# ENTROPY AS AN ESTIMATE OF IMAGE STEGANOGRAPHY

George Corser
Computer Science and Engineering
Oakland University
Rochester, MI 48309, USA
gpcorser@oakland.edu

**Abstract**

Steganography has been called concealed writing, security through obscurity, and the art of hiding messages in plain sight. It long ago advanced beyond invisible ink, and entered the digital domain. One example: embedding messages in images.

How can we detect image steganography? In theory, the more data hidden in a file, the higher that file's entropy. That is, if bits are too disorderly, if data are too random, steganography may be suspected. This paper compares entropies of files with various-sized messages steganographically embedded.

## 1 INTRODUCTION

What is steganography? Steganography, or "stego," is secret writing, or more generally, any covert communication achieved by hiding messages in unsuspected carriers. In earlier times, messages may have been concealed using invisible ink or secret compartments. In the digital age stego involves embedding secret messages in image files, such as Jpegs.

Steganalysis aims to detect steganography. Blind steganalysis would detect steganography regardless of the method used to embed secrets in a carrier image.

Why are steganography and steganalysis important? Because sometimes encryption may alert observers that secret communication is taking place, especially if concealed payloads are large. Stego does not even necessarily involve encryption. Large amounts of secret data may be appended to pictures or videos and censors may not be able to detect it unless they are looking for it specifically.

Wang & Wang recently observed, “The battle between steganography and steganalysis represents an important part of 21st century cyber warfare with a profound influence on information security" [2]. Stego has broader applications. It may benefit any adversaries where at least one possesses the ability to monitor the communication of the other.

When might steganography and steganalysis be used, particularly as applied to digital images? Consider the hypothetical situation where a spy has infiltrated a supposedly secure facility. He wishes to transmit secret data files to his allies, but encrypting it may raise suspicions. Defenders may detect the encryption, confront the spy, and request he decrypt the data, perhaps defeating his plan and leading to his exposure. A better solution for the spy: hide encrypted data inside unencrypted, innocuous-looking files.

## 2 BACKGROUND

Image files provide good cover as carrier files. They are big enough to contain a substantial amount of hidden information, and they are common enough that they are relatively unlikely to arouse suspicion. Further, a superficial visual inspection of the cover image is often insufficient to detect the hidden data.

### 2.1 Image Format Steganography

Perhaps the most straightforward method of concealing information within an image is to append the secret information to the cover carrier image. There are many easy ways to accomplish this. One could simply open a JPEG with a hex editor, then insert a message after the end-of-file marker. Similarly, Cheddad [9] suggests using a DOS append command:

```
C:\> Copy Cover.jpg /b + Message.txt
/b Stego.jpg
```

Consider an infamous example of image format steganography: DangerousKitten, the JPEG file containing a great number of hacker tools. The file provides weak cover, however, because the image file alone is less than 135k, while the image file with the tools embedded is over 570k. Consequently, even a casual observer can detect that something is unusual about the file. This counteracts the primary purpose of steganograhy: not raising suspicion.

### 2.2 Spatial Domain Steganography

Another straightforward stego method: least significant bit (LSB) modification. This technique adjusts a photo's pixel values just slightly, by changing the value of the LSBs of each pixel. Spatial domain modification does not

protect messages from unsophisticated defenses, however. For example, defenders could simply alter or strip the LSBs of *all* images. Then concealed stego messages would be damaged and never get through.

### 2.3 Frequency Domain Steganography

Images are often filtered or compressed (as in JPEG) for various reasons, which can damage the embedded message in a stego file. Frequency domain steganography protects against this problem.

The three most common transforms: discrete Fourier transform (DFT), discrete cosine transform (DCT), and discrete wavelet transform (DWT). Jsteg, the first publicly available Jpeg-altering stego tool, transforms images into the frequency domain using discrete cosine transform (DCT), then replaces the least significant bits (LSBs) of the DCT coefficients with the data to be concealed. [3]

Steganalysis techniques have improved to detect this. Farid showed that suspected images may be partitioned in the frequency domain to separate high and low frequency signals, which is advantageous because data are more often hidden in the higher frequency signals. Farid reported wavelet-like decomposition improved frequency domain steganalysis strategies by enabling statistical analysis of image files. [4]

### 2.4 Image Steganography for Espionage

After reports that enemies of the United States were posting steganographic messages on eBay, researchers in [3] investigated such images. They reported, "From our eBay and Usenet research, we so far have not found a single message." [3]

On June 25, 2010, the FBI was granted a warrant to arrest several people suspected of conspiring to conduct espionage. Liu calls this "the first confirmed use of steganography for espionage." [5]

In this case, the Russian Federation's Moscow Center supposedly provided software which enabled the alleged conspirators to encrypt and decrypt data in images, and post the images on websites. According to the FBI complaint, "well over 100" readable text files were recovered by law enforcement from such images. [6]

There appears to have been an increase in the number of publications on the topic of steganography. A search of the ACM Digital Library and IEEE Xplore using the search term, "steganography" revealed 86 scholarly publications by the ACM, and 209 by IEEE in 2011 alone. In 2006, these figures were 41 and 123, respectively. Based on these figures, scholarly interest in stego seems to have roughly doubled in the last five years.

### 2.5 Visual Inspection Steganalysis

When the method of steganographic embedding is known, the method of searching for that embedding may be tailored. This is referred to as *specific* steganalysis. When the method is unknown, search techniques are necessarily more general in nature. This is called *blind* steganalysis. [7] This paper focuses on the latter.

To avoid suspicion, cover images should be altered as little as possible. Again, a defender may not be aware *that* a stego image was embedded, let alone *how* it was embedded, but if the image raises any suspicion at all the steganography is defeated. Most stego methods successfully prevent steganalysis by visual inspection.

### 2.6 Frequency Domain Steganalysis

Frequency domain methods better maintain cover image and stego message integrity, protecting against both compression and filtering that an image may undergo. However, they are limited in the amount of payload they may carry.

Cheddad concludes, "The emerging techniques such as DCT, DWT and Adaptive steganography are not too prone to attacks, especially when the hidden message is small. This is because they alter coefficients in the transform domain, thus image distortion is kept to a minimum. Generally these methods tend to have a lower payload compared to spatial domain algorithms." [9]

Frequency domain methods include those that replace LSBs of DCT coefficients, rather than the LSBs of pixel intensities. Jsteg is an example of this. These methods are detectable by statistical means. [3]

### 2.7 Statistical Steganalysis and Entropy Tests

Chi-squared tests, pairwise comparisons and other statistical techniques can measure the randomness of the data in an image. If bits are too disorderly, meaning data are too compact, steganography may be suspected. That is, the more data hidden in a file, the higher the entropy. [3]

Entropy is measured by multiplying a discrete set of random events, $a_j$, by their probabilities, $P(a_j)$. Summing these products over all j, $\Sigma_j$, yields a measure of the average information per source output. [13]

$$H = -\Sigma_j P(a_j) \log P(a_j)$$

In the case of an image, we use the histogram of the observed image to estimate the symbol probabilities of the source, where *k* ranges from zero (0) to L-1 (usually 255) [13], as below:

$$\hat{H} = -\Sigma_k p_r(r_k) \log_2 p_r(r_k)$$

Certainly, methods have been concocted to try to defeat this type of steganalysis, however, entropy remains one of many tools in the steganalyst's toolkit.

### 2.8 Unbreakable Stego: One-time Pads

While it is not directly addressed in this paper, it is important to mention: Steganography coupled with cryptography may be unbreakable in certain circumstances. Anderson and Petitcolas observed that one-time pads have been proved to be undefeatable in the field of cryptography. They consider situations where a message must never be revealed to an enemy—even years later when technology may be able to crack an encryption using brute-force methods.

"When such concerns arise in cryptography—for example, protecting traffic that might identify an agent living under deep cover in a foreign country—the standard solution is to use a one-time pad; Shannon provided us with a proof that such systems are secure regardless of the computational power of the opponent." [11] Anderson and Petitcolas speculate whether steganography theory will ever include a similar principle.

One-time pads are a form of encryption, not steganography. However, they may help defeat statistical steganalysis. Theoretically, if the keys to one-time pads remain statistically random and never-reused not only are they unbreakable but they do not reduce the information entropy of the stego file.

## 3 METHOD

This study compared entropies of Jpeg files before and after embedding hidden messages. The process was as follows:

First, "cover" carrier files were created. Each was cropped to 450x250 pixels. File sizes differed depending on how much internal variation of color and shape appeared the picture.

Second, "hidden" files, i.e., files to hide in the cover carrier files, were created. File `sw`: a 15-character text file containing the single word, "Rumplestiltskin." File `pm`: a 670-character file containing a poem, Shakespeare's Sonnet #18. File `h2`: the first 1626 characters of the Project Gutenberg EBook of The Hunting of the Snark, by Lewis Carroll [15].

Note: Files larger than 1626 characters were not embeddable using Steghide. But large files were embeddable using Cheddad's method, including the entire Project Gutenberg EBook of The Hunting of the Snark, size: 51.9KB, and the Project Gutenberg EBook of War and Peace, by Leo Tolstoy [16], size: 3.13MB.

Third, two sets of "stego" files, i.e., the cover files including the hidden files, were created. One set used Chedded's image format technique [9], the other and Steghide's DCT technique. [10]

Finally, entropies were computed using John Walker's `ent` program [14].

## 4 EXPERIMENTAL RESULTS

Entropy as a measurement of steganography contained in a file proved inconclusive in this study. Consider the entropies of the stego files created using Steghide, below.

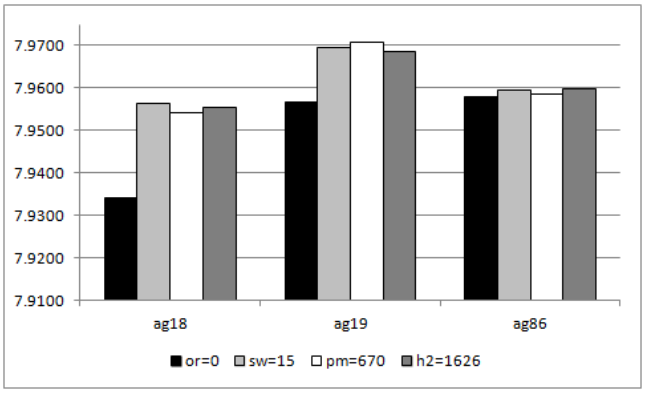


**Figure 1: Entropies of stego files, original (or), single word (sw), poem (pm), and truncated Hunting of the Snark (h2). Stegonography performed using Steghide.**

The chart above shows a slight increase in entropy as more hidden information is added. But this is not always the case. While the entropies of the original files increase in entropy as the pictures become "busier," containing more color and texture variation, the addition of more hidden data does not always produce a corresponding increase in entropy.

The chart below shows slight *decreases* in entropy as more hidden data were added. This is expected as the hidden data were not encrypted. Unencrypted textual data tends to have low entropy because it is more orderly.

This study shows that images transformed by steganography, as performed by Steghide, have increased entropy, but perhaps not sufficiently increased to detect steganography. See Figure 1. Further, the maximum amount of data that can be hidden is limited. Steganography using the image format technique may decrease entropy when the host image file entropy exceeds the hidden file entropy. Plaintext hidden files, as used in this study, produce varying entropy changes when

comparing original cover files to corresponding stego files. See Figure 2.

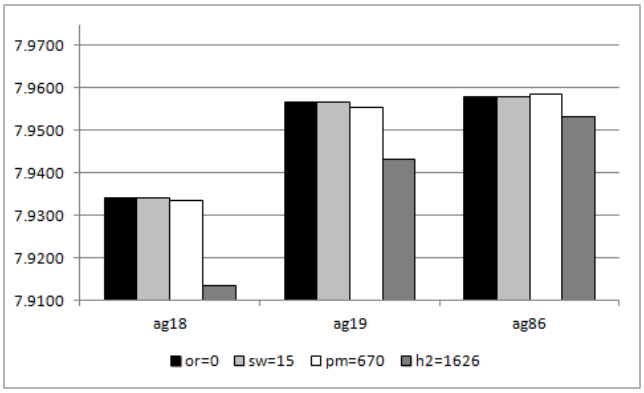


**Figure 2: Entropies of stego files, original (or), single word (sw), poem (pm), and truncated Hunting of the Snark (h2). Steganography performed using image format technique.**

## 5 CONCLUSIONS

This study shows that the magnitude of change in entropy is small from a cover file, a file that has no embedded hidden data, to a stego file, a file that does have embedded hidden data. Further, the type of steganography applied and the types of Jpeg images used as cover can tend to either increase or decrease the entropy of resultant stego files.

## 6 REFERENCES

[1] Tsu, S. The Art of War: Translation by Samuel B. Griffith. Oxford University Press. 1998.

[2] Wang, H., Wang, S. Cyber Warfare: Steganography vs. Steganalysis. *Communications of the ACM* . 2004.

[3] Provos, N., Honeyman, P. Hide and Seek: An Introduction to Steganography. *IEEE Security & Privacy*. 2003.

[4] Farid, H. Detecting Hidden Messages Using Higher Order Statistical Models. *Proc. IEEE International Conference on Image Processing*. 2002.

[5] Liu, Q., Sung, A., Qiao, M. Neighboring Joint Density-Based JPEG Steganalysis. *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, No. 2, Article 16, Publication date: February 2011.

[6] http://www.justice.gov/opa/documents/062810complaint2.pdf.

[7] Luo, X. et al. On the Typical Statistic Features for Image Blind Steganalysis. *IEEE Journal On Selected Areas In Communications.* August 2011.

[8] Peticolas, F., et. al. Information Hiding – A Survey. *Proceedings of the IEEE, special issue on the protection of multimedia content.* July 1999.

[9] Cheddad, A., Condell, J., Curran, K., and Mc Kevitt, P. Digital Image Steganography: Survey and Analysis of Current Methods. *15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems.* 2008.

[10] http://steghide.sourceforge.net/download.php.

[11] Anderson, R., Petitcolas, F. On the Limits of Steganography. *IEEE Journal On Selected Areas In Communications*, Vol. 16, No. 4, May 1998.

[12] http://web.vu.union.edu/~shoemakc/watermarking/watermarking.html.

[13] Gonzalez, R., Woods, R. Digital Image Processing, Third Ed. Pearson Prentice Hall, Upper Saddle River, NJ. 2008. pp. 532

[14] http://www.fourmilab.ch/random/.

[15] http://www.gutenberg.org/files/13/13.txt.

[16] http://www.gutenberg.org/cache/epub/2600/pg2600.txt .

[17] http://steghide.sourceforge.net/download.php.