Steganography In Images

ABSTRACT:

In this paper, we aim to present a general introduction to steganography or data-hiding as it is sometimes just known. We then turn to data-hiding in images. When examining these data-hiding techniques, we bear in mind Bender's specifications, such as degradation of the cover data must be kept to a minimum, and the hidden data must be made as immune as possible to possible attack from manipulation of the cover data.

Steganography in images has truly come of age with the invention of fast, powerful computers. Software is readily available off the Internet for any user to hide data inside images. These softwares are designed to fight illegal distribution of image documents by stamping some recognisable feature into the image.The most popular technique is Least Significant Bit insertion, which we will look at. Also, we look at more complex methods such as masking and filtering, and algorithms and transformations, which offer the most robustness to attack, such as the Patchwork method which exploits the human eye's weakness to luminance variation.

we will take a brief look at steganalysis, the science of detecting hidden messages and destroying them. We conclude by finding that steganography offers great potential for securing of data copyright, and detection of infringers. Soon, through steganography,personal messages,files, all artistic creations, pictures, and songs can be protected from piracy.

INTRODUCTION:

Steganography, from the Greek, means covered, or secret writing, and is a long-practised form of hiding information. Although related to cryptography, they are not the same. Steganography's intent is to hide the existence of the message, while cryptography scrambles a message so that it cannot be understood.

More precisely,

> *``the goal of steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second secret message present."*

Steganography includes a vast array of techniques for hiding messages in a variety of media. Among these methods are invisible inks, microdots, digital signatures, covert channels and spread-spectrum communications. Today, thanks to modern technology, steganography is used on text, images, sound, signals, and more.

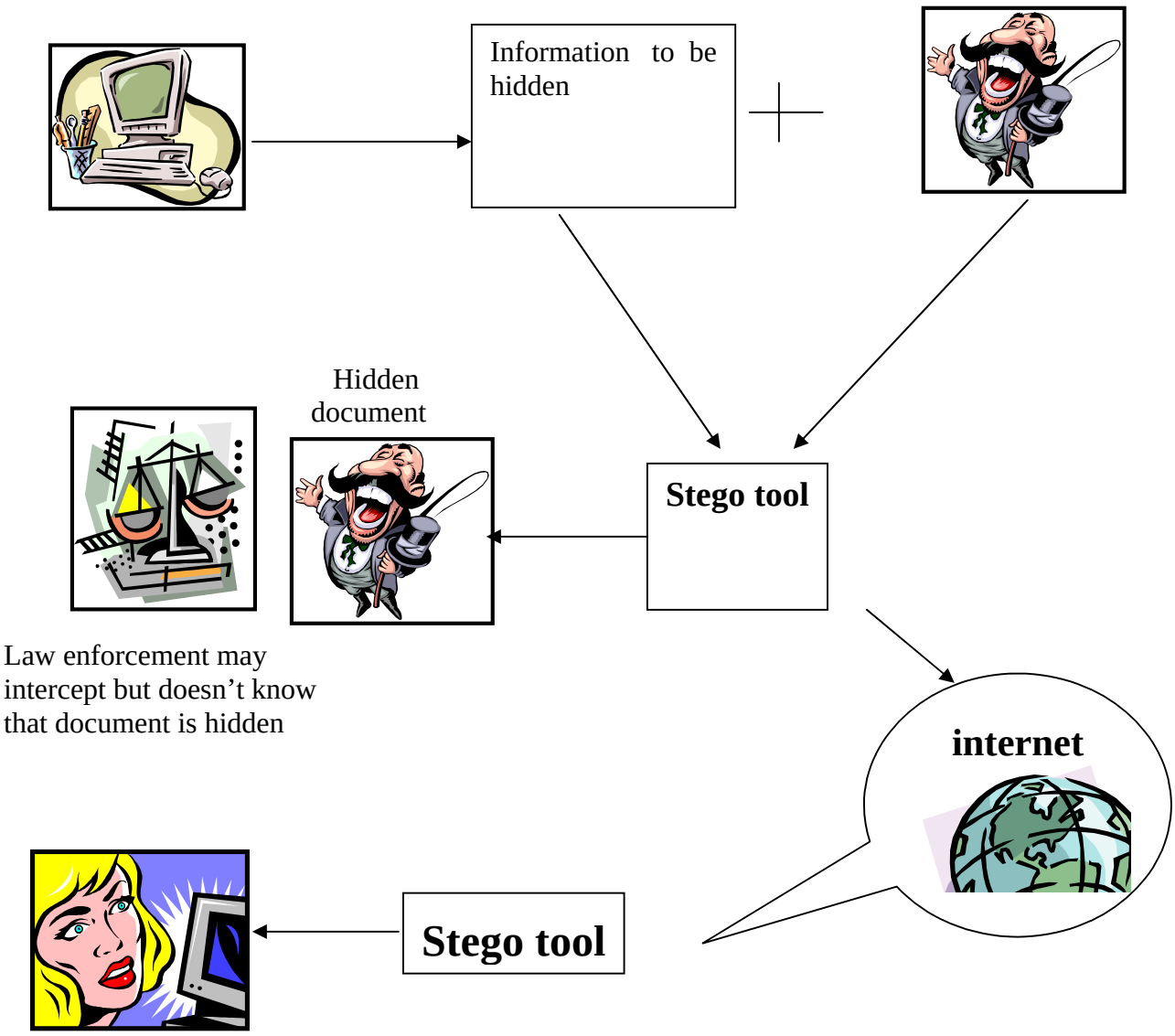
In the following sections we will try to show how steganography can and is being used through the media of images.

KERCKOFF PRINCIPLE:

In cryptography. This principle states that "the security of the system has to be based on the assumption that the enemy has full knowledge of the design and implementation details of the steganographic system". The only missing information for the enemy is a

short, easily exchangeable random number sequence, the secret key.

STEGANOGRAPHY DIAGRAMATIC FLOW:



User can get the hidden information using password

When embedding data, it is important to remember the following restrictions and features:

- The cover data should not be significantly degraded by the embedded data, and the embedded data should be as imperceptible as possible. (This does not mean the embedded data needs to be invisible; it is possible for the data to be hidden while it remains in plain sight.)
- The embedded data should be directly encoded into the media, rather than into a header or wrapper, to maintain data consistency across formats.
- The embedded data should be as immune as possible to modifications from intelligent attacks or anticipated manipulations such as filtering and resampling.
- Some distortion or degradation of the embedded data can be expected when the cover data is modified. To minimise this, error correcting codes should be used.
- The embedded data should be self-clocking or arbitrarily re-entrant. This ensures that the embedded data can still be extracted when only portions of the cover data is available. For example, if only a part of image is available, the embedded data should still be recoverable.

Steganography in Images:

In this section we deal with data encoding in still digital images. In essence, image steganography is about exploiting the limited powers of the human visual system (HVS). Within reason, any plain text, cipher text, other images, or anything that can be embedded in a bit stream can be hidden in an image. Image steganography has come quite far in recent years with the development of fast, powerful graphical computers, and steganographic software is now readily available over the Internet for everyday users.

IMAGES:

To a computer, an *image* is an array of numbers that represent light intensities at various points, or *pixels*. These pixels make up the image's *raster data*. An image size of 640 by 480 pixels, utilizing 256 colors (8 bits per pixel) is fairly common. Such an image would contain around 300 kilobits of data.

Digital images are typically stored in either 24-bit or 8-bit per pixel files. 24-bit images are sometimes known as *true colour* images. Obviously, a 24-bit image provides more space for hiding information; however, 24-bit images are generally large and not that common. A 24-bit image 1024 pixels wide by 768 pixels high would have a size in excess of 2 megabytes. As such, large files would attract attention were they to be transmitted across a network or the Internet. Image compression is desirable. However, compression brings with it other problems, that shall be explained shortly.

Alternatively, 8-bit colour images can be used to hide information. In 8-bit colour images, (such as GIF files), each pixel is represented as a single byte. Each pixel merely points to a colour index table, or *palette*, with 256 possible colours. The pixel's value, then, is between 0 and 255. The image software merely needs to paint the indicated colour on the screen at the selected pixel position.

If using an 8-bit image as the cover-image, many steganography experts recommend using images featuring 256 shades of *gray* as the palette, for reasons that will become

apparent. Grey-scale images are preferred because the shades change very gradually between palette entries. This increases the image's ability to hide information.

When dealing with 8-bit images, the steganographer will need to consider the image as well as the palette. Obviously, an image with large areas of solid color is a poor choice, as variances created by embedded data might be noticeable. Once a suitable cover image has been selected, an image encoding technique needs to be chosen.

Image Compression:

Image compression offers a solution to large image files. Two kinds of image compression are *lossless* and *lossy* compression. Both methods save storage space but have differing effects on any uncompressed hidden data in the image.

Lossy compression, as typified by JPEG (Joint Photographic Experts Group) format files, offers high compression, but may not maintain the original image's integrity. This can impact negatively on any hidden data in the image. This is due to the lossy compression algorithm, which may ``lose" unnecessary image data, providing a close approximation to high-quality digital images, but not an exact duplicate. Hence, the term``lossy" compression. Lossy compression is frequently used on true-colour images, as it offers high compression rates.

Lossless compression maintains the original image data exactly; hence it is preferred when the original information must remain intact. It is thus more favoured by steganographic techniques. Unfortunately, lossless compression does not offer such high compression rates as lossy compression. Typical examples of lossless compression formats are Compuserve's GIF (Graphics Interchange Format) and Microsoft's BMP (Bitmap) format.

Image Encoding Techniques:

Information can be hidden many different ways in images. Straight message insertion can be done, which will simply encode every bit of information in the image. More complex encoding can be done to embed the message only in ``noisy" areas of the image that will attract less attention. The message may also be scattered randomly throughout the cover image.

.

The most common approaches to information hiding in images are:

- Least significant bit (LSB) insertion
- Masking and filtering techniques
- Algorithms and transformations

Each of these can be applied to various images, with varying degrees of success. Each of them suffers to varying degrees from operations performed on images, such as cropping, or resolution decrementing, or decreases in the colour depth.

LEAST SIGNIFICANT BIT INSERTION:

One of the most common techniques used in steganography today is called least significant bit (LSB) insertion. This method is exactly what it sounds like; the least significant bits of the cover-image are altered so that they form the embedded information. The following example shows how the letter A can be hidden in the first eight bytes of three pixels in a 24-bit image.

Pixels: (00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

A: 10000001

Result: (00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001000 00100111 11101001)

The three underlined bits are the only three bits that were actually altered. LSB insertion requires on average that only half the bits in an image be changed. Since the 8-bit letter A only requires eight bytes to hide it in, the ninth byte of the three pixels can be used to hide the next character of the hidden message.

A slight variation of this technique allows for embedding the message in two or more of the least significant bits per byte. This increases the hidden information capacity of the cover-object, but the cover-object degrades more statistically, and it is more detectable. Other variations on this technique include ensuring that statistical changes in the image do not occur. Some intelligent software also checks for areas that are made up of one solid color. Changes in these pixels are then avoided because slight changes would cause noticeable variations in the area.

Advantages of LSB Insertion:

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

- A slight variation of this technique allows for embedding the message in two or more of the least significant bits per byte. This increases the hidden information capacity.

Masking and filtering :

Masking and filtering techniques hide information by marking an image in a manner similar to paper watermarks. Because watermarking techniques are more integrated into the image, they may be applied without fear of image destruction from lossy compression. By covering, or masking a faint but perceptible signal with another to make the first non-perceptible, we exploit the fact that the human visual system cannot detect slight changes in certain temporal domains of the image.

Technically, watermarking is not a steganographic form. Strictly, steganography conceals data in the image; watermarking extends the image information and becomes an attribute of the cover image, providing license, ownership or copyright details.

Masking techniques are more suitable for use in lossy JPEG images than LSB insertion because of their relative immunity to image operations such as compression and cropping.

Algorithms and transformations:

Because they are high quality colour images with good compression, it is desirable to use JPEG images across networks such as the Internet. Indeed, JPEG images are becoming abundant on the Internet.

JPEG images use the discrete cosine transform (DCT) to achieve compression. DCT is a lossy compression transform, because the cosine values cannot be calculated precisely, and rounding errors may be introduced. Variances between the original data and the recovered data depends on the values and methods used the calculate the DCT.

Images can also be processed using fast Fourier transformation and wavelet transformation. Other properties such as luminance can also be utilised. The HVS has a very low sensitivity to small changes in luminance, being able to discern changes of no less than one part in thirty for random patterns. This figure goes up to one part in 240 for uniform regions of an image.

Modern steganographic systems use spread-spectrum communications to transmit a narrowband signal over a much larger bandwidth so that the spectral density of the signal in the channel looks like noise.

The two different spread-spectrum techniques these tools employ are called direct-sequence and frequency hopping. The former hides information by phase-modulating the data signal (carrier) with a pseudorandom number sequence that both the sender and the receiver know. The latter divides the available bandwidth into multiple channels and hops between these channels (also triggered by a pseudorandom number sequence).

The Patchwork method is based on a pseudorandom, statistical process that takes advantage of the human weaknesses to luminance variation. Using *redundant pattern encoding* to repeatedly scatter hidden information throughout the cover image, like a patchwork, Patchwork can hide a reasonably small message many times in a image. In the Patchwork method, *n* pairs of image points *(a,b)* are randomly chosen. The brightness of *a* is decreased by one and the brightness of *b* is increased by one. For a labeled image, the expected value of the sum of the differences of the n pairs of points is then 2n. Bender shows that after JPEG compression, with the quality factor set to 75, the message can still be decoded with an 85

This algorithm is more robust to image processing such as cropping and rotating, but at the cost of message size. Techniques such as Patchwork are ideal for watermarking of images. Even if the image is cropped, there is a good probability that the watermark will still be readable.

Other techniques encrypt and scatter the hidden throughout the image in some pre-determined manner. It is assumed that even if the message bits are extracted, they will be useless without the algorithm and stego-key to decode them. Although such techniques

do help protect against hidden message extraction, they are not immune to destruction of the hidden message through image manipulation.

SYSTEM DESIGN:

These are the steps followed in image hiding while transmission and de noising after receiving:

1. Get a cover image (publicly accessible material)
2. Take the information to be hidden (message or image)
3. Combine cover image with the information to be hidden(we follow LSB algorithm for this)
4. While transmission it will be corrupted by noise
5. Use any of the filtering methods, ex: wiener filtering for de noising in wavelet domain
6. Here filter is employed in order to remove the noise
7. During extraction a password check is provided
8. If password is matched then extraction of hidden information.

Conclusion:

In this paper, we take an introductory look at steganography. Several methods for hiding data in, images were described, with appropriate introductions to the environments of each medium, as well as the strengths and weaknesses of each method.The key algorithm for designing the steganography system has been dealt. Most data-hiding systems take advantage of human perceptual weaknesses, but have weaknesses of their own. We conclude that for now, it seems that no system of data-hiding is totally immune to attack.

However, steganography has its place in security. Though it cannot replace cryptography totally, it is intended to supplement it. Its application in watermarking and fingerprinting, for use in detection of unauthorised, illegally copied material, is continually being realised and developed.

Also, in places where standard cryptography and encryption is outlawed, steganography can be used for covert data transmission. Steganography can be used along with cryptography to make an highly secure data high way.Formerly just an interest of the military, Steganography is now gaining popularity among the masses. Soon, any computer user will be able to put his own watermark on his artistic creations.

Bibliography:

1.M.Kuhn.

Steganography mailing list.

WWW: http://www.jjtc.com/Steganography/steglist.htm, 1995.

Private Site, Hamburg, Germany

2. N.F. Johnson.

Steganography.

WWW: http://www.jjtc.com/stegdoc/.

George Mason University

3. C. Kurak and J. McHugh.

A cautionary note on image downgrading.

In *Proceedings of the 8th Annual Computer Security Applications Conference*, pages 153-159, 1992.

4. W. Bender, D. Gruhl, N. Morimoto, and A. Lu.
Techniques for data hiding.
In *IBM Systems Journal, Vol. 35, Nos. 3-4*, pages 313-336, February 1996.