Hidden Data Transmission using Image Steganography

Dharmesh Mistry
Dept. of Computer Engineering
Dwarkadas J Sanghvi College of Engineering, India

Richa Desai
Dept. of Computer Engineering
Dwarkadas J Sanghvi College of Engineering, India

Megh Jagad
Dept. of Computer Engineering
Dwarkadas J Sanghvi College of Engineering, India

ABSTRACT
Steganography can be defined as the science of hiding data behind images in such a way that no one except the sender and receiver has any idea about the transfer of hidden message taking place. In this paper, light is thrown on the algorithm using LSB of the image pixels to store data and transfer in a hidden manner, such that it won't attract attention and the observers will be unaware of any data transmission taking place. The project involved implement and compare three algorithms and evaluate the results to find the most efficient algorithm.

General Terms
Security, Algorithms, Encryption, Steganalysis

Keywords
Steganography, steganalysis, cover image, steg image, LSB

1. INTRODUCTION
The term 'Steganography' hints to the art of "invisible" communication. Steganography is very different from cryptography, in the aspect that cryptography's chief aim is to protect the message from unauthorized attackers, whereas steganography strives to conceal the existence of the message itself and its transmission taking place. When a message is encrypted, it is visible to the whole world and also vulnerable to malicious attacks, where as if the data is hidden in an image, apart from the entities involved in the communication process, no third party will be able to sense this transmission of data. Hence, being more secure and robust, steganography finds a wider range of applications as compared to encryption. This technique of hiding information by embedding other, seemingly harmless messages with our message to be transferred, Steganography employs the replacement of bits that apparently are useless or not used, with bits that contain some important, invisible data within it. The type of data that can be hidden ranges from plain and cipher texts to images. A plethora of carrier file formats are available, however, images are used due to their high frequency and availability over the Internet. A number of different steganographic techniques are available for hiding information, some more complex than the rest. However, on the whole, every method has its own advantages as well as drawbacks. Different techniques can be used based on the varied requirements of different applications. One application may require containing and hiding really large data, whereas, on the other hand, some applications may need the process to be more discrete. Based on the requirements, appropriate

Steganographic techniques are selected and used. A major application as well as an advantage of steganography is that it can be used in a situation where the usual practice of encryption may not be feasible or may prove to be futile and fail. Steganography is also known be enhance the efficiency of encryption process. A message can be encrypted using the common encryption algorithms. Now, this encrypted message, also known as cipher text can be cracked by an expert hacker, who can find patterns in the message. Hence, encryption itself is prone to attacks. However, if this cipher text is hidden under an image and sent, even if the hacker happens to extract the cipher key from the image, he'll probably not understand anything from the decoded message as it's still encoded. Through this paper, a brief overview of the process of image steganography is given, along with its different techniques and their respective applications. Also, the criteria that a steganography algorithm must meet in order to be efficient and serve its purpose are given [1].

2. GENERAL CONCEPTS
This section introduces the basic terms and concepts associated with steganography, which one may come across. Starting off with the framework in which steganography is presented; the whole process is introduced with the help of an example of the prisoner's problem in which there are two inmates, Bob and Alice, who need to communicate in order to plan an escape. However, this communication isn't that easy, as all communication between them is examined by Wendy, the warden who'll not hesitate to individually confine them if she gets the slightest air of some discreet communication going on.

Now consider a specific example of this problem. Alice has a message m that she wishes to convey to Bob, secretly. For doing the same, Alice "embeds" her message, say m into a cover-object, represented by c, and obtains a stego-object called s. Now this stego-object, s obtained is then sent to Bob via the public channel. Till now, following definitions have been covered:

Cover-object: A cover object refers to the object which is used as the carrier into which, messages are embedded. Images, audio, video, html pages, file structures, etc. are the different types of objects that can be employed to embed messages into them.

Stego-object: A stego-object refers to the object which contains the message hidden. Once a cover object is selected, along with a message to be transmitted, the steganographer aims to produce a stego object which then carries the hidden message. In an ideal situation, this process of embedding and transferring the message remains hidden from Wendy. But usually, it is assumed that only the key which is used by the algorithm is kept under wraps, and not the algorithm. This assumption is also known as Kirchhoff's principle in the field of cryptography.

Wendy is completely aloof of the secret key that is shared between Alice and Bob, although she knows about the algorithm which might possible be used for embedding and sending out secret messages. It should be notes that this secret key can be anything like a password which triggers a pseudo-random number generator to pick out pixel locations in an image i.e. cover-object where data can be hidden.

Now, all the communication going on as well as all the messages are open to warden Wendy's access and examination, who can be either in an active or passive mode. In the passive mode, the warden just examines the messages superficially and tries to identify hidden messages, if any. In case it if found that a hidden message is being transmitted, the message may be discarded or any inhibitory action can be taken. On the other hand, when a warden is active, she can modify the message being transmitted even if she can't find anything suspicious, just to ruin any kind of secretive communication that might still be taking place between Bob and Alice. The extent to which the warden can modify the message depends totally on the cover object and the model being employed.

Consider the case with images. For images, the warden should be permitted to alter only till the extent where she doesn't modify the subjective visual quality of the image significantly. In an ideal case, it is assumed that warden Wendy doesn't make any changes to the stego-object. The difference between cover objects and stego-objects shouldn't be apparent to Wendy. She shouldn't be able to differentiate between the objects containing a secret message hidden within it, and those lacking it.

Here, the term steganalysis embodies the modus operandi which helps Wendy in identifying the dissimilarities between the cover objects and stego objects. Please note that here; Wendy is supposed to identify this distinction without having any idea about the secret key that is being shared by Bob and Alice and also without the knowledge of any secret algorithm that might have been employed to embed the secret message into the image.

Therefore, steganalysis can be categorized as a tedious problem. But, it shouldn't be missed out the fact that Wendy doesn't have to explore into the contents of the hidden message m or try to decipher it. All she has to do is determining the presence of a message hidden. This makes her task way simpler. The interest in exploring and analyzing different steganalysis techniques is due to the large scale availability of steganography techniques and tools and their high paced development. Recently, there has been in rise in the number of innovative and robust steganalysis techniques reported in literature. A number of them have proved to be quite efficient, with respect to specific embedding methods. These techniques are reviewed in the coming sections [2].

3. DIFFERENT TYPES OF STEGANOGRAPHY

3.1 Fragile Steganography

In Fragile Steganography, the file in which data is to be hidden is embedded, gets destroyed if it is modified. This technique isn't used to record the copyright holder of files, as it is extremely easy to remove. However, in cases where proving that the file is in its original state is necessary, for example: for legal purposes or as an evidence, this method is employed as the watermark wouldn't be present had the file been modified. It is generally observed that implementing Fragile Steganographic techniques is much simpler than implanting the robust ones.

3.2 Robust Steganography

In Robust Steganographic technique, data is embedded into files which aren't easy to detect or modify. Though no kind of mark is completely robust, a system is called robust if all efforts to remove the mark would modify the file to the extent that it no longer serves its original purpose or is of any use. Hence, it's better to hide the mark where it's easier to detect and remove.

Robust marking is further divided into two more categories. The first one is watermarking. Watermarks are used to detect the owner or creator of the file leaked. They are generally hidden in a way that it's extremely tedious to detect and eliminate them, and hence are termed as 'imperceptible watermarks. But, this isn't the case every time. At times, watermarks that are visible are used. They often form a kind of pattern on the original image. Such patterns are similar to use of watermarking in hard format. One such example is the watermarking found in the currency of Britain.

Watermarking is used to identify the owner of the file, who has the copyright. On the other hand, fingerprinting makes use of a distinct mode of identification for every person who possesses the file and hence, is authorised to use it. This fingerprint can be used to detect the violation of agreements if the file is found with anyone else apart from the owner.

The main difference between fingerprinting and watermarking is that, fingerprinting is used to identify the customer of a particular file whereas watermarking aims to find the owner who has the copyright. Fingerprints help in identifying those agents who have an illegal copy of the data, and hence prosecute them for violation of the terms of license agreement. Although fingerprinting should be used for production and distribution of objects in bulk, it is practically not possible to assign a unique fingerprint to every copy [3].

4. STEGANOGRAPHIC ALGORITHMS

There are four algorithms implemented here. Some of them filter the image before embedding data where as others don't. The common factor among them is that all of them involve the Least Significant Bit (LSB) algorithm for embedding the data to be hidden into the pixels of the cover image.

The Steganographic algorithms are as follows:

4.1 BlindHide

Using a lossless image like a Bitmap image and then substituting its least significant bits with the data to be hidden is the easiest way of data embedding. This substitution takes place line by line. However, this techniques isn't very secure as a malicious attacker might try to repeat this process multiple times to get access to the hidden data.

This technique is called 'BlindHide' as it simply hides the data in a blind fashion, and hence isn't very efficient at the same. It is possible that some part of the image gets totally distorted while remaining portion is as it is [4].

4.2 FilterFirst

Now, only say x least significant bits are replaced, and the y most significant bits are used for filtering. The range for x is [1,7] and y is x less than 8 i.e. (8-x). The total recovery of the hidden message is assured as the bits used for filtering aren't modified during embedding the data.

No extra data is required, like the original picture, and still makes sure that embedding and retrieval are done with the same pixels [5].

4.3 HideSeek

HideSeek algorithm involves the random distribution of message bits across the pixels of the cover image. HideSeek algorithm derives its name from a steganography tool of Windows 95, called "Hide and Seek" which uses a technique similar to our algorithm. A password generates a random

seed, and this seed is used to select the first position to embed information in it.

This random generation of positions keeps going on until the entire data is embedded. This algorithm is relatively difficult to crack, as every pixel combination needs to be tried in case the password isn't known. There is still scope for improvement in this algorithm as the pixels aren't evaluated if they're good for hiding data. This can be dealt with by checking which regions of the image are better to embed the data within [6].

5. TYPES OF FILTERS

5.1 Laplacian of Gaussian

Laplacian filters use derivative functions to detect regions in the image having drastic changes in color i.e. to detect a distinctly colored pixel in a uniformly colored block. Usually, the images are smoothened making use of Gaussian filter prior to applying the Laplacian filter, as the derivative filters are sensitive to noise. As a filter plus derivative, both actions are taking place, it's called a two-step process, the Laplacian of Gaussian (LoG) operation.

$$L(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

The approximate discrete convolution kernel can be found by a number of distinct ways. This kernel can approximate Laplacian's effect. An example of a kernel is as given below.

$$\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}$$

As the central peak is a negative value, this filter is known as negative Laplacian. To have a positive Laplacian, the signs need to be inversed. This won't change its meaning and hence, won't matter [7].

For protecting from noise, Gaussian filters are used for smoothing. Hence, simply encapsulating the Gaussian and the Laplacian into a single equation gives:

$$LoG(x,y) = -1/\pi\sigma^4 [1 - (x^2+y^2)/2\sigma^2] e^{-z}$$

Where $z = (x^2+y^2)/2\sigma^2$

5.2 Sobel

Sobel-Feldman operator, also known as Sobel is basically a discrete differentiator operator. It is used to calculate an approximate value of the gradient of the intensity function of an image. A gradient vector or a norm of that vector is generated by the Sobel operator at each point on an image. The Sobel operator isn't that expensive when it comes to calculations as it involves convolving of the image with a small, integer valued, and separable filter in both horizontal as well as vertical directions. However, the gradient approximation produced is comparatively crude, especially for high frequency variations found in the image [8].

In order to compute the approximations of derivatives, two kernels are used of size 3x3. These kernels are convolved with the original image and the calculations are made, one for vertical and horizontal each. Let Gx and Gy be two images containing horizontal and vertical derivative at every point and let A be the source image. The computation then will be as follows:

$$G_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} * A \qquad \text{and}$$

$$G_y = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix} * A$$

Here, * stands for the 2-D convolution operation.

The gradient is calculated along with smoothing as the kernels can be considered as a product and separated into a differentiating and an average kernel. Example: G_s can be

$$\begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} = \begin{matrix} 1 \\ 2 \\ 1 \end{matrix} \begin{matrix} -1 & 0 & 1 \end{matrix}$$

The x-coordinate is defined to be increasing towards the "right"-direction, whereas the y-coordinate is defined to be increasing along the "down"-direction. Gradient approximations for each point in an image can be combined to give the following value of gradient magnitude:

$$G = (G_x^2 + G_y^2)^{1/2}$$

This information help to compute the direction of the gradient

$$\Theta = \text{atan}^2(G_x, G_y)$$

Here, for a vertical edge, the value of Θ is 0 [9].

6. LEAST SIGNIFICANT BIT

Usual computer images nowadays utilize 24 bits to represent each pixel's color. The intensity of red part, green part and blue part are stored using eight bits, ranging from 00000000 to 11111111, thus having 256 distinct values. It is good to have a wide range of colors; however human eye isn't very efficient at distinguishing minor differences between close shades of colors. This very fact helps encapsulate information into the image. As eyes are incapable of telling the difference in values of two adjacent color, last binary digit or the LSB can be manipulated. Set the LSB as 0 to store the hidden bit value as 0, and as 1 to store hidden bit value as 1. This is very easy to implement with little modification of the Boolean bits. The maximum difference in the value of a color will be of a single unit. Due to the three color channels (Red, Green, Blue), three hidden bits of data can be contained in every pixel [10].

The message to hide is converted into its ASCII equivalent character and subsequently into binary digit. For example if the character "t", then as ASCII value for "t" is 116 and binary value for it is 1110100. Any image has pixel which are contributed from red, green and blue components and each pixel has numbers from the color components (for 24-bit bitmap image each of red, blue and green pixel has 8 bit). At the 8th bit of the color number, if least significant bits are changed, human visual system cannot identify a change in pixels, hence making it possible to substitute the message bits with image pixel bit. Consider the pixel value 10111011, and to store the data at the worst possible situation, in the least significant bit, the pixel changes to 10111010, examinations shows that HVS cannot identify this alteration. Hence, message is embedded into the least significant bits of color. Changing the LSB in a byte of an image is basically adding or subtracting one from the value represented by it. The conversion of data to byte format is the first step, followed by storing in a byte array. The message is embedded into the LSB position of each pixel.

The embedding process operates over the image, and embeds the message character into cover-image pixel by pixel at a time. Once all the characters of the message are embedded into the cover-image, the target character represented in bit by

101010, is inserted in the pixel of the cover-image immediately next to the one containing the last input character of the message. The target character is a special symbol and is called the Terminator Character. Because it is the last character that is embedded and after embedding the target character (101010), insertion process stops from next row onwards. This helps the decoding process to stop extracting of data from stego-image by informing that the target character signifies the end of the message [11].

7. BENCHMARK EVALUATIONS

Evaluated all three algorithms of Steganography and analyzing their Benchmarks, distinct values obtained for all tested attributes which are given as follows:

	Mean Squared Error	**Peak Signal to Noise Ratio**
BlindHide	1.69	345
HideSeek	1.35	432
FilterFirst	1.71	340

As given in the table, it is observed that HideSeek algorithm has the least mean squared error, which means that the cover image contains the minimum error, hence it's the closest to the original cover image. Also, the peak signal to noise ratio is highest for HideSeek, indicating that the proportion of noise present in the image is low. Thus, the HideSeek algorithm is the best among the three [11].

8. CONCLUSION

Steganography does not intend to take the place of cryptography but rather support and supplement it. Consider that a message is initially encrypted and then hidden with a steganographic method, it provides a double layer of protection and reduces the chances of the hidden message getting detected.

Steganography is still a fairly new and developing concept to people, though the case is different in the field of cryptography and secrecy.

Steganography definitely has a bright future and a large scope for further research and exploration. Much more sophisticated and efficient techniques of steganography and steganalysis are expected to come up in future [12].

9. ACKNOWLEDGMENTS

We take this opportunity to express our profound gratitude and deep regards to our professors for their guidance, monitoring and constant motivation throughout the course of our project. The encouragement and help given by them regularly has made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

We would like for believing in us and sanctioning the project. An honorable mention goes to the entire teaching and non-teaching staff for their cordial support, valuable information and guidance, which helped us in completing this project till design phase. This work would not have been possible without the support of these people.

We would also like to thank various laboratories for their invaluable guidance and immense faith in us without which the take-up of this project was impossible. Although there may be many who remain unacknowledged in this humble note of gratitude, there are none who remain unappreciated.

10. REFERENCES

[1] Arron Miller Thesis on Steganography

[2] Steganalysis- Detecting hidden information, forensic analysis 1014.

[3] Data-genetics Blog, March 12,2012

[4] Detecting LSB Steganography in Images Ankit Gupta, Rahul Garg.

[5] Lee, Y-K. ; Bell, G., Huang, S-Y., Wang, R-Z. and Shyu, S-J. (2009), "An Advanced Least Significant-Bit Embedding Scheme for Steganographic Encoding", PSIVT 2009, LNCS 5414, Springer, pp. 349–360.

[6] Chan, Chi-Kwong, and L. M. Cheng. (2004), "Hiding data in images by simple LSB substitution." Pattern Recognition Vol. 37, no. 3, pp. 469-474.

[7] M.W. Chao, C.H. Lin, C.W. Yu and T.Y. Lee, A high capacity 3D steganography algorithm, IEEE Transactions on Visualization and Computer Graphics, 15(2)(2009)274-284.

[8] C.C. Chang, P. Tsai and M.H. Lin, An adaptive steganography for index-based images using codeword grouping, Advances in Multimedia Information Processing-PCM,Springer, (3333)(2004)731-738.

[9] R. Böhme and A. Westfeld, Breaking cauchy model-based JPEG steganography with first order statistics, in: Proceedings of the European Symposium on Research in Computer Security, ESORICS 2004, Valbonne, France, 13th Sept. 2004, LNCS, vol.3193, pp.125-140.

[10] J. Fridrich, M. Goljan and D. Hogeg, steganalysis of JPEG images: Breaking the F5 algorithm, in: Proceedings of Information Hiding: 5th International Workshop, IH 2002 Noordwijkerhout, The

[11] Netherlands, LNCS, Springer, October 7-9, 2002, 2578/2003, pp. 310-323.

[12] Andreas Westfeld. F5-a steganographic algorithm. In Information Hiding, pages 289–302, 2001.