 0110001**1**)

Although the number was embedded into the first 8 bytes of the grid, only the 3 underlined bits needed to be changed according to the embedded message. On average, only half of the bits in an image will need to be modified to hide a secret message using the maximum cover size [10]. Since there are 256 possible intensities of each primary color, changing the LSB of a pixel results in small changes in the intensity of the colors. These changes cannot be perceived by the human eye - thus the message is successfully hidden. With a well-chosen image, one can even hide the message in the least as well as second to least significant bit and still not see the difference [5].

In the above example, consecutive bytes of the image data from the first byte to the end of the message are used to embed the information. This approach is very easy to detect. A slightly more secure system is for the sender and receiver to share a secret key that specifies only certain pixels to be changed. Should an adversary suspect that LSB steganography has been used, he has no way of knowing which pixels to target without the secret key [6]. In its simplest form, LSB makes use of BMP images, since they use lossless compression. Unfortunately to be able to hide a secret message inside a BMP file, one would require a very large cover image.

The advantage of LSB embedding is its simplicity and many techniques use these methods [5]. LSB embedding also allows high perceptual transparency. However, there are many weaknesses when robustness, tamper resistance, and other security issues are considered. LSB encoding is extremely sensitive to any kind of filtering or manipulation of the stego-image. Scaling, rotation, cropping, addition of noise, or lossy compression to the stego-image is very likely to destroy the message. Furthermore an attacker can easily remove the message by removing (zeroing) the entire LSB plane with very little change in the perceptual quality of the modified stego-image.

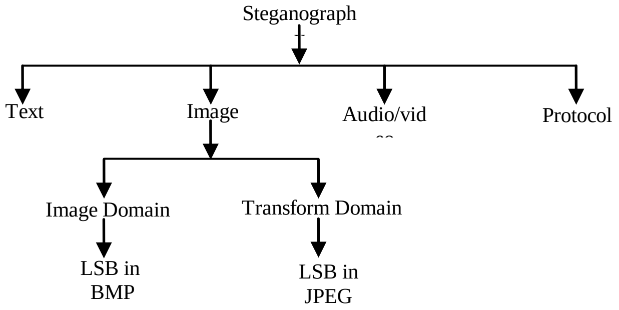

Fig.3 Categories of Image Steganography

A. Advantages of LSB

1. Major advantage of the LSB algorithm is it is quick and easy.
2. There has also been steganography software developed which work around LSB color alterations via palette manipulation.
3. LSB insertion also works well with gray-scale images

B. The LSB Algorithm

1. Select *cover-object* (BMP/JPEG) *c* as an input.
2. Encode the *c* in binary [16].
3. The Secret Message, *m*.
4. Encode the *m* in binary [16].
5. Choose one pixel of the *c* randomly.
6. Use a pixel selection to hide information in the *c* .
7. Save the new image (*Stego-object*) *s* .

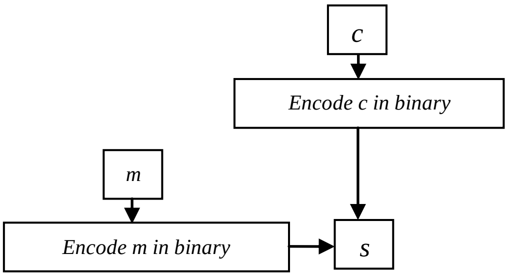

Fig.4 the LSB Algorithms

C. LSB in BMP

Since BMP is not widely used the suspicion might arise, if it is transmitted with an LSB stego. When image are used as the carrier in Steganography they are generally manipulated by changing one or more of the bits of the byte or bytes that make up the pixels of an image. The message can be stored in the LSB of one color of the RGB value or in the parity bit of the entire RGB value. A BMP is capable of hiding quite a large message. LSB in BMP is most suitable for applications, where the focus is on the amount of information to be transmitted and not on the secrecy of that information. If more number of bits is altered, it may result in a larger possibility that the altered bits can be seen with the human eye. But with the LSB the main objective of Steganography is to pass a message to a receiver without an intruder even knowing that a message is being passed is being achieved.

D. LSB in BMP LSB in JPEG

The most commonly used method to embed a bit is LSB embedding, where the least significant bit of a JPEG coefficient is modified in order to embed one bit of message. Once the required message bits have been embedded, the modified coefficients are compressed using entropy encoding to finally produce the JPEG stego image. By embedding information in JPEG coefficients, it is difficult to detect the presence of any hidden data since the changes are usually not visible to the human eye in the spatial domain. During the extraction process, the JPEG file is entropy decoded to obtain the JPEG coefficients, from which the message bits are extracted from the LSB of each coefficient.

LSB embedding [22], [23], [24] is the most common technique to embed message bits DCT coefficients. This method has also been used in the spatial domain where the least significant bit value of a pixel is changed to insert a zero or a one. A simple example would be to associate an even coefficient with a zero bit and an odd one with a one bit value. In order to embed a message bit in a pixel or a DCT coefficient, the sender increases or decreases the value of the coefficient/pixel to embed a zero or a one. The receiver then extracts the hidden message bits by reading the coefficients in the same sequence. And decoding them in accordance with the encoding technique performed on it. The advantage of LSB embedding is that it has good embedding capacity and the change is usually visually undetectable to the human eye. If all the coefficients are used, it can provide a capacity of almost one bit per coefficients using the frequency domain technique.

V. THE APPLYING AND EVALUATION

A. The original image (before hiding)

Fig.5a *BMP Image*

Fig.5b *JPEG Image*

Fig.6a *BMP Image*

Fig.6b *JPEG Image*

Table 2*: Properties of BMP& JPEG Images*

Name	**BMP$_{(a)}$**			**JPEG$_{(b)}$**		
	Size **MB**	*Dimension* **X*Y**	*Depth* **bpp**	*Size* **MB**	*Dimension* **X*Y**	*Depth* **bpp**
Fig.5	14.06	2560*1920	24	1.64	2560*1920	24
Fig.6	1.37	800*600	24	0.08	800*600	24

B. the image after hiding

Fig.7a *BMP Image*

Fig.7b *JPEG Image*

Fig.8a *BMP Image*

Fig.8b *JPEG Image*

Table 3: Comparison of LSB for BMP & JPEG Images

	BMP	**JPEG**
Efficient when amount of data reasonable	High	Medium
Amount of embedded data	High	Low
Steganalysis detection	Low	Medium
Percentage Distortion less resultant image	High	Medium
Robustness against image manipulation	Low	Medium
Invisibility	High	High
Robustness against statistical attacks	Low	Medium
Independent of file format	Low	Low
Payload capacity	High	Medium
Unsuspicious files	Low	High

Table 4: Comparison of LSB for BMP & JPEG Images

	BMP	**JPEG**
Efficient when amount of data reasonable	2	1
Amount of embedded data	2	0
Steganalysis detection	0	1
Percentage Distortion less resultant image	2	1
Robustness against image manipulation	0	1
Invisibility	2	2
Robustness against statistical attacks	0	1
Independent of file format	0	0
Payload capacity	2	1
Unsuspicious files	0	2

(High = 2, medium =1 and low =0)

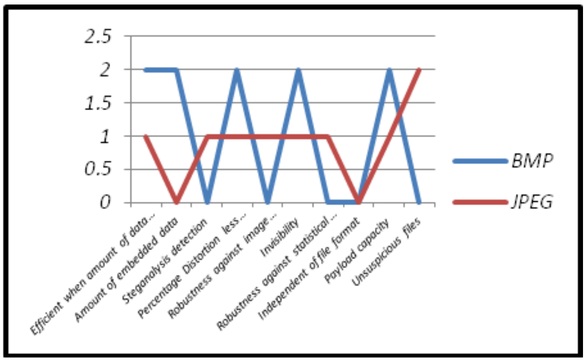

Fig.9 Comparison of LSB for BMP & JPEG Images

VI. CONCLUSION

In the images of the kind BMP, data type that can be embedded is large and the image is not distorted because of the ability of this kind of images for carrying amount of data without notice. For the image of kind JPEG we find very medium data embedded, as if resistance to statistical attacks, robustness against image manipulation is low, and when we increase the amount of data the image becomes distorted and is subject to discovery.

REFERENCES

1. Eltyeb E.Abed Elgabar, Haysam A. Ali Alamin, “Comparison of LSB Steganography in GIF and BMP Images ”, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-4, September 2013.
2. "Watermarking Application Scenaros and Related Attacks ", IEEE international Conference on Image Processing, Vol. 3, pp. 991 – 993, Oct. 2001.
3. Moerland, T., “Steganography and Steganalysis”, Leiden Institute of Advanced Computing Science, www.liacs.nl/home/ tmoerl/privtech.pdf.
4. Henk C. A. van Tilborg (Ed.), "Encyclopedia of cryptography and security", pp.159. Springer (2005).
5. Johnson, N.F. & Jajodia, S., “Exploring Steganography: Seeing the Unseen”, *Computer Journal*,February 1998.
6. Krenn, R., “Steganography and Steganalysis”, http://www.krenn.nl/univ/cry/steg/article.pdf
7. Pallavi Hemant Dixit, Uttam L. Bombale, " Arm Implementation of LSB Algorithm of Steganography", International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-3, February 2013.
8. “Reference guide:Graphics Technical Options and Decisions”, http://www.devx.com/ /Article/1997.
9. Artz, D., “Digital Steganography: Hiding Data within Data”, IEEE Internet Computing Journal, June 2001.
10. NXP & Security Innovation Encryption for ARM MCUs ppt.
11. Owens, M., “A discussion of covert channels and steganography”, SANS Institute, 2002.
12. “MSDN:About Bitmaps” <http://msdn.microsoft. com/library/default.asp?url=/library/enus/gdi/bitmaps_99ir.asp?frame =tru>, 2007,M Corporation.
13. V. Lokeswara Reddy, Dr. A. Subramanyam and Dr.P. Chenna Reddy, " 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).
14. Neeta Deshpande, Snehal Kamalapur and Jacobs Daisy, “Implementation of LSB steganography and Its Evaluation for Various Bits”, 1st International Conference on Digital Information Management, 6 Dec. 2006 pp. 173-178.
15. J. E. Boggess III, P. B. Nation, M. E. Harmon, “Compression of Colour Information In Digitized Images Using an Artificial Neural Network”, Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, Issue 23-27 May 1994 Page(s):772 - 778 vol.2.
16. Atallah M. Al-Shatnawi, "A New Method in Image Steganography with Improved Image Quality", Applied Mathematical Sciences, Vol. 6, 2012, no. 79, 3907 - 391.
17. Ze-Nian Li and Marks S.Drew, "Fundamentals of Multimedia, School of computing Science Simon Faster University, Pearsoll Education, Inc, 2004.
18. J. Fridrich, M. Goljan, and R. Du, “Detecting LSB steganography in color, and gray-scale images,” IEEE Multimedia, vol. 8, no. 4, pp. 22–28, Oct. 2001.
19. Priya Thomas," Literature Survey On Modern Image Steganographic Techniques", International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 5, May - 2013 ISSN: 2278-0181.
20. V. Lokeswara Reddy, Dr. A. Subramanyam, Dr.P. Chenna Reddy ,"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).
21. Roshidi Din and Hanizan Shaker Hussain, “The Capability of Image In Hiding A Secret Message”, Proceedings of the 6th WSEAS International Conference on Signal, Speech and Image Processing, September 2006.
22. D. Llamas, C. Allison, and A. Miller, \Covert channels in internet protocols: A survey," in Proceedings of the 6th Annual Postgraduate Symposium about the Convergence of Telecommunications, Networking and Broadcasting, 2005.
23. Neil R. Bennett, JPEG STEGANALYSIS & TCP/IP STEGANOGRAPHY, University of Rhode Island , 2009.
24. H. Wu, N. Wu, C. Tsai, and M. Hwang, “Image steganographic scheme based on pixel-value differencing and LSB replacement methods,” IEE Proceedings-Vision, Image and Signal Processing, vol. 152, no. 5, pp. 611–615, 2005.
25. R. Chandramouli and N. Memon, “Analysis of LSB based image steganography techniques,” in Image Processing, 2001. Proceedings. 2001 International Conference on, vol. 3, 2001.
26. Y. Lee and L. Chen, “High capacity image steganographic model,” IEE Proceedings-Vision, Image and Signal Processing, vol. 147, no. 3, pp. 288–294, 2000.
27. W. Pennebaker and J. Mitchell, JPEG still image data compression standard. Kluwer Academic Publishers, 1993.

AUTHOR PROFILE

Dr.Eltyeb Elsamani Abd Elgabar Elsamani, Assistant Professor(2009) in the Computer Science at Faculty of Computer Science and Information Technology, Information Technology Department - Khulais - King Abdul Aziz University- Jeddah - Saudi Arabia. Assistant Professor in the Computer Science at the Department of Computer Science, Faculty of Computer Science and Information Technology - Alneelain University - Khartoum - Sudan. . Main specialization is Information Security in particular and Encryption in specific. A member of the committee of Standard specifications for Computers Hardware and Peripherals in the National Information Center (NIC) - Khartoum -Sudan , member of Standard specifications for Network Hardware in the National Information Center (NIC) - Khartoum -Sudan, and member of Curriculum of information technology department - Faculty of Kamleen Ahlia- Gazera Sudan.