Information Hiding in Color Image Using Steganographic technique

Sundos A. Hameed Al-azawi
Abbas A. AbulHameed
Al-Mustainsiriyah University ,
College of Sciences , Computer sciences Dep.

Abstract

Steganography is one of the important research subjects in the field of information security. It enables secret communication by embedding messages in the texts, images, audio, video files or other digital carriers. Among all the image information hiding methods, LSB embedding is widely used for its high hiding capacity and it is with great significance to detect the images with hidden messages produced by LSB embedding effectively, accurately and reliably. Therefore, many experts made efforts on the LSB steganography and steganalysis research over the years. This research presents a steganographic technique based on using LSB of one of the pixel color components in the image and changes them according to the message's bits to hide. The rest of bits in the pixel color component selected are also changed in order get the nearest color to the original one in the scale of colors. This new method has been tested with others that work in the spatial domain through applying some common metrics which give us good result as a compared with the other steganographic tools.

1. Introduction:

Information hiding , steganography, and watermarking are three closely related fields that have a great deal of overlap and share many technical approaches. Information hiding (or data hiding) is a general term encompassing a wide range of problems beyond that of embedding messages in content. The term hiding here can refer to either making the information imperceptible (as in watermarking) or keeping the existence of the information secret. Steganography, coming from Greek word stegos, meaning covered and graphia, which means writing, is the art and science of hiding the information within information. According to Greek historian Herodotus, the famous Greek tyrant Histiaeus, while in prison, used unusual method to send message to his son-in-law. He shaved the head of a slave to tattoo a message on his scalp. Histiaeus then waited until the hair grew back on slave's head prior to sending him off to his son-in-law.

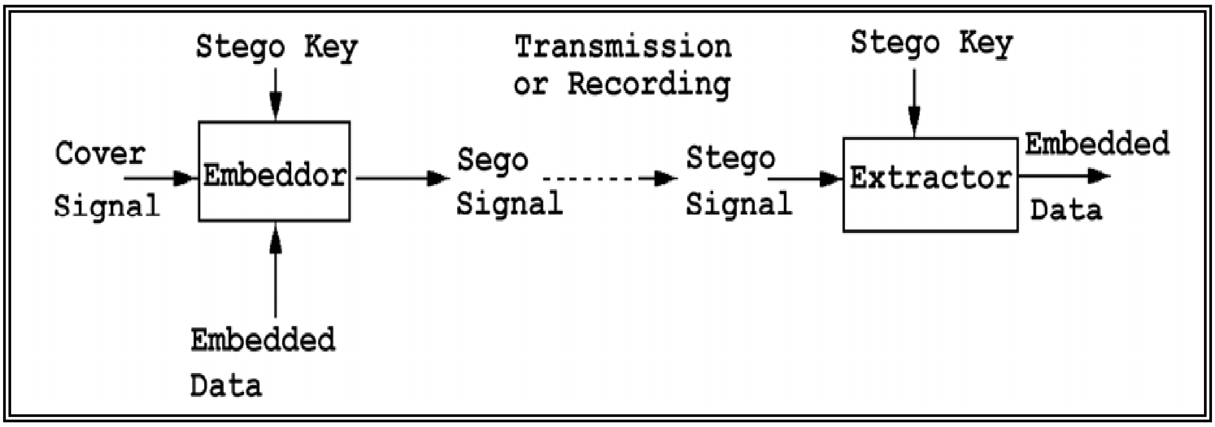
Invisible inks have always been a popular method of steganography. Ancient Romans used to write between lines using invisible inks based on readily available substances such as fruit juices, urine and milk. When heated, the invisible inks would darken, and become legible. Ovid in his "Art of Love" suggests using milk to write invisibly.

In a digital world, Steganography and cryptography are both intended to protect information from unwanted parties. Both Steganography and Cryptography are excellent means by which to accomplish this but neither technology alone is perfect and both can be broken. Steganography is often compared to Cryptography in its ability to restrict unauthorized access to information. Cryptography is used to encrypt or scramble the data in a fashion that only the intended recipient can decrypt it. While Steganography is used to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second message present.[1,2,3]

2. Basic Data Hiding Model

General principles of data hiding technology are illustrated in Figure 1. A data message is hidden within a cover signal (object) in the block called embeddor using a stego key, which is a secret set of parameters of a known hiding algorithm. The output of the embeddor is called stego signal (object). After transmission, recording, and other signal processing which may contaminate and bend the stego signal, the embedded message is retrieved using the appropriate stego key in the block called extractor [4].

Fig. 1 Block diagram of data hiding and retrieval.



3. Data Hiding in the Graphic Files

Coding secret messages in digital images is by far the most widely used of all methods in the digital world of today. This is because it can take advantage of the limited power of the human visual system (HVS). Almost any plain text, cipher text, image and any other media that can be encoded into a bit stream can be hidden in a digital image. To a computer, an image is an array of numbers that represent light intensities at various points, or pixels. These pixels make up the images raster data. When dealing with digital images for use with Steganography, 8-bit and 24-bit per pixel image files are typical. 8-bit images are a great format to use because of their relatively small size. The drawback is that only 256 possible colors can be used which can be a potential problem during encoding. Usually a gray scale color palette is used when dealing with 8-bit images such as (.GIF) because its gradual change in color will be harder to detect after the image has been encoded with the secret message. 24-bit images offer much more flexibility when used for Steganography. The large numbers of colors (over 16 million) that can be used go well beyond the HVS, which makes it very hard to detect once a secret message, has been encoded. The other benefit is that a much larger amount of hidden data can be encoded into 24-bit digital image as opposed to an 8-bit digital image. Information can be hidden many different ways in images. The most common approaches to information hiding in images are:

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

Each of these can be applied to various images, with varying degrees of success [2].

4. 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 : (01000001)

Result: (00100110 11101001 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 begin hiding the next character of the hidden message [5].

5. Proposed Technique

Most of the algorithms that work in the spatial domain using a LSB method (or any of its derivatives) as the algorithm for information hiding, that is, hide one bit of information in the least significant bit of each color of a pixel. But these methods can't stand a type of statistical analysis, even if partly camouflaged in the amount of information hidden.

The problem stems from the fact that modifying the three colors of a pixel produces a major distortion in the resulting color. This distortion is not visible to the human eye, but detectable by statistical analysis. So that the proposed technique supposes:

- For example, if a pixel of the cover image with the RGB (Red-Green-Blue code) color A8A8A8 #is used, in binary:

(10101000-10101000-10101000)

and 1 bit with value 1 is set on each LSB bit of each color component, to hide the message 111, the result would be :

(1010100**1**-1010100**1**-1010100**1)**

Table .1 illustrates the results that obtained by hiding the message 111 in the pixel (10101000-10101000-10101000) with the LSB method.

Table.1 the results of hiding in three colors

	Hexadecimal	**Decimal**	**Red**	**Green**	**Blue**
Original pixel	A8A8A8	11053224	168	168	168
Modified pixel	A9A9A9	11119017	169	169	169

In theory, the three least significant bits of the pixel have changed, introducing a small distortion, but the difference between the old and new color represents a leap of 65793 colors in the scale of colors.

- The proposed technique introduces more efficiency and less distortion that store the 3 bits of information to hide in the same color.

Using the same example, the 3 bits of information will be introduced in the 3 LSB bits of green color as follow: (10101000-10101**111**-10101000).

Table. 2 illustrates the results that obtained by hiding the message 111 in the pixel (10101000-10101000-10101000) with the proposed technique.

Table.2 the results of hiding in one color

	Hexadecimal	**Decimal**	**Red**	**Green**	**Blue**
Original pixel	A8A8A8	11053224	168	168	168
Modified pixel	A8AFA8	11055016	168	175	168

In this case the leap in the scale of colors is 1792 colors (in the case of changing the color green, if modify the blue color difference would be only 7 colors), that being the extreme case because it has been replaced last 3 bits with 0 value for 3 bits with a 1 value, that is, in most cases the distortion will be much lower.

- In order to choose the color for the concealment, the proposed technique preliminary select the color with higher ratio because it represents more diversity, leading to less noticeable changes. Thus, the chosen color will be the one that provides greater distortion and, therefore, the result of the withholding of information will be less detectable.
- Then, calculate the distance between the original color and the steganographic color. Should the distance is greater than a certain value (determined by the number of bits to hide), the color is decremented to get a final color closest to the original, implying a further reduction in the distortion caused by the hidden information.

For example, using a cover byte 11001000 to hide 3 bit of information (111), with a simple LSB results in 11001**111**, this has a difference of 7 values with respect to the original.

- Applying the method proposed here to the above example (in this case, decreasing the 4th least significant bit, which have been used 3 bits LSB to hide information) results in 1100**0111**, with a distance of 1 from the original byte but with the same hidden information.

6. The Results:

The proposed technique used Figures (2.a) & (3.a) as covers with BMP (Bit Mapped Picture) format image and a hidden message of TXT file. This section presents a comparison of the results of the metrics of distortion [6,7] (Mean Squared Error, Peak Signal to Noise Ratio) applied to steganographic images obtained by different tools. Obviously, a lower distortion represents a better steganographic method because it is closer to the values of the original image. The quality of the stego-images Figures (2.b) & (3.b) are measured by the peak signal-to-noise ratio (PSNR). It is the most popular criterion to measure the distortion between the cover image and stego-image, assuming that the orginal image $f(i\ ,j)$ that contains M by N pixels and the stego image $f^{*}(i\ ,j)$. First, the Mean Squared Error (MSE) is computed between the cover image and stego-image as follows:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[f(i,j) - f*(i,j)]^{2}$$

The Peak Signal-to-Noise Ratio (PSNR) metric is defined as:

$$PSNR = 10\log_{10}\frac{(2^{L} - 1)^{2}}{MSE}$$

Where L: is the bit per pixel (i.e. 24 bit/pixel for the image used in this work).

Fig. 2.a Cover Image.

Fig. 2.b Stego Image

The results of above metrics are illustrated in table.3. The actual value is not meaningful, but the comparison between many values for different steganographic tools gives one measure of quality.

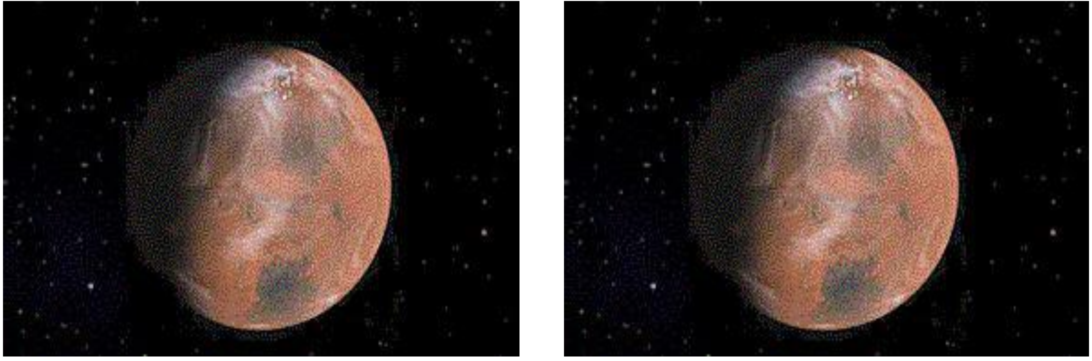


Fig. 3.a Cover Image   Fig. 3.b Stego Image

Table. 3 the results of the metrics

Tool	**MSE**	**PSNR**
Original image	0.000	0.0
the algorithm (1 bit/pixel)	0.020	2.0
the algorithm (2 bits/pixel)	0.043	9.9
the algorithm (3 bits/pixel)	0.137	3.1
Digital Invisible Ink Toolkit [8]	0.947	414685.4
Hide4PGP [9]	0.477	409701.5
Steghide [10]	0.320	1063210.6
S-tools [11]	0.025	7805527.2
wbStego [12]	0.947	413435.6
Hide in Picture[13]	0.665	410218.1
White Noise[14] Storm	1.101	311260.8

This table can verify that the new algorithm (in any of its three versions) offers best results in the metrics MSE, PSNR when compared with the other steganographic tools.

7. Conclusions

This research presents a new technique that modified the performance of the LSB method hiding information in only one color of the three colors at each pixel of the cover image. The color is decremented to get the final color is as close as possible to the original one through calculating the distance between the original color and the steganographic color. Also, applying the metrics of distortion (MSE, PSNR) on steganographic image obtained by different tools leads to get good results for the proposed technique.

References

[1] Singh, Pradeep Kumar & Aggrawal, R.K. “ Enhancement of LSB based Steganography for Hiding Image in Audio ” , (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 05, 2010, 1652-1658.

[2] Thampi, Sabu M. “ Information Hiding Techniques: A Tutorial Review ”, Department of Computer Science & Engineering ,LBS College of Engineering, Kasaragod, Kerala- 671542, S.India, 26 feb 2008 .

[3] Amin, Muhalim Mohamed , Ibrahim, Subariah , Salleh, Mazleena & Rozikatmin, Mohd “ Information Hiding using Steganography”, Department of Computer System & Communication Faculty of Computer Science and Information system, University Teknologi Malaysia, 2003.

[4] Dutta1, Poulami, Bhattacharyya1, Debnath & Kim2, Tai-hoon “ Data Hiding in Audio Signal: A Review ” International Journal of Database Theory and Application, Vol. 2, No. 2, June 2009.

[5] Press, Marsland “ Implementation of Improved Steganographic Technique for 24-bit Bitmap Images in Communication ” , Journal of American Science 2009:5(2) 36-42, V.P.O Sahauran, Tehsil Kharar, Distt. Mohali, Punjab- 140104, INDIA 91-098786-77624.

[6] Katzenbeisser, S. & Petitcolas, F. “ Information hiding techniques for steganography and digital watermarking ” Artech House Books, 1999.

[7] Simone,Francesca De, Ticca, Daniele “ A comparative study of color image compression standards using perceptually driven quality metrics ” , Multimedia Signal Processing Group, Institute of Electrical Engineering, Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, August 11-14, 2008.

[8] Hempstalk, K., Digital Invisible Ink Toolkit 1.5, 2009-07-07. Available in http://diit.sourceforge.net

[9] Hempstalk, K., Digital Invisible Ink Toolkit 1.5, 2009-07-07. Available in http://diit.sourceforge.net

[10] Hetzl, S., StegHide 0.5.1, last update 2003-10-15. Available in http://steghide.sourceforge.net

[11] Brown, A., S-tools 4, 2009. Available in http://www.spychecker.com/program/stools.html

[12] Bailer, W., wbStego 4.3, 2009-07-07. Available in http://www.8ung.at/wbailer/wbstego

[13] Figueiredo, D., Hide In Picture 2.1, 2009-10-01 . Available in http://sourceforge.net/projects/hide-in-picture

[14] Arachelian, R., White Noise Storm 2.10, 2009-07-07. Available in http://www.nic.funet.fi/pub/crypt/steganography/.

اخفاء المعلومات في الصوره الملونه باستخدام تقنية الكتابه المخفيه

سندس عبدالامير حميد العزاوي
عباس عبدالعزيز عبد الحميد
الجامعة المستنصريه , كلية العلوم , قسم علوم الحاسبات

الخلاصه

علم الخفاء هو أحد أهم مواضيع البحوث في مجال أمنية البيانات حيث يٌمكن من الاتصال بشكل سري من خلال أخفاء الرسائل في الملفات النصيه والصور والصوت والفيديو او غيرها. أن أحدى طرق الاخفاء المستخدمه بشكل واسع في الصور هي LSB وذلك لقدرتها العاليه على الاخفاء وقوتها على كشف الصور التي تحتوي على رسائل مخفيه بشكل كفوء دقيق وبأعتماديه عاليه ، مما أدى على مر السنوات الى بذل الجهود والخبرات لعمل بحوث كثيره في هذا المجال. هذا البحث يقدم تقنيه بأستخدام LSB بأخفاء داخل أحد الالوان للبكسل في ملف الصوره. أما بقية bits يتم تغييرها أيضا من اجل الحصول على لون يكون الاقرب الى اللون الاصلي. تم أختبار هذه التقنيه بأستخدام بعض المقاييس الشائعه، مما تم الحصول على نتائج جيده بالمقارنه مع بقية تطبيقات الأخفاء الموجوده والمستخدمه.