Computer and Information Science
www.ccsenet.org/journal.html
Vol. 1, No. 3
August 2008

Image Analysis for Online Dynamic Steganography Detection

Kevin Curran & Joanne Mc Devitt
Intelligent Systems Research Centre
Faculty of Engineering
University of Ulster, Northern Ireland, UK
E-mail: kj.curran@ulster.ac.uk

Abstract

In recent years there has been a noticeable growth in the quantity of available Steganography tools on the World Wide Web. Steganography may be used to hide messages within images and it is widely believed that terrorist organizations may be communicating through the use of steganography. With this in mind there is a need to detect hidden data using Steganalysis – the art of detecting messages using Steganography. This paper presents a Steganalysis tool which scans images on the Web to test if they have been affected by Steganography.

Keywords: Steganography, Steganalysis, Watermarking, Security, Image processing

1. Introduction

Steganography derives from the Greek word Steganos meaning covered and Graphos meaning writing or drawing (Cole, 2005). Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. Steganography involves selecting an appropriate carrier file such as an image or audio file, removing the less important information from that file and injecting the hidden message in its place. When the cover message and the secret message are combined a stego image is created. Figure 1 illustrates a typical steganography system.

Steganographic messages may be encrypted before they are inserted into the cover image for increased security. Only those who know the technique used to encrypt the message can recover the message. The purpose of steganography is not to keep others from knowing the hidden information but to keep them from knowing the very existence of the information. If a steganographic method causes someone to suspect the carrier medium, the method has failed (Artz, 2001). Although messages embedded into an image are often imperceptible to the human eye, they often disturb the statistical nature of the image (Farid, 2006). The choice of the cover image in steganography is very important as it influences the security of the technique in a huge way. Some steganographic experts recommend greyscale images are the best to use as cover images (Fridrich & Goljan, 2006.). Images are the most popular medium for hiding data. The more detailed an image is, the fewer constraints there are on how much data it can hide before it becomes suspect (Artz, 2001). The internet is said to be a storehouse of steganographic material (Callinan & Kemick, 2006). There have been numerous rumours that terrorists are using steganography to exchange images over the internet. USA Today in 2001 stated that Al-Qaeda as well as other terrorist groups were using steganography to plan and implement terrorist attacks (Callinan & Kemick, 2006). Many people think steganography is a relatively new area in research but some examples of steganography trace far back to 440BC. Demeratus, a Greek at the Persian Court, warned Sparta of an invasion by Xerxes, King of Persia. Demeratus removed the wax from a writing tablet, wrote his message on the wood underneath and then covered the message with wax (Katenzenbeisser & Petitcolas, 2000). Another example of early steganography involved Histiaeus, who shaved the head of his most trusted slave and tattooed a message. Once the slave's hair had regrown, he was sent to transport the message (Katenzenbeisser & Petitcolas, 2000). This method was still used by German spies at the beginning of the 20th century. In 1499, Johannes Trithemius published Steganographia, one of the first books on steganography.

As technology evolved throughout the years so did steganography techniques. Invisible ink, created by juice or milk was discovered. When heat was applied to the document, the hidden writing became visible (Silman, 2001). In World War II, Germans used microdots to hide large amounts of data on printed documents, masquerading as dots of punctuation (Cole, 2003). Steganography has taken a giant leap forward which started in the 1990s when governments, industries and terrorist organizations began using software applications to embed messages and photos into various types of media. Steganography has become a hot topic on the internet in terms of electronic privacy and copyright protection. In 1995, a search using steganography as a keyword on the World Wide Web produced fewer than a dozen responses. In 1996, the same search produced 500 hits. In 1998, such a search produced over a thousand hits (Cole, 2003). Today, a search in Google produces 2,200,000 hits. Computer technology has made it so much easier to hide messages and more difficult to discover that message.

There are important connections between steganography and cryptography but both have very different goals. The goal of steganography is to keep the existence of a message hidden or to hide the fact that communication is taking place. In contrast, the goal of cryptography is to obscure a message or communication so that it cannot be understood (Druid, 2006). With this in mind, the security ability of both techniques is also different. In cryptography, the message is not hidden therefore a hacker can try to intercept and decrypt the message but with steganography, the hacker must discover the medium before he/she can try to intercept it. Putting their differences aside, steganography and cryptography make great partners. It is very common to use these techniques together. As an additional security measure, a steganographer can encrypt the message before it is hidden using steganography (Grodzinsky et al, 2005). Steganography involves hiding data in such a way that it is difficult for an attacker to detect the existence of a secret message in the carrier file. Based on this, three principles can be used to measure the effectiveness of a steganography technique. The principles are amount of data, difficultly of detection and difficulty of removal (Cole, 2003).

- Amount of data suggests that the more data you can hide the better the technique.
- Difficulty of detection relates to how easy it is for someone to detect that a message has been hidden. Once you increase the amount of data hidden in a file, the risk that someone will be able to detect the message also becomes higher.
- Difficulty of removal suggests that someone intercepting your file should not be able to remove the data easily

Three steganography protocols exist to allow users more flexibility in their steganography systems and to prevent attacks on their systems. These protocols are Pure Steganography, Secret Key Steganography and Public Key Steganography. In Pure Steganography, the embedding and extraction algorithms should only be known by the message sender and the intended receiver (Anon, 2006). With Secret Key Steganography, it is assumed that a party other than the sender and intended receiver knows the embedding and extraction algorithms. The sender embeds a message in a cover object using a secret key known as a stego key. If a third party intercepts the stego object and extracts the information, the result will be scrambled. Only the intended receiver who possesses the same stego key can extract the original message. Public Key Steganography is based on the principles of Public Key Cryptography. In Public Key Steganography both a public key and a private key are used. The public key is used in the embedding process and the private key is used in the extraction process. This allows the sender and the receiver to avoid exchanging a secret message which might be compromised. However, this method is susceptible to a man-in-the-middle attack. There are two ways by which steganography techniques can be categorised; the file type and the method of hiding. File Type categorisation breaks down steganography based on the type of carrier file. Each file format has specific properties which control how data is hidden in that type of file. Based on this, knowing the host file type can give you an idea of where the data might be hidden (Cole, 2003). There are three general ways to hide data. They are injection, substation and generation.

- Injection finds the areas of the file which will be ignored and injects the hidden message in that part of the file.
- Substitution finds the insignificant information in the file and replaces it with the hidden message
- Generation creates a new overt file based on the information contained in the covert message.

Today there are many uses of steganography on the World Wide Web. Most modern uses are based in the area of security. Governments use steganography for secure communication and to hide information from other governments. Steganography is also used by many businesses to maintain privacy and protect trade secrets. Technophiles use steganography to send secret messages just for fun. Another popular use for steganography today is watermarking. Watermarking involves injecting copyright marks and serial numbers into electronic mediums such as books, audio and video so that its source can be tracked or verified. There are many negative uses of steganography today. Many people believe that terrorists are using steganography to communicate by posting their secret messages in pictures of pornographic websites and sports chartrooms. Criminals use steganography to transmit child pornography. Steganography can also be used for corporate espionage where a person can get a job within a specific company with the intent of stealing valuable information. The stolen information can be transported to the recipient using steganography (Kn1ght10rd, 2005).

2. Steganography in Images

Digital images are the most widely used medium for steganography today. Digital images take advantage of our limited visual perception of colour. This field is expected to continually grow as computer graphics power also grows (Calpe, 2006). This report will focus on image steganography. In a computer, images are represented as arrays of values. These values represent the intensities of the three colours red, green and blue where the value of each of the three colours describes a pixel (Queirolo, 2006). These pixels are represented row by row. The number of bits in a colour scheme, called the bit depth, refers to the number of bits used for each pixel. The least number of bits in a current colour scheme is eight which means 8-bits are used to describe the colour of each pixel. Monochrome and greyscale images use 8-bits for each pixel and are able to display 256 different colours or shades of grey. Digital colour images are typically stored in 24-bit files and use the RGB colour cube (Morkel et al, 2005). The RGB Colour cube is illustrated in Figure 3.

Large images are the most desirable for steganography because they have more space to hide the data (Queirolo, 2006). The palette and composition of the image also contribute to how well the steganography tool does its job. Any images with gradual colour gradients or greyscale images are best for steganography as it is easier to insert "small errors" into them. The changes also appear more gradually and are less likely to be detected (Queirolo, 2006). Image compression is the application of data compression on digital images. The objective is to reduce redundancy of the image data in order to be able to store or transmit data in an efficient form. There are two types of image compression: lossy and lossless. A lossy data compression method is one whereby compressing data and then decompressing it retrieves data that may well be different from the original but is "close enough" to be useful in some way. Lossless data compression is a class of data compression algorithms that allows the exact original data to be reconstructed from the compressed data. Compression plays is a key factor in choosing which steganography algorithm to use. Lossy compression techniques result in smaller file sizes but they increase the possibility that some of the hidden message may be lost due to the removal of excess image data. With lossless compression, the original digital image will remain the same without the chance of loss but the image will not be compressed to the extent it would be with lossy compression (Morkel et al, 2005). Different steganography techniques have been developed to handle both of these compression types.

Common methods for hiding information within images include least significant bit insertion, masking and filtering, and transformation. Least significant bit insertion is a very popular steganographical technique because it is the most basic. The least significant bit is the lowest bit in a series of binary numbers. The LSB is located at the far right of a string. E.g., in the binary number 1011101, the least significant bit is the far right '1. In this method, the LSB of each byte in an image is used to store the secret data. The resulting changes are too small to be recognised by the human eye. A steganography tool which uses this method uses the RGB colour cube to make a copy of the image palette. The copy is rearranged so that the colours near each other in the RBG cube are near each other in the palette. The LSB of each pixel's binary number is replaced with one bit from the hidden message. A new RGB colour in the copied palette is found and a new binary number of the RGB colour in the original palette is found. The pixel is changed to the binary number of the new RGB colour (Si, 2004).

Masking and filtering techniques are usually restricted to 24-bit images or greyscale images (Krenn, 2006). These methods hide information in a manner similar to watermarks. While masking does not change the visible properties of an image, it can be performed in such a way so that it is not visible to the human eye. This method is much more robust than LSB modification with respect to compression since the information is hidden in the visible parts of the image. Transformation is a more complex way of hiding information. Various algorithms and transformations are applied to the image to hide information in it (Gupta, 2005). Discrete Cosine Transformations (DCT) is one such method. This method is used by the JPEG compression method to transform 8X8 pixel blocks of the image into 64 DCT Co-efficients each (Krenn, 2006). Steganography tools can use the LSB of the DCT Co-efficient to hide information. In addition to DCT, there are two other transformation steganographical techniques; fast fourier transformation and wavelet transformation (Gupta, 2005).

2.1 Current Steganography Software

S-Tools is a Steganographical tool that hides files in BMP, GIF and WAV files. It is a freeware program with a drag and drop interface that runs on most versions of Windows (from Windows 95 onwards). S-Tools offers the ability to hide multiple secret messages in one host file. It hides data in the three least significant bits of each byte of data. Hide and Seek is freeware software. It consists of a series of DOS programs which embed data in GIF files. Hide and Seek conceals data in the GIF files using the least significant bit of each data byte to encode characters. It then uses dispersion to spread the data throughout the GIF file in a pseudo-random fashion (Cole, 2003).

EZ Stego is a Java program which injects data in the least significant bit of GIF images. It sorts the colour palette so that closest colours appear next to one another. The message is then inserted into the least significant bits. The palette is then unsorted by renumbering all of the colours with their original values. The recipient can find the message by using the same algorithm to extract the message from the sorted palette (Si, 2004). Image Hide is a Steganography tools which can conceal data in a variety of formats. It alters the colour table of the pixels by replacing the least significant bits with bits of the hidden message, without increasing the size of the image. Digital Picture Envelope is a Windows 95/98/NT program which hides information in a bitmap file format. It can be used to conceal a large amount of data in an image without changing the size of the image file itself (Cole, 2003).

Outguess is a universal steganographic tool that allows the insertion of hidden information into the redundant bits of data sources. The program relies on data specific handlers that will extract redundant bits and write them back after modification. Currently only the PPM, PNM and JPEG image formats are supported for this method. Outguess allows for the hiding of two distinct messages in the data. GifShuffle is used to conceal messages in GIF image files by shuffling the colour map which leaves the image visibly unchanged. GifShuffle works with all GIF images including those with transparency and animation. It also provides for compression and encryption of the concealed message.

3. Steganalysis

Steganalysis is a relatively new research area with few articles appearing before the late 90s. Steganalysis is the process of detecting steganography by looking at variances between bit patterns and usually large file sizes. It is the art of discovering and rendering useless covert messages (Si, 2004). Steganalyst is a relatively new term that was developed to refer to someone who tries to break steganography techniques. Once a steganalyst has determined that a file has a message hidden in it and the message has been extracted, if it is encrypted, it then becomes the job of a cryptanalyst to try and break the cipher text and figure out what the plaintext message is (Cole, 2003). In steganalysis, comparisons are made between the cover-object, the stego-object and the possible portions of the message; the end result is the stego-object. If the message is encrypted, cryptanalysis techniques may be applied to further understand the message (Katzenbeisser & Petitcolas, 2000). The goal of steganalysis is to identify suspected information streams, determine whether or not they contain any hidden message and if possible, recover the hidden message (Si, 2004). The challenge of steganalysis is that:

- The suspect information stream may or may not contain hidden data
- The hidden data may have been encrypted before it was inserted into the carrier file
- Some of the suspect file may have noise or irrelevant data encoded into them which can make analysis very time consuming
- Unless it is possible to fully recover, decrypt and inspect the hidden data, often one has only a suspect information stream and cannot be sure if it is being used to transport secret data

In steganalysis, the steganalyst does not care which bits carry what information, he/she is trying to determine the existence of a hidden message (Zhang & Ping, 2006). If one can show that an image conceals a message, whether it can be read or not, then the stego system has failed (Zhang & Ping, 2006). Attacks and analysis on hidden information may take several forms: detecting, extracting and disabling or destroying hidden information (Chandramouli & Memon, 2006). A steganalysis attack represents the technique with which a steganalyst attempts to recover, modify or remove a stego message. An attack approach is dependent of what information is available to the steganalyst. Six steganalysis attacks exist which are incidentally derived from four cryptanalysis techniques: stego-only, known-cover, known-message, chosen-stego, chosen-message and known-stego. In the "stego-only attack" only the stego-object is available for analysis (Katzenbeisser & Petitcolas, 2000). This is the most difficult attack approach since there is no starting point from which to extract the hidden message. When the original cover data and the stego message are both available, this is known as the "known-cover attack". The "known-message attack" assumes either a part of the entire hidden message is available to the steganalyst (Brainos II, 2006). Analyzing the stego-object for patterns that correspond to the hidden message may be beneficial for future attacks against that system. Even with the message available, this attack may be considered equivalent to the "stego-only attack" (Cole, 2003).

In the "chosen-stego attack" both the steganography tool (algorithm) and the stego-object are known. In this case the key, if the message is encrypted and the hidden message are unknown (Brainos II, 2006). "Chosen – message attack" occurs when the steganalyst generates a stego-object from some steganography tool or algorithm from a chosen message. The goal in this attack is to determine corresponding patterns in the stego-object that may point to the use of steganography tools and algorithms. In the "known-stego attack" the steganography tool is known and both the original and stego-objects are available (Cole, 2003).

3.1 Types of Steganalysis

Steganalysis can be classified into two categories, Passive Steganalysis and Active Steganalysis. Passive Steganalysis involves detection only. The steganalysis process ends when the following question has been answered, "Does the media contain steganographic data?" In Active Steganalysis, the process is complete only after the hidden data is removed, destroyed or strategically altered to render it useless.

Steganographic systems often leave detectable traces within a medium's characteristics (Provos & Honeyman, 2006). This allows an eavesdropper to detect modified media, revealing that secret communication is taking place (Provos & Honeyman, 2006). With steganalysis, although the secret content is not exposed, its existence is revealed, which defeats the main purpose of steganography (Provos, Honeyman). Steganographic methods insert information and manipulate the images in ways to remain invisible (Katzenbeisser & Petitcolas, 2000). However, any manipulation to the image introduces some amount of distortion and degradation of some aspect in the 'original' image's properties (Katzenbeisser & Petitcolas, 2000). In images that have colour palettes or indexes, colours are typically ordered from the most used to least used, to reduce table lookup time. The changes between colour values may change gradually but rarely. Greyscale image colour indexes do shift in 1-bit increments, but all the RGB values are the same. Applying a similar approach to monochromatic images other than greyscale, normally two of the RGB values are the same with the third generally being a much stronger saturation of colour. Having occurrences of single pixels outstanding may point to the existence of hidden information (Katzenbeisser & Petitcolas, 2000). Added content to some images may be recognisable as exaggerated noise. This is a common characteristic for many bit plane tools when applied to 8-bit images. Using the 8-bit images without manipulating the palette will cause colour shifts as the raster pointers are changed from one palette to another (Katzenbeisser & Petitcolas, 2000). If the adjacent palette colours are very similar there may be little change. However, if the adjacent palette entries are dissimilar, then noise due to the manipulation of the LSB is obvious (Katzenbeisser & Petitcolas, 2000). There are two main types of steganalysis techniques; Visual Analysis and Statistical Analysis.

The majority of steganographic algorithms embed messages by replacing carefully selected bits of an image with bits of the secret message. The human ability is used for visual analysis (Westfeld & Pfitzmann, 2006). Analysing repetitive patterns may reveal the identification of a steganography tool or hidden information. An approach used to identify such patterns is to compare the original cover images with the stego images and note visible differences (Katzenbeisser & Petitcolas, 2000). This is called a "known-cover attack". Another visual clue to the presence of hidden information is the padding or cropping of an image. With some stego tools, if an image does not fit into a fixed size, it is cropped or padded with black spaces. There may also be a difference in the file size between the stego image and the cover image which indicates steganography. Another indicator is a large increase or decrease in the number of unique colours in a palette which increase incrementally rather than randomly (except greyscale images) (Si, 2004). The idea of visual attacks is to remove all parts of the image covering the message. The human eye can then distinguish whether there is a potential message or still image content (Westfeld & Pfitzmann, 2006). The filtering process depends on the presumed steganographic utility. The process is shown in Figure 4.

Statistical analysis is frequently used in steganography detection. Statistical tests can reveal that an image has been modified by steganography by determining that an image's statistical properties deviate from the norm (Provos & Honeyman, 2006). This is done by measuring the colour pairs in a image. These two colours, which do not differ greatly, are known as a close colour pair. In a steganographic image, the individual colours become "less different" when a file is embedded into an image. The occurrences of these colours are measured using statistics (Marcus, 2006). One specific technique is the Raw Quick Pairs (RQP) method. In this method, the ratios between the close pairs and the colour pairs are measured in an image. Then, a message is deliberately embedded into the image and the measurements are taken again. If there is a large difference between the two measurements, then it is unlikely that a message has been embedded into the original image (Marcus, 2006). A more sophisticated technique is Ueli Maurer's Universal Statistical Test for Random Bit Generators (Provos, 2001). If we use a block size of 8 bits, the expected result from the maurer test for a truly random source is 7:184. We expect images with hidden data to have higher entropy than those without. Westfeld & Pfitzmann outline an interesting statistical attack in their paper "Attacks on Steganographic System" (Westfeld &Pfitzmann, 2006). They observe that for a given image, the embedding of encrypted data changes the histogram of colour frequencies in a particular way (Provos, 2001). When using the least significant bit insertion method to embed encrypted data into an image that contains colour two more often than colour three, colour two is changed more often to colour three than the other way around. As a result, the differences in colour frequency between two and three are reduced in the embedding (Provos & Honeyman, 2006).

The most popular steganalysis tool available is Stegdetect. Stegdetect, provided by Neils Provos, is an automated tool for detecting steganographic content in images. It is capable of detecting several different steganographic methods to embed information in JPEG images. Currently, the detectable schemes are Jsteg, Jphide (Unix and Windows), Invisible Secrets, Outguess 01.3b, F5 (header analysis) and Appendix and Camouflage. Stegdetect 0.6 supports linear discriminant analysis. Given a set of normal images and a set of images that contain hidden data by a steganographic application, Stegdetect can automatically determine a linear detection function that can be applied to unclassified images. Linear discriminant analysis computes a dividing hyperplane that separates the non-stego images from the stego images. The hyperplane is characterised as a linear function. The learned function can be saved for later use of new images. Stegdetect includes StegBreak which is a brute force attack tool for determining the pass phrase (if any), which has been assigned to the cover image embedded with a hidden message (Si, 2004). Stego Suite is another popular steganalysis tool. It is a software bundle available for the investigation, detection, analysis and recovery of digital steganography. This product is comprised of three products; StegWatch is WetStone's flagship steganography tool, Stego Analyst is a state of the art image and audio file analyzer and Stego Break is a password cracker.

It is important to detect hidden messages within images. On the steganography side, this is important in order to find methods to improve the algorithm implementing steganography. By exposing the flaws of the algorithm, the user can further improve the algorithm in order to make it more difficult to detect whether or not data is hidden in the images (Blanco et al, 2005). Steganalysis is also extremely important with regards to security. In the age of Internet, images are sent via email or by posting them on websites. Detecting whether or not data is hidden in the images will allow us to monitor and further analyse the suspicious images in order to find what the hidden message is (Blanco et al, 2005).

4. Steganalysis Prototype

The Steganalysis system consists of two main components; the Steganalysis application and a web spider. Each component has an individual role within the system. The Steganalysis application and the web spider is installed on the user's computer and with the use of an internet connection, the system is connected to the World Wide Web. Figure 5 illustrates the Steganalysis system.

The Steganalysis application allows the user to load images into the system for steganography detection. The application scans the image based on the least significant bit scan method. Once the scan has completed the application will inform the user of the result of the scan. Web Spiders are specialist types of robots or automated programs that search the web. The web spider used within the Steganalysis system downloads images from specific website input by the user. The spider allows the user to download all images on the World Wide Web based on the image file type choice which is also chosen by the user. The user will setup a location where the downloaded images will be saved.

The user must enter a website URL which the spider can connect to. The user must also specify the location on the computer to which the downloaded images can be saved. The user can select the type of images which he/she wants to download. The downloaded images can be viewed within the web spider interface. Once the user clicks on the scan button, the image will be scanned using the least significant bit detection method. The system will then run the appropriate scan on the image.

The result of the scan will be displayed to the user. WebRipper is used to quickly download all the images contained in a specific website and store them to a specific folder on the computer. These will be the images used within the steganalysis system to detect steganography. The steganalysis system is based on the least significant bit steganography technique. The system will detect steganography in images by checking if the least significant bits of the image have been modified. To detect lsb steganography using the steganalysis system, the user must first select an image to scan. This is illustrated in Figure 6. When the user selects an image and clicks on the open button, the image will be displayed as shown in Figure 7.

The user then clicks on the detect button which will run the detection algorithm by calling the extract class. If the system detects the lsb of the image has been modified and the message contains hidden data, a message will be displayed to the user. If no hidden message has been detected, the system will inform the user of the result.

The steganography demonstration part of the system is an extra feature to allow the user to see steganography in action. This feature of the system demonstrates steganography to the user. To do this, the user must first select an image using the JFileChooser as discussed previously. Next, the user must click on the button named Steganography Demonstration which will call the steganographyImageDeconstructor class to convert the image pixel data into an array. This class then calls the steganography demonstration class which will hide a predefined message in a copy of the original image so that no permanent changes are made to the original image. The steganographyImageDeconstructor class will then build a JFrame which will display the original image and the new steganographic image which contains the hidden message.

The result of the scan will be displayed to the user. WebRipper is used to quickly download all the images contained in a specific website and store them to a specific folder on the computer. These will be the images used within the steganalysis system to detect steganography. The steganalysis system is based on the least significant bit steganography technique. The system will detect steganography in images by checking if the least significant bits of the image have been modified. To detect lsb steganography using the steganalysis system, the user must first select an image to scan. This is illustrated in Figure 6.

A number of images were included which we know to be steganographic images. As it is known that this image has been affected by steganography, the system responds with a message to say it has detected steganography. The result is shown in Figure 8. If the result is negative then the popup in Figure 9 is shown. Figure 10 illustrates the steganography demonstration.

5. Conclusion

At present, the system only reveals the existence of a message; it is not concerned with what the message is. This feature could be included within the system. Decryption could also be used within this feature as most steganography programs use encryption to make it more difficult to reveal the message in its original form. In conclusion, the project has been successful in designing and implementing a steganalysis program to detect lsb steganography. Although the steganalysis system could be improved it effectively serves as a basic detection tool which can identify steganographic images.

References

Artz, D. (2001). Digital Steganography: Hiding Data within Data.

http://www.cc.gatech.edu/classes/AY2003/cs6262_fall/digital_steganography.pdf

Blanco, D. et al. (2005). Importance of Steganalysis.

http://cnx.org/content/m13179/latest

Brainos II, A.C. (2006). A Study of Steganography and the Art of Hiding Information. http://www.infosecwriters.com/text_resources/pdf/steganographyDTEC6823.pdf

Callinan, J., Kemick, D. (2006). Detecting Steganographic Content in Images Found on the Internet.

http://www.chromesplash.com/jcallinan.com/publications/steg.pdf

Calpe, A. (2006). Steganography in Images.

http://www.cs.ucf.edu/courses/cot4810/fall04/presentations/Steganography_in_Images.ppt

Cole, E. (2005). Hiding in Plain Sight. Wiley, John & Sons, Incorporated.

Chandramouli, R., Memon, N.D. (2006). Steganography Capacity: A Steganalysis Perspective. http://citeseer.ist.psu.edu/cache/papers/cs/27278/http:zSzzSzwww.ece.stevens-tech.eduzSz~moulizSzstegcap03.pdf/chandramouli03steganography.pdf

Esoteric Archives. (2006). Steganographia (Secret Writing), by Johannes Trithemius.

http://www.esotericarchives.com/tritheim/stegano.htm

Druid. (2006). Steganography Primer: Introduction to Steganography.

http://druid.caughq.org/presentations/Steganography-Primer/

Farid, H. (2006). Detecting Steganographic Messages in Digital Images.

http://www.cs.dartmouth.edu/~farid/publications/tr01.pdf

Fridrich, J., Goljan. M. (2006). Practical Steganalysis of Digital Images – State of the Art.

http://www.ws.binghamton.edu/fridrich/Research/Steganalysis01.pdf

Gupta, S. (2005). All about Steganography.

http://palisade.plynt.com/issues/2005Apr/steganography

Katzenbeisser, S., Petitcolas, F.A.P. (2000). Information Hiding Techniques for Steganography and Digital Watermarking. Artech House Publishers.

Knlghtl0rd. (2005). Steganography: The World of Data Hiding.

http://www.infonomicon.org/text/Research.rtf

Krenn, R. (2006). Steganography and Steganalysis.

http://www.krenn.nl/univ/cry/steg/article.pdf

Marcus, I. (2006). Steganography Detection.

http://www.uri.edu/personal2/imarcus/stegdetect.htm

Morkel, T. et al. (2005). An overview of Image Steganography.

http://mo.co.za/open/stegoverview.pdf

Provos, N., Honeyman, P. (2006). Detecting Steganographic Content on the Internet.

http://niels.xtdnet.nl/papers/detecting.pdf

Roberts, M. (2006). The RGB Colour Model.

http://www.cs.bham.ac.uk/~mer/colour/rgb.html

Si, B. (2004). Introduction to Steganography.

http://www.infosyssec.com/infosyssec/Steganography/menu.htm

Silman, J. (2001). Steganography & Steganalysis: An Overview.

http://www.sans.org/reading_room/whitepapers/steganography/553.php

Queirolo, F. (2006). Steganography in Images.

http://www.cse.buffalo.edu/~peter/cse741/Presentations/Refs/Queirolo.pdf

Westfeld, A., Pfitzmann, A. (2006). Attacks on Steganographic Systems – Breaking the Steganographic Utilities EzStego, Jsteg, Steganos and S-Tools and Some Lessons Learned.

http://www.ece.cmu.edu/~adrian/487-s06/westfeld-pfitzmann-ihw99.pdf

Zhang, T., Ping, X. (2006). A Fast and Effective Steganalytic Technique Against JSteg Like Algorithms. http://citeseer.ist.psu.edu/cache/papers/cs/26891/http:zSzzSzwww.rbfn.comzSzTaoZhangzSzPaperzSzacm_sac_2003_zhang.pdf/zhang03fast.pdf

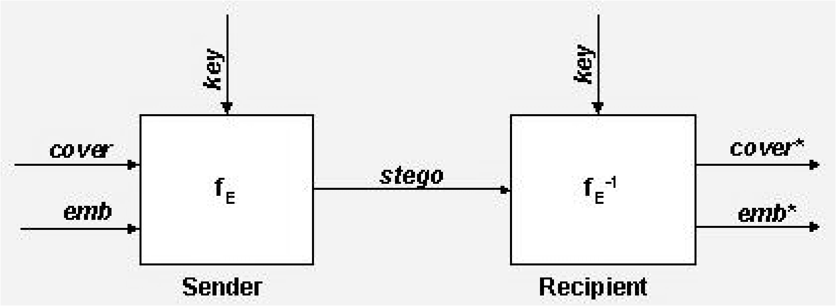
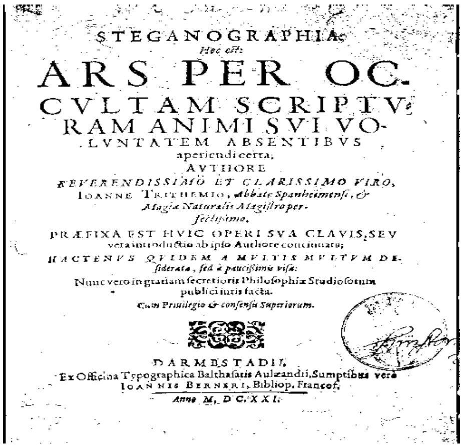


Figure 1. Steganography System (Si, 2004)


STEGANOGRAPHIA:
Hoc eſt:
ARS PER OC-
CVLTAM SCRIPTV-
RAM ANIMI SVI VO-
LVNTATEM ABSENTIBVS
aperiendi certa;
AVTHORE
REVERENDISSIMO ET CLARISSIMO VIRO,
IOANNE TRITHEMIO, *Abbate Spanheimenſi, &*
Magiæ Naturalis Magiſtro per-
fectiſsimo.
PRÆFIXA EST HVIC OPERI SVA CLAVIS, SEV
vera introductio ab ipſo Authore concinnata;
HACTENVS QVIDEM A MVLTIS MVLTVM DE-
ſiderata, ſed à pauciſsimis viſa:
Nunc vero in gratiam ſecretioris Philoſophiæ Studioſorum
publici iuris facta.
Cum Priuilegio & conſenſu Superiorum.

DARMBSTADII,
Ex Officina Typographica Balthaſaris Aulæandri, Sumptibus vero
IOANNIS BERNERI, Bibliop. Francof.
Anno M. DC. XXI.

Figure 2. Steganographia by Trithemius (Esoteric Archives, 2006)

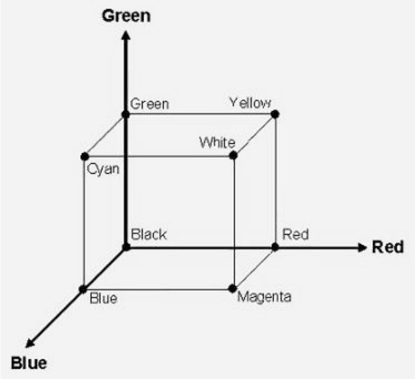


Figure 3. RGB Colour Cube (Roberts, 2006)

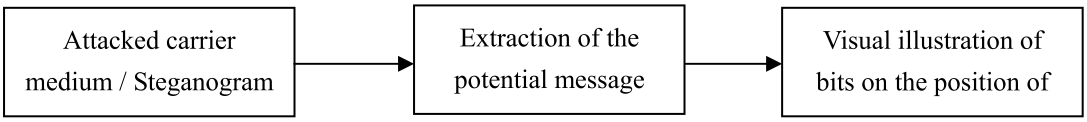

Attacked carrier medium / Steganogram
Extraction of the potential message
Visual illustration of bits on the position of

Figure 4. Filtering Process (Westfeld &Pfitzmann, 2006).

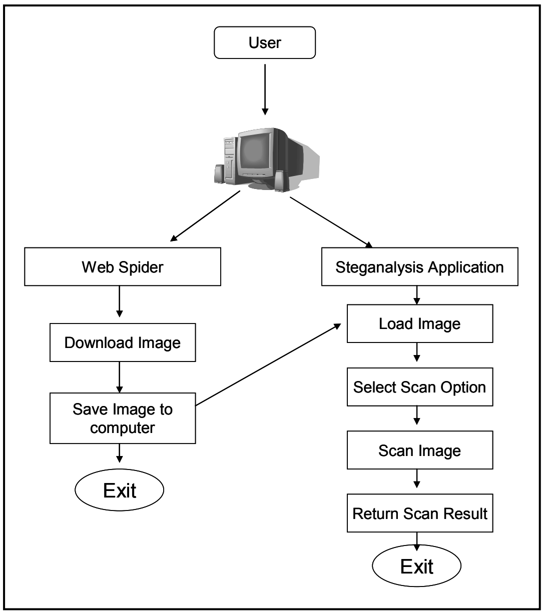
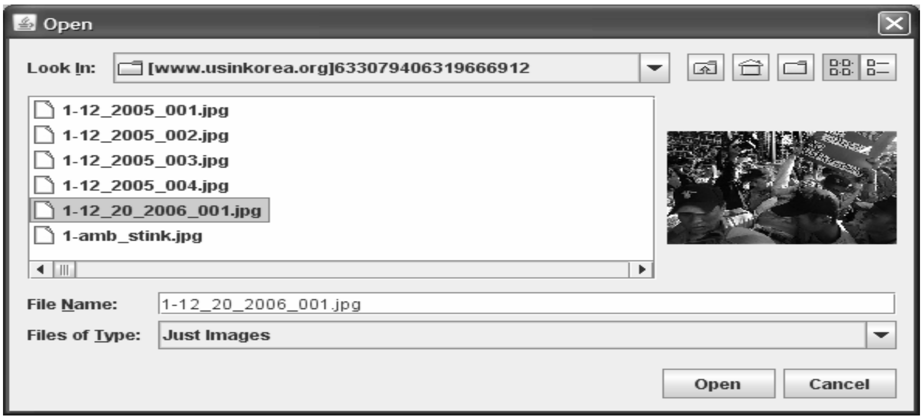


Figure 5. Architecture of Steganalysis System



Look In: [www.usinkorea.org]633079406319666912

1-12_2005_001.jpg
1-12_2005_002.jpg
1-12_2005_003.jpg
1-12_2005_004.jpg
1-12_20_2006_001.jpg
1-amb_stink.jpg

File Name: 1-12_20_2006_001.jpg

Files of Type: Just Images

Figure 6. JFileChooser with image preview.

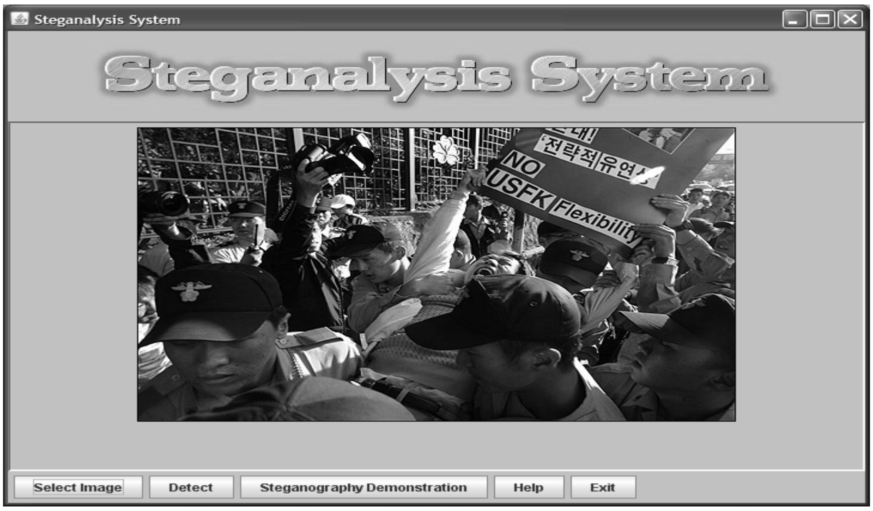
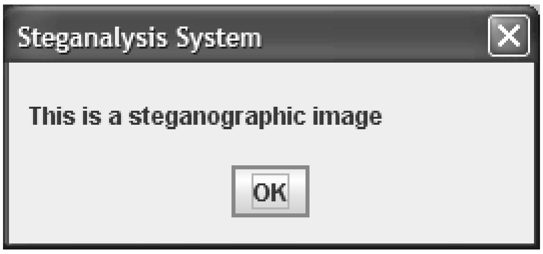
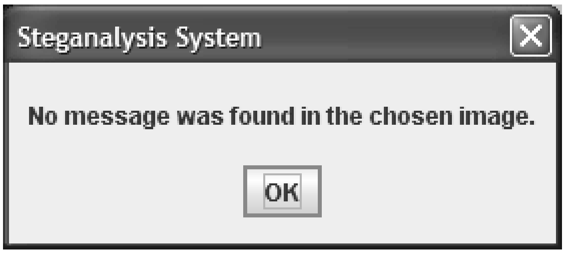


Figure 7. Main GUI displaying selected image.



This is a steganographic image



No message was found in the chosen image.

Figure 8. Detected Steganography Message

Figure 9. No steganography message detected

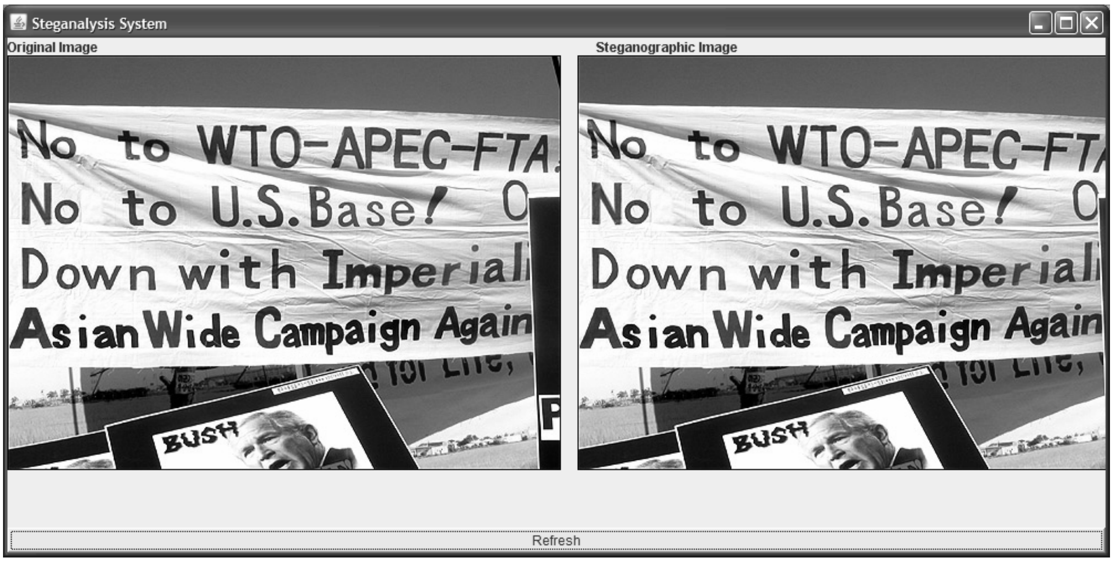


Figure 10. Steganography Demonstration.