

A New Robust and Secure Steganographic System for Greyscale Images

Hesham A. El Zouka

Computer Engineering Department, Arab Academy for Science and Technology,
Alexandria - Egypt

helzouka@aast.edu , hae@cs.nott.ac.uk

Abstract:

The research work in this paper shows that the currently available steganographic methods might be quite easily detected by using sufficiently careful analysis of the transmitted data. In order to minimize the error introduced due to hiding foreign message carrier into the cover image, a robust and secure algorithm will be introduced, which might be used efficiently to protect the embedded message against attacks. The idea in the new approach depends on spreading the secret message over the cover image using both a pseudo random number generator and a hash function that drive one bit from each block of pixels in a random sequence manner. The intensity of each random pixel is predicted based on the average weighted sum value of the surrounding pixels. The difference between each random pixel is calculated causing a deviation error to be introduced. An error correction function will then be used to minimize the error introduced due to hiding foreign message carrier into the cover image and hence minimizing degradation of the embedded images.

Keywords: Steganography; Information hiding; Secure communications; Stego-analysis

1. Introduction

Steganography is the art of secret communication. Its purpose is to hide the very presence of communication as opposed to cryptography whose goal is to make communication unintelligible to those who do not possess the right keys. By embedding a secret message into a digital image, a stego-image is obtained. It is important that the stego-image does not contain any easily detectable artifacts due to message embedding that could be detected by eavesdropper. There are many different steganographic methods that have been overviewed and analysed by many researchers over the last few years, for e.g. hiding files in the least significant bits of digital images [1]. However, one common drawback of all current data embedding methods is the fact that the original image is distorted by some small amount of noise due to the data embedding itself. This noise could reveal the existence of secret message and hence, weaken the security value of the covert channel. In this paper we investigated most of the steganography techniques, which have been proposed in the last few years. After analysing the drawbacks in the analysed systems, we proposed an approach, which maintains such noise in a way that makes the transmitted signal undetectable. In addition, we built the software programs that provide the simulation results including the histograms of both the cover images and the stego images, and the variance between them. Also, the simulation results for all equivalent substitution techniques, which are covered in this paper, are used here for comparison purposes.

2. Previous Work

There are many different steganographic methods that have been proposed over the last few years. Most of the simple techniques can be broken by careful analysis of statistical properties of the channel's noise [2]. In substitution steganography techniques for example, one can observe that: these methods substitute insignificant parts of the image, e.g. the noise component of the cover with the secret message. These parts have specific statistical properties and the embedding process usually does not pay attention to them, and change the statistical profile of the cover significantly. A simple attack such as "laplace filtering" [3] can exploit this fact and detect the use of the steganographic system. In addition, these systems are extremely sensitive to cover modification and the attacker, who is not able to extract or prove the existence of a secure message can add a random noise to the transmitted cover or simply convert the image into another file format in attempt to destroy the secret message. With transform domain hiding techniques [4], more substantial processing is required to disable the readability of the embedded information. Meanwhile, most of these stenography systems have a vital drawback, which is that the system doesn't discard image blocks where the desired relation of DCT coefficients [5] cannot be enforced without severely damaging the image data contained in this specific block. TCP/IP packet headers [6], [7] can also be reviewed easily. For e.g., firewall filters are set to test the validity of the source and destination IP addresses. Those filters can also be configured to catch packets that have information in supposed unused or reserved space. Based on the analysis of spread spectrum techniques [8], it can be observed that phase coding provide robustness against resembling of the carrier signal, but at the same time it has a low data transmission rate. These techniques have a problem with the absolute phase of all following segment that followed the first modified one, since all of them will have a change that could be noticeable to the attacker. Moreover, at the receiver end; the embedding process is reversed and image restoration technique such as adaptive wiener filter [9] is needed to estimate the original image.

3. A Hybrid System using Greyscale Images

The proposed technique in this paper distorts the image insignificantly by making small modifications over a large number of pixels. Therefore, we will spread the secret message over a large area of cover image to produce a small modification on the carrier media. The new approach combines both cryptography and steganography to exchange secret messages in a way that it's impossible to discover without the knowledge of the cover image and the secret key that have been used. Firstly, the parity bits from pixels c_1, c_2, \dots, c_i are computed and encoded with the corresponding bits in the text file, which contains the secret message. The process is repeated for the whole stream of bits. If the computed parity bit c_i and the secret bit m_i are equal, then the encoded bit is zero and if the 2 input bits are different, then the output is one. Finally, the encoded bits are lined up to reconstruct the encoded file. Now, the file is ready to be encrypted and sent in any insecure channel to the receiver who had both, the secret key and a copy of the cover image which has been used. Therefore, the receiver of the encoded message will decipher the message using his secret key and the shared cover image.

3.1 Spreading Secret Message over an Untraceable Cover Area

After the encoding process had taken place, the output file was encrypted and sent directly to the other party as a cipher file. However, instead of sending an encrypted stream of bits, an alternative scheme can be adopted by injecting the stream of bits back to the cover image with a probability of 50% of changing the LSB of embedded pixels in the image. Our goal here is to embed one bit of the secret message m_i into one pixel of the cover image c_i , where C is composed of all the pixels $\{c_1, c_2, \dots, c_i\}$, and since $L(m) < L(C)$ the rest of the

image can left unchanged. Moreover it's possible to select only some message pixels c_i in a rather random manner according to a secret key and leave the other unchanged. Therefore, the idea depends on spreading the secret message over the cover image using both a pseudo random number generator and a hash function [10] that specifies one bit from each block of pixels randomly as follows:

$$P(I) = \sum_{j \in i} LSB(c_j) \bmod 2$$

If the parity bit of one cover block c_i doesn't match with the secret bit m_i , the program will flip the LSB of one pixel in the block in a way to that makes $p(I_i)$ equals to m_i . Studying the properties of pixels surrounding the target pixel, we could invoke a statistical command that will fix up the number of 1's or 0's inside the chosen block in a way that conceals any statistical existence of a hidden information inside the stego image. This is done by studying the neighbouring pixels surrounding the chosen bit and changing its value to match the adjacent one in a way that prevent any statistical tracing. The Gibbon cover image in figure 1 provides special features and will be used in this research work as a test image.

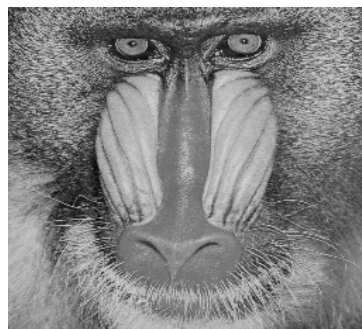


Figure1. Gibbon Cover Image [11]

4. Implementation and Design

Before communication starts, both sender and receiver have to agree on the location of the pixels c_i . These pixels will be used as subjected pixels, from which the parity bits $p(c_i)$ are computed. Hence, the computed parity bits are xored with the corresponding bits in the secret message file to produce the encoded message. A parity bit of a given pixel is computed from the following hash function:

$$p(c_i) = \sum_{j \in i} s_j \bmod 2$$

At this stage, the output file could be encrypted using standard encryption technique such as triple DES or IDEA [10]. The file is then compressed and sent directly to the other parity or injected back to the transmitted image with the encoded parity bits as mentioned before. The embedding process is preceded by leaving or altering the parity bit according to the value of the embedded bits. In the decoding process, a reverse process is launched to reconstruct the secret message.

4.1 Error Correction Function

This function is used to minimize the error due to hiding the foreign message carrier into cover images. The method is based on statistical analysis of images and it is very robust to changes that happen due to file format conversions or blurring filter such as a Gaussian convolution. The error correction algorithm proceeds by dividing the image into blocks of 4*4 pixels and chooses the blocks in sequence according to a given seed number. The intensity of each pixel $x[i][j]$ within the cover area is predicted according to the value of

pixels in a specific neighborhood. Hence, the difference between the intensity of each pixel and its adjacent pixels is calculated as follows:

$$out[i][j] = in[i][j] - \frac{1}{16} \sum_{i=1-4} \sum_{j=1-4} x[i][j]$$

Where $x(i,j)$ represents the pixel coordinates in the selected cover region c . For each tested pixel in the block, the average weighted sum of the surrounding pixels is computed and compared with the target pixel.

5. Simulation Results and Comparisons with Related Systems

The simulation results showed that the text message could be embedded without any degradation of the image. Figure 2 shows the stego image after an edge detection algorithm has been run.

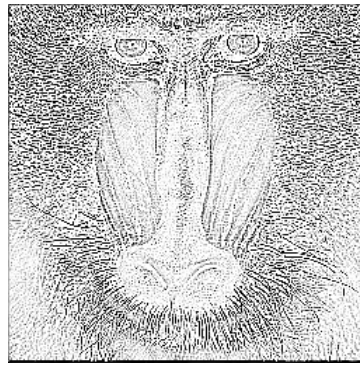


Figure 2. Stego Image with Sharpen Edges

Studying the histogram of Laplace filter in figure 4 for the provided image, we notice that on average, the amount by which the image is modified is smaller than some known substitution embedding systems that we investigated.

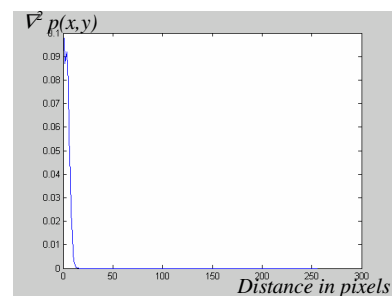
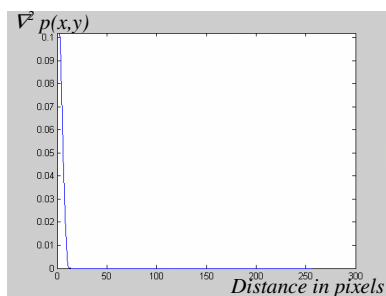


Figure 3. Laplace Filter of the Cover Image Figure 4. Laplace Filter of the new technique

For example comparing the distortion introduced by PGMStealth [12], and the new technique, we can clearly see that the new technique provides visibly fewer and less peaks than PGMStealth filtered histogram which has a wider band and many peaks clustered around zero as seen in figure 5.

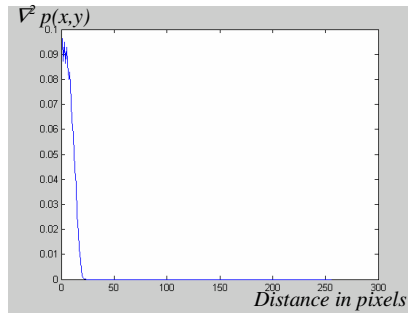


Figure 5. Laplace Filter of PGMStealth Stego Image

In hide and seek version 4.1 [13] the same cover image of a Gibbon was used to embed the same message. The original image of Gibbon is 256 x 256 pixels and 256 shades of grey. However, the resulting image was forced to 320 x 480 pixels as shown in figure 6. Instead of "stretching" the image to fit, large black areas were added to the image making it 320 x 480.

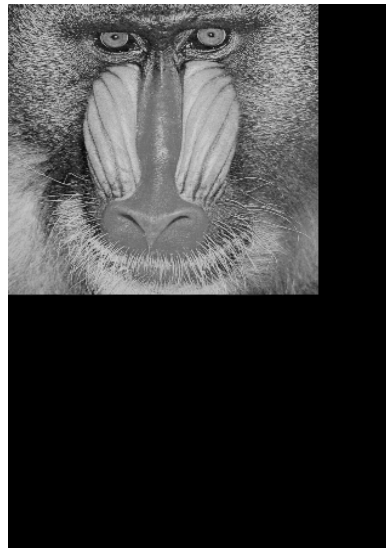


Figure 6. Stego Image Using Hide and Seek Technique

Black Wolfs Picture Encoder version 0.90a [14], uses the LSB method to embed the secret messages. The program consists of series of programs and it only works with 256 x 256 images with 256 grey levels. Despite the size restrictions, which are necessary for this program to run efficiently, the stego image is obviously distorted. The stego image was cropped and padded to a 320 x 200-pixel image. Figure 7 illustrates this distortion when the text message is embedded.



Figure 7. The Black Wolf Encoder Stego Image

Out of all the steganography techniques have been analyzed and tested, the one which gave the best results was the Snow steganography system [15] since there is no visual change in sight of the image. The “Whitespace” system as it is called in some literature hides the secret messages into an image in a way that nearly indistinguishable from normal random noise. However, using some other powerful steganalysis techniques such as the discrete Laplace filter, its possible to detect secret message in the image. Also giving the histogram of the repeated pixels within the greyscale Gibbon image, as shown in figure 8, one can notice that: the histogram of the original image has a limited peak height at certain locations, while the histogram of stego image has higher peaks at many points.

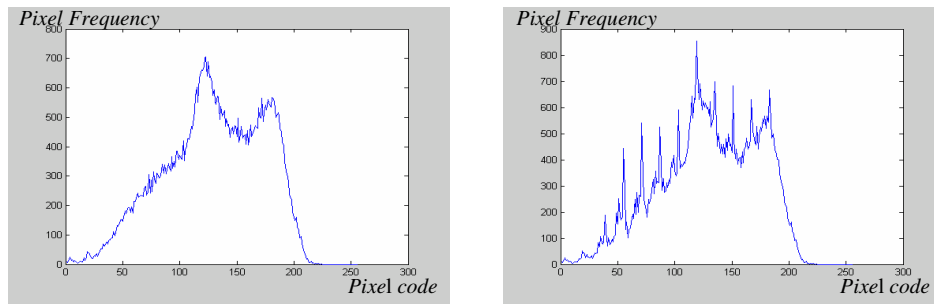


Figure 8. (A) Histogram of the Cover Image (B) Histogram of Snow Stego Image

However, studying the statistical properties of neighbouring pixels in the proposed steganographic system, the embedding process gave us better results. That is because our new system has an algorithm that determines whether a candidate pixel can be used or not by checking the variance in luminosity of the neighbouring pixels. After looking to the pixel repetition histogram of the stego image in figure 9 and compare it with histogram of original image, it can be observed that there is very little difference between them and there are a few thin lines distributed in various part of the histogram.

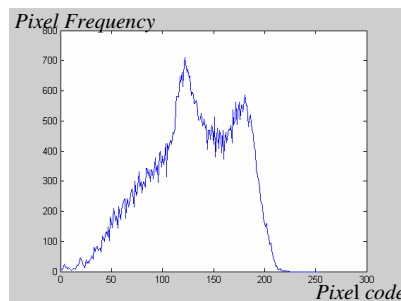


Figure 9. Histogram of the New Technique

For some one who doesn’t hold a copy of the original cover image, it’s hard for him to discover the differences between the normal and embedded one. For example, relative to the histogram of white noise, it’s obviously seen that the white noise histogram contains many high peaks in the middle of the histogram, which could give the attacker the implication that the picture was subjected to some modifications.

6. Robustness of the Proposed System

Our first approach which does not change the cover is perfect in the sense above, but it does have the disadvantage of needing a key to be communicated separately. On the other hand our parity encoding scheme does introduce small changes to the image, but as shown in the results, it is very hard to detect. This method also achieves a very low-cost of embedding foreign data into the image. This benefit is due to the fact that the original function model is

non-linear. The changes introduced by this model are nonlinear and hence are not easily noticeable or detectable and almost impossible to model unless great sophistication is used by the attacker. Figure 5 showed the histograms of Laplace filtered greyscale image printed in one coordinate system for the public domain program PGMStealth, which hides secret information in the LSB of every cover-pixel, while Figure 4 showed the histogram of Laplace filtered for the proposed new model. Since the embedding process adds noise to the picture, which is statistically quite different from true random noise, the histogram of PGMStealth program differs extremely. On the other hand, the histogram of our model does not prove the existence of a secret message which minimises the chance that the embedded information will be detected. The histogram and analysis tests show that our method is more robust than any simple bit substitution model in common use. First the sender can choose which element should be modified and according to both the random number in the sequence and the calculated parity bit. Studying the error function and changing the seed or the width of the pixel on which the parity bit is calculated could change the cover statistics least.

7. Conclusion and Future Work

7.1 Conclusion

The results of the proposed approach showed that the encoding process distorted the image insignificantly by making small modifications over large numbers of pixels. The algorithm divides the image into small blocks that are analysed for parity check values equivalent to the embedded bit. The technique was designed with the intent of maximising the quality of the stego image by the aid of error correction function that introduced extremely small modification to the cover images. Initial investigations showed that this modification was difficult to detect visually, and there is no tell-tale artifact could be picked up during the investigation process. In order to compare the provided approach with other established methods, many stego-analysis techniques were investigated and applied to the stego image. Testing our proposed steganographic algorithm's using an edge detection method, we found that the stego image does not show any artifacts and thus, it gives no indication that the image contains any hidden information. Comparing the Laplace filter histogram for the provided cover image with the one which contains the embedded message, we noticed that on average, the amount by which the image is modified is smaller than other known substitution steganographic systems that we investigated. Looking at the pixel repetition histogram of the stego image and comparing it with the histogram of the original image, it can be observed that there are only very small differences between them, and there are a few fine lines distributed over some parts of the histogram.

7.2 Future Work

We accepted that it is hard to build a model that adapts to all of the parameters needed to define the statistical properties of a steganographic model. We noticed also that there is always a trade-off between robustness and data rate, which may prevent any embedding process from meeting the needs of all applications. However for the future work of this work we recommend that the cryptography methods should be taken more seriously into account in order to design a more successful steganographic system and in an attempt to provide a secure function to the steganography process. In addition to make the communication even more secure, we recommend that the secret message should be compressed or encoded before the encryption process takes place. This is important because in this way we will minimise the amount of information that is sent, and hence minimizing the chance of degrading the image. We recommend also using efficient error correction coders and programs that could run in parallel with the embedding process in order to detect any suspect pixel with distinctive features that could lead the attacker for further investigation.

8. References

- [1] Asha, A. Information hiding - The Art of Steganography, GSEC practical. SANS Institute, 2005.
- [2] Elke ; Fraz. Steganography preserving statistical properties, proceeding of the 5th internationally Workshop on information Hiding, Noordwijkerhout, The Netherlands, October 2002, LNCS 2578, pp. 278-294, Springer 2003.
- [3] Petitcolas, F. A. P., R. J. Anderson ; Kuhn, M. G. Information Hiding- A Survey, Proceedings of the IEEE, vol. 87, no.7, Jul. 1999, pp. 1062-1078.
- [4] Stefan Katzenbeisser ; F. A.P. Petitcolas ; Information Hiding Techniques for Steganography and Digital Watermarking . Artech House ; ISBN: 1580530354, 2000.
- [5] Andres, A.; Huertas ; Gerard Medioni. Detection of Intensity Changes with Subpixel Accuracy Using Laplacian- Gaussian Masks, IEEE Transactions on pattern analysis and machine intelligence, vol. pami-8, no. 5, pp. 651-664, September 1986.
- [6] Anderson, R. J.; Needham, R. M. ; Shamir, A.. The Steganographic File System, in Proceedings of the Second International Workshop on Information Hiding, vol. 1525 of Lecture Notes in Computer Science, Springer, 1998, pp. 73-82.
- [7] Piscitello, D. and Chapin, A. L., Open Systems Networking: TCP/IP and OSI, Addison-Wesley, Reading, Mass., 1997
- [8] Petitcolas, J., Gabriella, C. A Bayesian Approach to Spread Spectrum Watermark Detection and Secure copyright protection for Digital Libraries. IEEE Conference on Computer vision and pattern recognition (CVPR'99, Colorado, USA, June , 1999.
- [9] Westfield, A. and Pfitzmann, A., "Attacks on seteganographic Systems". In: Proc. 3rd Information Hiding Workshop, Dresden, Germany, September (1999) 61-75.
- [10] Mohammad Peyravian ; Nev Zunic. Hash-Based Encryption System. Computers & Security Vol. 18, No.4, pp.345-350 , 2003.
- [11] USC – SIPI. Image database, Signal & image processing Institute. University of Southern California. <http://sipi.usc.edu/publications.html>
- [12] Kwan, M. How gifshuffle works, Technical report, Helsinki University of Technology, June 2004.
- [13] Money, C. Hide and Seek. <ftp://ftp.csua.berkeley.edu/pub/cypherpunks/st~ganography/hdsk41b.zip> ,1994-1997.
- [14] Black Wolf's Picture Encoder vO.90B, <ftp://ftp.csua.berkeley.edu/pub/cypherpunks/Steganography>
- [15] . Arachelian, R. White Noise Storm. <ftp://ftp.csua.berkeley.edu/pub/cypherpunks/steganography/wns210.zip>.
- [16] Cover, T.M.; Thomas J. A. Elements of Information Theory. New York, Chichester: John Wiley & Sons, 1991.