A Survey on Secure Image Steganography based on F5 Algorithm

Renu Mishra, Deeksha Mishra, Anshu Ranjan, Harsha Gupta

Abstract – Steganography is a technique which is used for securing the secret information from the illegal activity. Steganalysis is technique of finding the hidden text from the stego image. A Steganalysis based on the DCT value of the image is proposed in this paper. Steganography F5 algorithm is greater secure than other Algorithm. In this paper, we present a steganalytic method that can reliably detect messages (and estimate their size) hidden in JPEG images using the steganographic algorithm F5. The key element of the method is estimation of the cover-image histogram from the stego-image. This is done by decompressing the stego-image, cropping it by four pixels in both directions to remove the quantization in the frequency domain, and recompressing it using the same quality factor as the stego-image. The number of relative changes introduced by F5 is determined using the least square fit by comparing the estimated histograms of selected DCT coefficients with those of the stegoimage. Experimental results indicate that relative modifications as small as 10% of the usable DCT coefficients can be reliably detected. The method is tested on a diverse set of test images that include both raw and processed images in the JPEG and BMP formats.

F5 Steganography Algorithm

I. INTRODUCTION

The Cryptography is used for protecting information security from illegal activity by making message illegible. But in cryptography the data which exist is not hidden. Steganography refers to the technique of hiding data in media (audio, image, video) in order to keep secret (hide) the existence of the information. Steganography hide the data which is existing to be observed. The media which contain with and without information are called the stego image and cover image. Images are the digital media which is widely used and exchanged through the internet. Images are the best cover media to hide secret information inside. Secret information bits are inserted in an area of the cover file (image) that is not to be observed by an eye. Steganography communication system consists of an algorithm for embedding and an extraction. To provide a secret message, the original image is slightly modified by the embedding algorithm. Through which, the stego-image is acquire . Steganography can be used for both illegal and legal purpose. Civilians may use it for protecting privacy while terrorists may use it for spreading terrorism. It is new technique for establishing a secure communication . In past null cipher technique is used as a Steganography communication. But in that method simple algorithm is used for the data transfer. Steganalysis, from an opponent's perspective, is an art of detecting stego image and avoiding from innocent ones. In first step it is need to determine whether a message is hidden within the image. Then find the type of Steganography algorithm used for producing the secret image. Then estimating the length of the message and then try to estimate the message is hidden behind the stego image. This paper will try to give survey on current Steganalysis algorithm . Also discussed proposed work and also given some implementation part (histogram of image).

Secure steganographic algorithms hide confidential messages within other, more extensive data (carrier media). An attacker should not be able to find out, that something is embedded in the steganogram (i.e., a steganographically modified carrier medium).

II. PRESENT WORK

The F5 Algorithm

First step of Steganalysis is to find the image, which is stego image or cover image. Then after try to find the information is whether hidden behind the cover image or not. Next step is to find the true length of the hidden information behind the image. And at last if possible then try to what information is hidden within the cover image. Different types of algorithm are used for the Steganalysis purpose such as Attack on LSB algorithm, Attack on F5 algorithm, Attack on outguess algorithm, Attack on JSteg/JP hide algorithm.

F5 Steganography Algorithm

Step 1: Input one RGB color cover-image.
Step 2: Calculate 2D DCT value of 8*8 image blocks.
Step 3: Apply quantization to DCT coefficients in each block of the image. The quality factor Q is used to build quantization table.
Step 4: Capture a password from the user as a seed to generate random embedding positions in a DCT block (8 x 8).
Step 5: Choose available coefficients to hide messages.
Step 6: Implement entropy coding algorithm.
Step 7: Save the stego-image in JPEG format with a given Q.

F5 Steganalysis Algorithm

F5 Steganalysis attack is mainly used for estimating the true message length. Some of the characteristics of the histogram of DCT coefficients, like the symmetry and monotonicity are retain by the F5 Steganography algorithm. But the F5 Steganography algorithm modifies
the shape of the histogram of DCT coefficients. This drawback of the Steganography algorithm is used for the Steganalysis attack. The procedure for the Steganalysis algorithm (procedure for getting the cover image from stego image). First the stego image is decompressed to spatial domain. After that this decompressed image is cropped by 4 columns from all sides and then re-compressed it using the same quantization parameters as that of the stego image. Before the re-compressed operation a (blurring operation) low pass filter is applied (as a preprocessing step) to remove possible jpeg blocking artifacts from the cropped image. Then the resulting DCT coefficients will provide the estimation of the cover image histograms. Then the probability of none zero AC coefficient being modified, denoted by β, may be estimated by the least square approximation minimizing the square error between the stego image histograms.

**Steps for the F5 Steganalysis algorithm**

**Step 1:** Input the stego image for performing Steganalysis.

**Step 2:** Decompressed the stego image.

**Step 3:** Crop the image by 4x4 column from all sides.

**Step 4:** Apply blurring operation to remove artifacts.

**Step 5:** Then re-compressed the image.

**Step 6:** Count the different histogram value for the stego image and cover image.

**Step 7:** Calculate the difference

\[\text{Difference = stego image value} - \text{cover image value}.\]

**III. FUTURE WORK**

**Compression**

Storing data in a format that requires less space than usual.

Data compression is particularly useful in communications because it enables devices to transmit or store the same amount of data in fewer bits.

Here in this paper we can compress the size of message, or a doc file within an image, audio, or video.

The compression range may lie between the 0-9.

**IV. IMPLEMENTATION**

The algorithm F5 has the following coarse structure:

1. Start JPEG compression. Stop after the quantisation of coefficients.
2. Initialise a cryptographically strong random number generator with the key derived from the password.
3. Instantiate a permutation (two parameters: random generator and number of coefficients).
4. Determine the parameter \(k\) from the capacity of the carrier medium, and the length of the secret message.
5. Calculate the code word length \(n = 2k - 1\).
6. Embed the secret message with \((1, n, k)\) matrix encoding.
   (a) Fill a buffer with \(n\) nonzero coefficients.
   (b) Hash this buffer (generate a hash value with \(k\) bit-places).
   (c) Add the next \(k\) bits of the message to the hash value (bit by bit, xor).

(d) If the sum is 0, the buffer is left unchanged. Otherwise the sum is the buffer’s index \(1, \ldots, n\), the absolute value of its element has to be decremented.
(e) Test for shrinkage, i.e., whether we produced a zero. If so, adjust the buffer (eliminate the 0 by reading one more nonzero coefficient, i.e., repeat step 6a beginning from the same coefficient). If no shrinkage occurred, advance to new coefficients behind the actual buffer. If there is still message data continue with step 6a.

7. Continue JPEG compression (Huffman coding etc.).

**Attack on F5**

We divided our attack on F5 into two separate parts:

(1) Finding distinguishing statistical quantities \(T\) that correlate with the number of modified coefficients.

(2) Determining the baseline values of the statistics \(T\).

In fact, it is not that difficult to find a quantity that changes with embedded message length. For example, the number of coefficients equal to zero increases while the number of remaining non-zero coefficients decreases. Another measure that can be used is the “blockiness” or the measure of discontinuity at the boundaries of the \(8 \times 8\) grid. Actually, the blockiness is likely to increase for any method that embeds message bits by modifying the quantized DCT coefficients of the cover-JPEG image (for example, in we use the blockiness increase as the distinguishing quantity to successfully attack the Optimal Guess. What is difficult, however, is finding the baseline values or their estimates for the distinguishing statistics \(T\) – the original value(s) of \(T\) for the cover image.

In the following subsection, we first analyze how F5 changes the histogram values.

Then, we describe a method for obtaining the estimate of the cover-image histogram from the stego-image. We continue with a detailed description of a detection method that is capable of estimating the message length. Finally, we close Sect. 3 with experimental results and their discussion.

**V. CALCULATION**

**Analysis of Histogram Modifications**

Let \(h(d), d = 0, 1, \ldots\) be the total number of AC coefficients in the cover-image with absolute value equal to \(d\) after the image has been compressed inside the F5 algorithm (Step 2 above). In a similar manner, we denote \(hk(d)\) the total number of AC DCT coefficients corresponding to the frequency \((k, l)\), \(1 \leq k, l \leq 8\), whose absolute value is equal to \(d\). The corresponding histogram values for the stego-image will be denoted using the capital letters \(H\) and \(Hk\).

Let us suppose that the F5 embedding process changes \(n\) AC coefficients. The probability that a non-zero AC coefficient will be modified is \(\beta = n/P\), where \(P\) is the total number of non-zero AC coefficients \((P = h(1) + h(2) + \ldots)\).

**Estimating the Cover-Image Histogram**

Accurate estimation of the cover-image histogram \(h\) is absolutely crucial for our detection method to work. We
first decompress the stego-image to the spatial domain, then crop the image by 4 columns, and recompress the cropped image using the same quantization matrix as that of the stego-image. The resulting DCT coefficients will provide the estimates \( h' \) for our analysis. Because the accuracy of the estimates is the major factor influencing the detection accuracy, we include a simple preprocessing step to remove possible JPEG blocking artifacts from the cropped image before recompressing. We have experimented with several spatial blocking-removing algorithms, but the best results were obtained using a simple uniform blurring operation with a 3×3 kernel \( B, B = 22 = (2^2 - 4e, B = 21 = B23 = B12 = B32 = e, and \) \( Bij = 0 \) otherwise. This low-pass filter helps remove some spurious non-zero DCT coefficients produced by “discontinuities” at the block boundaries, which are in the middle of the 8×8 blocks of the cropped image.

The effect of F5 embedding on the histogram of the DCT coefficient. The main reason why we decided to use histograms of individual low-frequency DCT coefficients rather than the global image histogram is as follows. Even with the low-pass pre-filtering, the spatial shift by 4 pixels introduces some non-zero coefficients in high frequencies due to the discontinuities at block boundaries. And the values that are most influenced are 0, 1, and –1, which are the most influential in our calculations. Individual histograms of low frequency coefficients are much less susceptible to this onset of spurious non-zero DCTs.

We have identified two cases when the estimated histogram obtained using the algorithm described above does not give accurate values. This may occur, for example, when the cover-image sent to F5 has already been saved in the JPEG format with a different quality factor \( Q1 \neq Q2 \), or when the image contains some regular structure with a characteristic length comparable to the block size. Fortunately, both cases can be easily identified and our detection procedure correspondingly modified to obtain accurate results in those cases as well.

**VI. CONCLUSION**

Many steganographic algorithms offer a high capacity for hidden messages, but are weak against visual and statistical attacks. Tools withstanding these attacks provide only a very small capacity. The algorithm F4 combines both preferences:

Resistance against visual and statistical attacks as well as high capacity. Matrix encoding and permittive straddling enable the user to decrease the necessary number of steganographic changes and to equalise the embedding rate in the steganographic. F5 accomplishes a steganographic proportion that exceeds 13% of the JPEG file size. Please understand this result as a friendly provocation for security analysts. On the other hand F5 is able to decrease the embedding rate arbitrarily. The software with its source code is public.

Steganalysis F5 algorithm is mainly used for detecting the secret information hidden by the steganography F5 algorithm. When the steganography F5 algorithm is applied on the image at that time the statsics of the image is changed, based on that Steganalysis F5 algorithm is applied to take the DCT and histogram analysis of the cover image and stego image and compare them based on that we can find the image is stego or not. In Future the implementation of F5 algorithm will be done. When embedding rate is decreased from 10% to 5% then its accuracy is decreased that needs to improve and also need to decrease the processing time for that algorithm. Steganalysis F5 algorithm will be implemented using re-implementation of image instead of double compression.

**REFERENCES**


AUTHOR’S PROFILE

Deeksha Mishra
Persuing her B.Tech. in Computer Science and Engineering from Galgotia’s College of Engineering and Technology Greater Noida (Uttar Pradesh). Her area of interest includes Web Technology, Networking, and Cryptography. She has also worked on Web Based Project “Online Shopping”.

Harsha Gupta
Persuing her B.Tech in Computer Science and Engineering from Galgotia’s College Of Engineering and Technology Greater Noida (Uttar Pradesh). Her area of interest includes Web Technology, Networking, and Cryptography. She has also worked on Web Based Project" Online Shopping".

Anshu Ranjan
Persuing her B.Tech in Computer Science and Engineering from Galgotia’s College Of Engineering and Technology Greater Noida (Uttar Pradesh). Her area of interest include JAVA Programming, Web Technology, Networking and Cryptography. She has also worked on JAVA Programming Based Project “Online Shopping”, "Digisafe" and "Online Attendance System".

Renu Mishra