A propose method for hiding image into image

Dr. Abdulkareem O. Ibadi
Baghdad University Collegefor Economic Sciences

Mr. Omar Z. Akif
Baghdad University
Ibn Al-Haytham College

Ms. Ann F.Razzak
Baghdad University
Ibn Al-Haytham College

Abstract

Information hiding is an important class of security which is widely used in computer and network security nowadays. In this research, a proposed technique is used to hide a secret image in several personal images. A new method was suggest to divided the secret image into eight partitions each partition will be embedded in the single image (cover image) by using a special algorithm.

The receiver have the same algorithm for extracting the hidden partitions and collect them to retrieve the hidden image.

The hidden image will be divided into nine parts (the head and eight parts of the image information) each of which will be embedded in single image, so the carrier is a collection of at least nine images and are sent as a wall of facebook for example.

1.Different kinds of Steganography

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy far greater than necessary for the object’s use and display. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding[1].

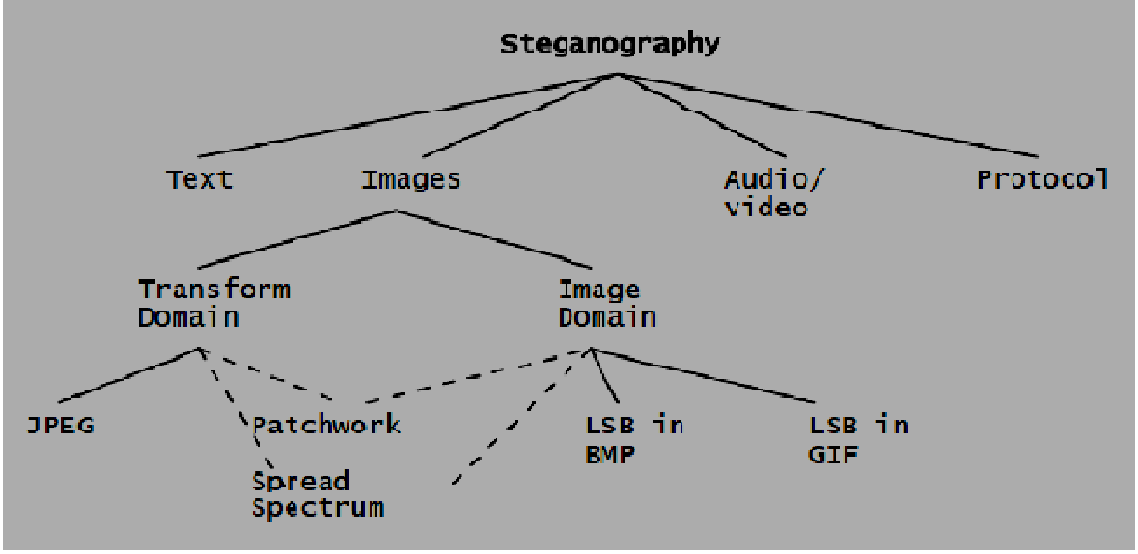

Figure 1 shows the four main categories of file formats that can be used for steganography and Categories of image steganography.

2.Image definition

To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image [2]. 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 colors. These pixels are displayed horizontally row by row.[3]

The number of bits in a colors scheme, called the bit depth, refers to the number of bits used for each pixel . The smallest bit depth in current colors schemes is 8, meaning that there are 8 bits used to describe the colors of each pixel. Monochrome and grayscale images use 8 bits for each pixel and are able to display 256 different colors or shades of grey. Digital colors images are typically stored in 24-bit files and use the RGB colors model, also known as true colors . All colors variations for the pixels of a 24-bit image are derived from three primary colors: red, green and blue, and each primary colors is represented by 8 bits [2]. 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 colures that can be displayed, the larger the file size [3].

3.Image file 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 uses 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) [4].

4.Processing .bmp Image Files

Image files come in a variety of formats—jpg, gif, bmp, etc. Of these, bitmapped files are conceptually the simplest. A bitmap file (indicated with a .bmp extension) consists of two sections—an information section that contains information about the structure of the file, and the image section itself. For 24-bit color images, the image is represented as a series of three bytes per pixel, with each byte containing values for the blue, green, and red color “guns” (in that order) that are used to produce that pixel. This arrangement allows for 256×256×256=16,777,216 possible colors. There are other kinds of .bmp files that have fewer colors.

Because each pixel in a 24-bit color image requires three color bytes to define, bitmap images can be very large. In principle, .bmp files can be compressed, but this is generally not done. It is possible to apply “lossless compression” algorithms, such as the widely used ZIP compression algorithm, to .bmp files. The results depend greatly on the nature of the image itself. For example, an image with large blocks of single colors could be compressed significantly. However, this process has nothing to do with the image format itself because a compressed file needs to be uncompressed back to its original state before it can be used as an image file. [5]

5.Least Significant Bit insertion method

Least significant bit insertion is a common, simple approach to embed information in a cover file. The LSB is the lowest order bit in a binary value. This is an important concept in computer data storage and programming that applies to the order in which data are organized, stored or transmitted.

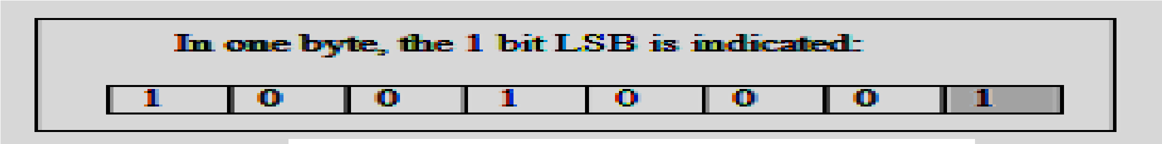

Fig. 2. Least Significant Bit

The last bit of the byte is selected as the least significant bit (as illustrated in Figure 2) because of the impact of the bit to the minimum degradation of images. The last bit is also known as right-most bit, due to the convention in positional notation of writing less significant digit further to the right.[6]

In this method, we can take the binary representation of the hidden data and overwrite the LSB of each byte within the cover image. If we are using 24-bit color, the amount of change will be minimal and indiscernible to the human eye.

As an example, suppose that we have three adjacent pixels (nine bytes) with the following RGB encoding:

10010101 00001101 11001001 10010110 00001111 11001010 10011111 00010000 11001011

Now suppose we want to "hide" the following 9 bits of data (the hidden data is usually compressed prior to being hidden): 101101101. If we overlay these 9 bits over the LSB of the 9 bytes above, we get the following (where bits in bold have been changed):

10010101 0000110**0** 11001001 1001011**1** 0000111**0** 1100101**1** 10011111 00010000 11001011

Note that we have successfully hidden 9 bits but at a cost of only changing 4, or roughly 50%, of the LSBs. This description is meant only as a high-level overview. Similar methods can be applied to 8-bit color but the changes are more dramatic. Gray-scale images, too, are very useful for steganographic purposes.[7]

6.The Proposed Method algorithm

Hiding image into an image is not a widespread technique used in the field of Steganography because of the large quantity of data that needed to be embedded. With least significant bit method a cover file of length eight fold the length of the hidden file (because single bit must be embedded in a byte), so, using image files of such length are revealed.

The proposed method is to hide the secret image into several images and send them in a suitable public media. Facebook is the most suitable media for transfer such number of photos with no suspicion because it is usually used for publishing the social occasions.

The proposed method can be described in the following steps:

1. Separate the header of secret image from the image information in a single part called H.
2. Divide the image information into eight parts called M_1,M_2,..,M_8 respectively. The secret hidden image is a nine distinct parts.
3. Chose a compound suitable cover consists of at least eight images of the same length.
4. Hide each secret part in single cover image.
5. Publish the cover images into facebook.

The hiding process is performed using the LSB method which described in the previous sections. The receiver will download the published images and retrieve the secret image in a secret predefined sequence.

Figure 3, describes the block diagram of the algorithm's main steps.

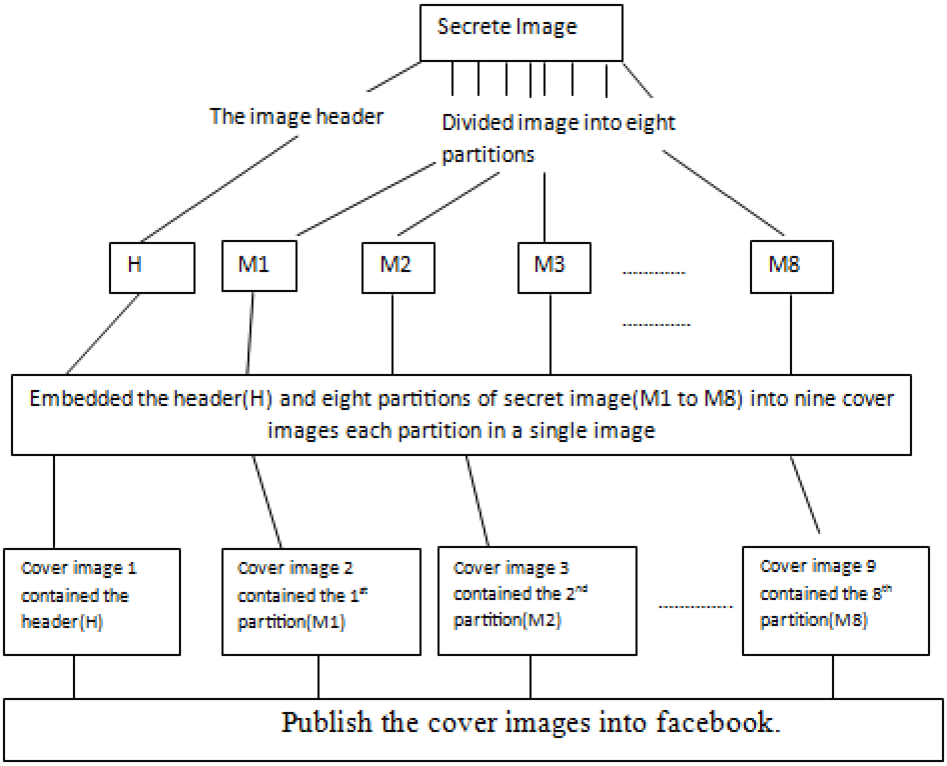
Secrete Image
The image header
Divided image into eight partitions
H
M1
M2
M3
....
M8
Embedded the header(H) and eight partitions of secret image(M1 to M8) into nine cover images each partition in a single image
Cover image 1 contained the header(H)
Cover image 2 contained the 1st partition(M1)
Cover image 3 contained the 2nd partition(M2)
Cover image 9 contained the 8th partition(M8)
Publish the cover images into facebook.

Figure 3, describes the block diagram of the algorithm's main steps.

7.System security

Using encryption in the proposed algorithm is not preferable because it can be detected easily by statistical analysis. The secrecy of the algorithm is obtained by using the cover images in predefined secret combination, so, by publishing nine cover images the system complexity will be equal to 9!=362880.

The system complexity can be increased by using more than cover images. If N cover images are used the system complexity will compute as follows:

$$X={}_{N}P_{9}=\frac{N!}{9!}$$

X is the system complexity which is equal to choosing 9 permutations from N. If N=15, X will be 3,603,600. The secrecy of the algorithm will be increased by using large value for N.

The complexity can be increased by adding new factor by permutated the secret image nine parts. The complexity can be computed as follows:

$$X={}_{N}P_{9}=\frac{N!}{9!}*9!=N!$$

So, for N=15 the complexity will be 1,307,674,368,000. At each possible permutation the attacker need to retrieve the compound secret image in a time consuming operations. If the operation can be performed in one second, for example, the attacker will retrieve the secret image in 42042 years.

8. Conclusions and future work

The conclusions:

1. A new method that was suggested in this research to hide image in the image without any recognized about this hide image by divided the secret image into eight partitions and additional partition to the header, then by using a privacy images as a cover images hide each partition in the single image by using LSB. The results is nine cover images sending to the receiver by using facebook.
2. The complexity of this method is very high because the probability it’s a very high the secrecy of the algorithm is obtained by using the cover images in predefined secret combination, so, by publishing nine cover images the system complexity will be equal to 9!=362880.

The future work:

1.Increase number of partitions secret image to increase the complexity.

References:

1.T. Morkel 1, J.H.P. Eloff 2, M.S. Olivier 3, Information and Computer Security Architecture (ICSA) Research Group, Department of Computer Science,University of Pretoria, 0002, Pretoria, South Africa.

2. Johnson, N.F. & Jajodia, S., “Exploring Steganography: Seeing the Unseen”, *Computer Journal*, February 1998.

3. Johnson, N.F. & Jajodia, S., “Steganalysis of Images Created Using Current Steganography Software”, *Proceedings of the 2nd Information Hiding Workshop*, April 1998.

4. Johnson, N.F. & Jajodia, S., “Exploring Steganography: Seeing the Unseen”, *Computer Journal*, February 1998.

5. Clean Coding Company TITLE: *BMP Format* SUBJECT: *Windows Bitmap File Format Specifications* AUTHOR: *Wim Wouters* VERSION: *V1.1* Copyright ®2008, Last printed on Tuesday 4 November 2008.

6. L.Y. Por, W. K. Lai2 Z. Alireza, T. F. Ang, M.T. Su, B. “StegCure: A Comprehensive Steganographic Tool using Enhanced LSB Scheme” Delina, Faculty of Computer Science and Information Technology University of Malaya 50603, Kuala Lumpur MALAYSIA.

7. Md. Manzoor Murshed “Steganography using LSB hiding” Upper Iowa University Fayette, Iowa, USA.