Steganography in BMP

Muhammad Naeem

A major goal of security techniques is "Confidentiality" – ensuring that adversaries gain no intelligence from a transmitted message. There are two major techniques for achieving confidentiality.

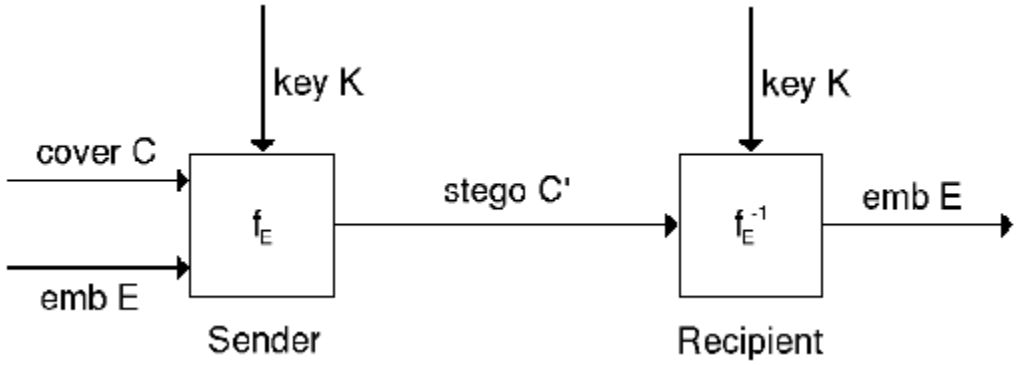
Encryption

Transforming the message to a ciphertext such that an adversary who overhears the ciphertext can't determine the message sent. The legitimate receiver possess a secret decryption key tha allows him to reverse the encryption transformation and retrieve the message. The sender may have used the same key to encrypt the message (with symmetric encryption schemes) or used a different, but related key (with public key schemes.) DES and RSA are familiar examples of encryption schemes.

Steganography

The art of hiding a secret message within a larger one in such a way that the adversary can't discern the presence or contents of the hidden message. For example, a message might be hidden within a picture by changing the low-order pixel bits to be the message bits.

The stegosystem is conceptually similar to the cryptosystem.



emb : The message to be embedded. It is anything that can be represented as a bit stream (an image or text).

cover: Data/Medium in which **emb** will be embedded.

stego: Modified version of the cover that contains the embedded message, **emb**.

key: Additional data that is needed for embedding & extracting.

f_E: Steganographic function that has cover, emb & key as parameters.[1][3]

In order to understand how steganography is applied to digital images, one must understand what digital images are.
On a computer, an image is an array of numbers that represent light intensities at various points (pixels). Images can have 8 bits per pixel or 24 bits per pixel.

With 8 bits/pixel, there are 2^8, or 256, color varieties. With 24 bits/pixel there are 2^{24} , or 16,777,216, color varieties.

Color variation for a pixel is derived from 3 primary colors: red, green, and blue.

24 bit image example:

24 bit images use 3 bytes to represent a color value (8 bits = 1 byte)

1 pixel = (00100111 11101001 11001000)
red green blue

Steganography Methods

Four steganography methods will be explored:

- Least Significant Bit Insertion
- Algorithms and Transformations
- Redundant Pattern Encoding
- Spread Spectrum Method

Least Significant Bit Insertion (LSB)

The idea behind the LSB algorithm is to replace the LSB of each pixel with the secret message or say insert the bits of the hidden message into the least significant bits of the pixels. These pixels may be chosen at random.

Simplified Example with a 24 bit pixel:

1 pixel:
(00100111 11101001 11001000)

Insert 101:
(00100111 11101000 11001001)
red green blue

Simplified Example with an 8 bit pixel:

1 pixel:
(00 01 10 11)
white red green blue

Insert 0011:
(00 00 11 11)
white white blue blue

Disadvantages of LSB Insertion:

As can be inferred from the example with the 8 bit pixel, applying LSB insertions can drastically alter the color constituents of the pixel. This could lead to noticeable differences from the cover image to the stego image, thus alerting observers of the existence of steganography. Color variations are less conspicuous with 24 bit images, however these files are much larger. Both 8 bit and 24 bit

images are vulnerable to image processing, such as cropping and compression.

Advantages of LSB Insertion: A major advantage of the LSB algorithm is it is easy and quick at small text.

Algorithm

My Work is related to explanation and implementation of the LSB Encoding Algorithm , by calculation I found the following formula which is related to the relationship b/w the amount of data to be encoded and the size of the image

''Formula for Storage of Characters in BMP
''Truncate(Ch) = ((W x H x 3) - (PpC + Off_Set)) / Target_Pixels
w = width of the Image
h = height of the image
PpC = Number of Pixels per Character
Off_Set = used to define maximum limit of the storage of the data
Ch = Number of Characters

Performance

As the algorithm is based on finding unique pixels. So initially all pixels would be available. But as the pixels locations would be utilized, finding unique pixel location will become processor intensive. So the performance will eventually degrade.

I took 10 images with random size. Hide maximum amount of data they can absorb inside themselves. The average graph of the data is as follows

Limiatations

There is no hard and fast limitations associated with the algorithm mentioned above. However it is to point out that all of the implementation was done in Visual C++ / VB. Both of the tools have an image/picture box control. As you save the snapshot from this control. So this require your machine to be set at 32bit pixel …… at video card settings

Conclusions and Application

There is a growing importance of steganography in intelligence work, as it is viewed as a serious threat to some governments. Most researchers believe that steganography's niche in security is to supplement cryptography, not replace it. For further reading on steganography, steganography software, and steganalyis, the process of detecting steganography, see the references below:

References

1. Chaffing and Winnowing: Rivest, Ronald L. "Chaffing and winnowing: Confidentiality without encryption." *CryptoBytes (RSA laboratories)* 4.1 (1998): 12-17.

2. Tanenbaum, Andrew S. "Computer networks, 4-th edition." ed: Prentice Hall (2003).