HIDING INFORMATION BY USING IMAGE STEGANOGRAPHY

ArunaBhadu,RajBalaTokas, Swati chug ,jyotima

M.Tech(Software Engineering)Engineering College

Bikaner, India

aruna7bhadu@gmail.com

MTech(computer Science)Swami Kesvanand Institute Of Technology , Jaipur

Rajtokas26@gmail.com

M.Tech(Software Engineering)Engineering College Bikaner, India

swatichug@gmail.com

***Abstract* - Steganographyis the art of hiding the fact that communication is taking place, by hiding information in other information.**

In this paper method is proposed to provide more security for the key information with the combination of image compression and data encryption method.This method requires less memory space and fast transmission rate because of image compression technique is applied.Many different carrier file formats can be used, but digital images are the most popular because of their frequency on the Internet.For hiding secret information in images, there exist a large variety of steganographic techniques some are more complex than others and all of them have respective strong and weak points. Someapplications may use absolute invisibility of the secret information, but others require a larger secret message to be hidden.This paper intends to give an overview of image steganography, its uses and technique andits provides better security for encrypted data and no distortion in the image quality.

***Keywords*- Steganography, image compression, cryptography, watermarking, fingerprinting**

I. INTRODUCTION

Internet is one of the most important factors of information technology and communicationhas been the security of information. One of the reasons that intruders can be successful is that most of the information they acquire from a system is in a form that they can read and comprehend. Intruders may reveal the information to others, modify it to misrepresent an individual or organization, or use it to launch an attack. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret.

Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique used to implement this, is called Steganography.

Steganography is a type of hidden communication that literally means "covered writing" (from the Greek words *steganoo*r "covered" and graphos or "to write").The goal of steganography is to hide an information message inside harmless cover medium in such a way that it is not possible even to detect that there is a secret message. It is the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. Today steganography is mostly used on computers with digital data being the carriers and networks being the high speed delivery channels. Steganography differs 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 . Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect 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.

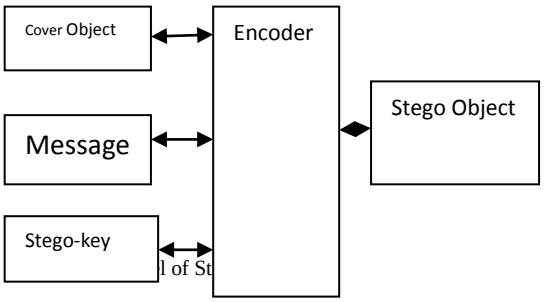
Information hiding is an emerging research area, which encompasses applications such as copyright protection for digital media, watermarking, fingerprinting, and steganography . All these applications of information hiding are quite diverse.

i)In watermarking applications, the message contains information such as owner identification and a digital time stamp, which usually applied for copyright protection.

(ii) Fingerprint, the owner of the data set embeds a serial number that uniquely identifies the user of the data set. This adds to copyright information to makes

it possible to trace any unauthorized used of the data set back to the user.

(iii) Steganography hide the secret message within the host data set and presence imperceptible.In those applications, information is hidden within a host data set and is to be reliably communicated to a receiver. The host Data set is purposely corrupted, but in a covert way, designed to be invisible to an informal analysis will only focus on information hiding using steganography approach .The basic model of steganography consists of Carrier, Message and Password. Carrier is also known as cover-object, which the message is embedded and serves to hide the presence of the message. Basically, the model for steganography is shown on Fig 1. Message is the data that the sender wishes to remain it confidential. It can be plain text, ciphertext, other image, or Password is known as stego-key, which ensures that only recipient who know the corresponding decoding key will be able to extract the message from a cover-object. The cover-object with the secretly embedded message is then called the stego-object .anything that can be embedded in a bit stream such as a copyright mark, a covert communication, or a serial number.



l of St

II. Image Steganography

Images are the most popular cover objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications .For these different image file formats, different steganographic algorithms exist. To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image. 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 .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 .The smallest bit depth in current color schemes is 8, meaning that there are 8 bits used to describe the color of each pixel . Monochrome and greyscale 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. All color variations for the pixels of a 24-bit image are derived fromthree primary colors: red, green and blue, and each primary color 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 colors. Not surprisingly the larger amount of colors that can be displayed, the larger the file size.

A. Image Compression

When working with larger images of greater bit depth, the images tend to become too large to transmit over a standard Internet connection. In order to display an image in a reasonable amount of time, techniques must be incorporated to reduce the image's file size. These techniques make use of mathematical formulas to analyze and condense image data, resulting in smaller file sizes. This process is called compression. In images there are two types of compression: lossy and lossless .Both methods save storage space, but the procedures that they implement differ. Lossy compression creates smaller files by discarding excess image data from the original image. It removes details that are too small for the human eye to differentiate, resulting in close approximations of the original image, although not an exact duplicate. An example of an image format that a use this compression technique is JPEG (Joint Photographic Experts Group) .Lossless compression, on the other hand, never removes any information from the original image, but instead represents data in mathematical formulas. The original image's integrity is maintained and the decompressed image output is bit-by-bit identical to the original image input. The most popular image formats that use lossless compression is GIF (Graphical Interchange Format) and 8-bit BMP (a Microsoft Windows bitmap file) Compression plays a very important role in choosing which steganographic algorithm to use. Lossy compression techniques result in smaller image file sizes, but it increases the possibility that the embedded message may be partly lost due to the fact that excess image data will be removed. Lossless compression though, keeps the original digital image intact without the chance of lost, although is does not compress the image to such a small file size.

B. *Image and Transform Domain*

Image steganography techniques can be divided into two groups: those in the Image Domain and those in the Transform Domain . Image – also known

as spatial – domain techniques embed messages in the intensity of the pixels directly, while for transform – also known as frequency – domain, images are first transformed and then the message is embedded in the image .Image domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation and are sometimes characterised as "simple systems" . The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format. Steganography in the transform domain involves the manipulation of algorithms and image transforms. These methods hide messages in more significant areas of the cover image, making it robust.Many transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression.

1) Image Domain

- Least Significant Bit:

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image. 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 embeddeddata [19]. 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:
(00101101 00011101 11011100)
(10100110 11000101 00001100)
(11010010 10101100 01100011)

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. 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. 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.

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. Nowadays, BMP images of 800 × 600 pixels are not often used on the Internet and might arouse suspicion . For this reason, LSB steganography has also been developed for use with other image file formats.

- LSB and Palette Based Images

Palette based images, for example GIF images, are another popular image file format commonly used on the Internet. By definition a GIF image cannot have a bit depth greater than 8, thus the maximum number of colors that a GIF can store is 256. GIF images are indexed images where the colors used in the image are stored in a palette, sometimes referred to as a color lookup table. Each pixel is represented as a single byte and the pixel data is an index to the color palette. The colors of the palette are typically ordered from the most used color to the least used colors to reduce lookup time.

GIF images can also be used for LSB steganography, although extra care should be taken. The problem with the palette approach used with GIF images is that should one change the least significant bit of a pixel, it can result in a completely different color since the index to the color paletteis changed. If adjacent palette entries are similar, there might be little or no noticeable change, but should the adjacent palette entries be very dissimilar, the change would be evident . One possible solution is to sort the palette so that the color differences between consecutive colors are minimized. Another solution is to add new colors which are visually similar to the existing colors in the palette. This requires the original image to have less unique colors than the maximum number of colors (this value depends on the bit depth used) . Using this approach, one should thus carefully choose the right cover image. Unfortunately any tampering with the palette of an indexed image leaves a very clear signature, making it easier to detect.

A final solution to the problem is to use greyscale images. In an 8-bit greyscale GIF image, there are

256 different shades of grey. The changes between the colors are very gradual, making it harder to detect.

2)Transform Domain

- JPEG compression

To compress an image into JPEG format, the RGB color representation is first converted to a YUV representation. In this representation the Y component corresponds to the for chrominance (or color). According to research the human eye is more sensitive to changes in the brightness (luminance) of a pixel than to changes in its color. This fact is exploited by luminance (or brightness) and the U and V components stand the JPEG compression by downsampling the color data to reduce the size of the file. The color components (U and V) are halved in horizontal and vertical directions, thus decreasing the file size by a factor of 2. The next step is the actual transformation of the image. For JPEG, the Discrete Cosine Transform (DCT) is used, but similar transforms are for example the Discrete Fourier Transform (DFT). These mathematical transforms convert the pixels in such a way as to give the effect of "spreading" the location of the pixel values over part of the image. The DCT transforms a signal from an image representation into a frequency representation, by grouping the pixels into 8 × 8 pixel blocks and transforming the pixel blocks into 64 DCT coefficients each .A modification of a single DCT coefficient will affect all 64 image pixels in that block. The next step is the quantization phase of the compression. Here another biological property of the human eye is exploited: The human eye is fairly good at spotting small differences in brightness over a relatively large area, but not so good as to distinguish between different between different strengths in high frequency brightness. This means that the strength of higher frequencies can be diminished, without changing the appearance of the image. JPEG does this by dividing all the values in a block by a quantization coefficient. The results are rounded to integer values and the coefficients are encoded using Huffman coding to further reduce the size.

- JPEG steganography

Originally it was thought that steganography would not be possible to use with JPEG images, since they use lossy compression which results in parts of the image data being altered. One of the major characteristics of steganography is the fact that information is hidden in the redundant bits of an object and since redundant bits are left out when using JPEG it was feared that the hidden message would be destroyed. Even if one could somehow keep the message intact it would be difficult to embed the message without the changes being noticeable because of the harsh compression applied. However, properties of the compression algorithm have been exploited in order to develop a steganographic algorithm for JPEGs. One of these properties of JPEG is exploited to make the changes to the image invisible to the human eye.

During the DCT transformation phase of the compression algorithm, rounding errors occur in the coefficient data that are not noticeable. Although this property is what classifies the algorithm as being lossy, this property can also be used to hide messages. It is neither feasible nor possible to embed information in an image that uses lossy compression, since the compression would destroy all information in the process. Thus it is important to recognize that the JPEG compression algorithm is actually divided into lossy and lossless stages. The DCT and the quantization phase form part of the lossy stage, while the Huffman encoding used to further compress the data is lossless.Steganography can take place between these two stages. Using the same principles of LSB insertion the message can be embedded into the least significant bits of the coefficients before applying the Huffmanencoding. By embedding the information at this stage, in the transform domain, it is extremely difficult to detect, since it is not in the visual domain.

3)*Image or Transform domain*

Some steganographic algorithms can either be categorized as being in the image domain or in the transform domain depending on the implementation.

Patchwork

Patchwork is a statistical technique that uses redundant pattern encoding to embed a message in an image. The algorithm adds redundancy to the hidden information and then scatters it throughout the image. A pseudorandom generator is used to select two areas of the image (or patches), patch A and patch B. All the pixels in patch A is lightened while the pixels in patch B is darkened. In other words the intensities of the pixels in the one patch are increased by a constant value, while the pixels of the other patch are decreased with the same constant value. The contrast changes in this patch subset encodes one bit and the changes are typically small and imperceptible, while not changing the average luminosity.

A disadvantage of the patchwork approach is that only one bit is embedded. One can embed more bits by first dividing the image into sub-images and applying the embedding to each of them. The advantage of using this technique is that the secret message is distributed over the entire image, so should one patch be destroyed, the others may still

survive. This however, depends on the message size, since the message can only be repeated throughout the image if it is small enough. If the message is too big, it can only be embedded once. The patchwork approach is used independent of the host image and proves to be quite robust as the hiddenmessage can survive conversion between lossy and lossless compression.

Spread Spectrum

In spread spectrum techniques, hidden data is spread throughout the cover-image making it harder to detect. A system proposed by Marvel et al. combines spread spectrum communication, error control coding and image processing to hide information in images. Spread spectrum communication can be defined as the process of spreading the bandwidth of a narrowband signal across a wide band of frequencies. This can be accomplished by adjusting the narrowband waveform with a wideband waveform, such as white noise. After spreading, the energy of the narrowband signal in any one frequency band is low and therefore difficult to detect. In spread spectrum image steganography the message is embedded in noise and then combined with the cover image to produce the stego image. Since the power of the embedded signal is much lower than the power of the cover image, the embedded image is not perceptible to the human eye or by computer analysis without access to the original image.

Table 1: Comparison of Image Steganography Algorithm

*-Depends on cover image used

	LSB in BMP	LSB in GIF	JPEG compression	Patchwork	Spread spectrum
Invisibility	High*	Medium*	High	High	High
Payload capacity	High	Medium	Medium	Low	Medium
Robustness against statistical attacks	Low	Low	Medium	High	High
Robustness against image manipulation	Low	Low	Medium	High	Medium
Independent of file format	Low	Low	Low	High	High
Unsuspicious files	Low	Low	High	High	High

sIII CONCLUSION

Although only some of the main image steganographic techniques were discussed in this paper, one can see that there exists a large selection of approaches to hiding information in images. All the major image file formats have different methods of hiding messages, with different strong and weak points respectively. Where one technique lacks in payload capacity, the other lacks in robustness. For example, the patchwork approach has a very high level of robustness against most type of attacks, but can hide only a very small amount of information. Least significant bit (LSB) in both BMP and GIF makes up for this, but both approaches result in suspicious files that increase the probability of detection when in the presence of a warden.Thus for an agent to decide on which steganographic algorithm to use, he would have to decide on the type of application he want to use the algorithm for and if he is willing to compromise on some features to ensure the security of others.

REFERENCES

[1]Moerland, T., "Steganography and Steganalysis", *Leiden Institute of Advanced Computing Science.*

[2]Jamil, T., "Steganography: The art of hiding information is plain sight", *IEEE Potentials*, 18:01, 1999

[3] Silman, J., "Steganography and Steganalysis: An Overview", *SANS Institut*e, 2001

[4]Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", *Communications of the ACM* 47:10, October 2004

[5] Marvel, L.M., Boncelet Jr., C.G. &Retter, C., "Spread Spectrum Steganography", *IEEE Transaction on image processing, 8:08,* 1999

[6] Artz, D., "Digital Steganography: Hiding Data within Data", *IEEE Internet Computing*

[7]Currie, D.L. & Irvine, C.E., "Surmounting the effects of lossy compression on Steganography", *19 th National InformationSystems Security Conferenc*e,1996