Volume 1, No. 5, December 2010
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

GENETIC ALGORITHM BASED SUBSTITUTION TECHNIQUE OF IMAGE STEGANOGRAPHY

Samir Kumar Bandyopadhyay*[1], Tuhin Utsab Paul[2] and Avishek Raychoudhury[3]

Department of Computer Science and Engineering, University of Calcutta Kolkata,India
skb1@vsnl.com[1] , tuhin@ieee.org[2] , avishekraychoudhury@gmail.com[3]

Abstract Steganography is the act of hiding a message inside another message in such a way that can only be detected by its intended recipient. Naturally, there are security agents who would like to fight these data hiding systems by steganalysis, i.e. discovering covered messages and rendering them useless. There is currently no steganography system which can resist all steganalysis attacks. The most notable steganalysis algorithm is the RS attack which detects the steg-message by the statistic analysis of pixel values. To ensure the security against the RS analysis, we presents a new steganography based on Genetic Algorithm in this paper. In this paper, we present a novel approach to resolve the remained problems of substitution technique of image steganography. Using the proposed Genetic Algorithm, message bits are embedded into different bits of the pixel grey level values, resulting in increased robustness. The robustness would be increased against those attacks which try to reveal the hidden message and also some unintentional attacks like noise addition as well.

Keywords Steganography, Bit updation, Genetic Algorithm, Least Significant Bit.

INTRODUCTION

The standard and concept of "What You See Is What You Get (WYSIWYG)" which we encounter sometimes while printing images or other materials, is no longer precise and would not fool a steganographer as it does not always hold true. Images can be more than what we see with our Human Visual System (HVS); hence, they can convey more than merely 1000 words. Steganography, the art of hiding messages inside other messages, is now gaining more popularity and is used on various media such as text, images, sound, and signals. However, none of the existing schemes can yet shield against all detection attacks. Using Genetic Algorithms that are based on the mechanism of natural genetics and the theory of evolution, we can design a general method to guide the steganography process to the best position for data hiding.

In recent years, many successful steganography methods have been proposed. Among all the methods, LSB (least significant bit) replacing method is widely used due to its simplicity and large capacity. The majority of LSB steganography algorithms embed messages in spatial domain, such as BPCS, PVD. Some others, such as Jsteg, F5, Outguess, embed messages in DCT frequency domain (i.e. JPEG images). In the LSB steganography, secret message is converted into binary string. Then the least significant bit-plane is replaced by the binary string. The LSB embedding achieves good balance between the payload capacity and visual quality. However, the LSB replacing method flips one half of the least-significant bits. Thus the artifacts in the statistics of the image are easy to be detected.

In this paper we have proposed a Genetic Algorithm approach to make the bit insertion technique more robust by inserting

message bits in different bit level of the pixel grey level values. The layers are selected in pseudo – random method thereby making it more robust against steganolytic attack. The proposed Genetic approach minimises the effect of bit updation on image grey value thereby reducing the risk the statistical stego attack. Moreover only the stego image is sent to the reciever end thereby reducing chances of suspicion.

REVIEW OF STEGANOGRAPHIC ALGORITHMS

Steganographic algorithms can be characterized by a number of defining properties. Three of them, which are most important for image steganographic algorithms, are defined below.

Transparency evaluates the image distortion due to signal modifications like message embedding or attacking. In most of the applications, the steganography algorithm has to insert additional data without affecting the perceptual quality of the host image. The fidelity of the steganography algorithm is usually defined as a perceptual similarity between the original and stego image. However, the quality of the stego image is usually degraded, either intentionally by an adversary or unintentionally in the transmission process, before a person perceives it. In that case, it is more adequate to define the fidelity of a steganography algorithm as a perceptual similarity between the stego image and the original host image at the point at which they are presented to a consumer. In order to meet fidelity constraint of the embedded information, the perceptual distortion introduced due to embedding should be below the masking threshold estimated based on the Human Visual System (HVS) and the host media.

Capacity of an information hiding scheme refers to the amount of information that a data hiding scheme can successfully embed without introducing perceptual distortion in the marked media. In the case of image, it evaluates the amount of possible embedding information into the host image. The embedding capacity is the all included embedding capacity (not the payload) and can be measured in percent (%), bits per image.

Robustness measures the ability of embedded data or watermark to withstand against intentional and unintentional attacks. Unintentional attacks generally include common data manipulations such as lossy compression, digital-to-analog conversion, re-sampling, re-quantization, etc. whereas intentional attacks cover a broad range of media degradations which include addition white and colored noise, rescaling, rotation (for image and video steganography schemes), resizing, cropping, random chopping, and filtering attacks . Also, the robustness of the algorithm is defined as an ability of the data detector to extract the embedded message after common signal processing manipulations. Applications usually require robustness in the presence of a predefined set of signal processing modifications, so that message can be reliably extracted at the detection side.

Image Steganography

Image steganography takes the advantage of limited power of human visual system (HVS). Here, unlike watermarks which embed added information in every part of an image, only the complex parts of the image holds added information [1-2]. Straight message insertion will simply encode every bit of information in the image. More complex encoding can be done to embed the message only in "noisy" areas of the image that will attract less attention [3]. The least significant bit (LSB) insertion method is probably the most well known image steganography technique. The main advantage of this method is that human eye is not able to notice the change; however unfortunately, it is extremely vulnerable to attacks, such as image manipulation. Masking and filtering techniques hide information by marking an image in a manner similar to paper watermarks. By covering a faint but perceptible signal with another to make the first non-perceptible, the fact that the HVS cannot detect slight changes in certain temporal domains of the image was exploited in [14]. Masking techniques are better choices for lossy JPEG images than LSB method because of their relative immunity to image operations such as compression and cropping [2-7]. JPEG image format due to

its good characteristics (having both reasonable quality and small size) is the most common image format for web and local usages. JPEG uses discrete cosine transform (DCT) to transform successive 8×8 pixel blocks of the image into 64 DCT coefficients. Here, LSBs of the quantized DCT coefficients are used as redundant bits. The modification of even a single DCT coefficient affects all 64 image pixels. In some image formats such as GIF, the visual structure of the image exists to some degree in all bit layers of the image. Steganographic systems which modify these formats are mostly vulnerable to visual attacks [8, 9, 13]. However this is not true about the JPEG format. As the modifications happen in the frequency domain rather than spatial domain, there is no visual attack against it.

Recently, several steganographic techniques for data hiding in JPEGs have been developed: JSteg [10], JP Hide&Seek [10], F5 [11], and OutGuess [12]. All these techniques manipulate the quantized DCT coefficients to embed the hidden message.

Substitution Techniques Based Image Steganography

The subatitution based steganographic algorithms were primarily developed for digital images and video sequences. In the past few years, several algorithms for the embedding and extraction of message in images have been presented. All of the developed algorithms take advantage of the perceptual properties of the HVS in order to add a message into a host image in a perceptually transparent manner. Many attacks such as geometrical distortions, spatial scaling are malicious against image steganography algorithms.

The theory of substitution technique is that simply replacing either a bit or a few bits in each sample will not be noticeable to the human eye. This method has high embedding capacity but it is the least robust. It exploits the absolute threshold of vision but is susceptible to attacks.

The obvious advantage of the substitution technique, the reason for choosing this technique, is a very high capacity for hiding a message. Obviously, the capacity of substitution techniques is not comparable with the capacity of other more robust techniques like spread spectrum technique or Discrete Cosine Transformation (DCT) technique that are highly robust but has a negligible embedding capacity.

2.3 Problems of Substitution Techniques of Image Steganography

Like all multimedia data hiding techniques, image steganography has to satisfy three basic requirements. They are perceptual transparency, capacity of hidden data and robustness. Noticeably, the main problem of audio substitution steganography algorithm is considerably low robustness.

There are two types of attacks to steganography and therefore there are two type of robustness. One type of attacks tries to reveal the hidden message and another type tries to destroy the hidden message. Substitution techniques are vulnerable against both types of attacks. The adversary who tries to reveal the hidden message must understand which bits are modified. Since substitution techniques usually modify the bits of lower layers in the samples -LSBs, it is easy to reveal the hidden message if the low transparency causes suspicious.

Also, these attacks can be categorized in another way: Intentional attacks and unintentional attacks. Unintentional attacks like transition distortions could destroy the hidden message if is embedded in the bits of lower layers in the samples -LSBs.

As a result, this paper briefly addresses following problems of substitution techniques of image steganography:

1) Having low robustness against attacks which try to reveal the hidden message.

2) Having low robustness against distortions with high average power.

A. First Problem

One type of robustness that is very critical for security is withstanding against the attacks which try to reveal or extract the hidden message. This paper is to improve this type of robustness. With an intelligent algorithm we hope to reach a

more robust substitution technique, as such, extracting the hidden message become inaccessible to adversary.

Certain way to withstand against these attacks is making more difficult discovering which bits are modified. Thus, the algorithm may not change some sample due to their situations. This selecting will improve the security of the method and robustness of the technique, because if somebody tries to discover the embedded message, he has to apply a specific algorithm to read some bits of samples. But if modified samples are secret, nobody can discover the message. It is remarkable that if we achieve float target bits, it will be novel.

As we know in samples LSBs are more suspicious, thus embedding in the bits other than LSBs could be helpful to increase the robustness. Furthermore, discovering which pixel samples are modified should be uncharted. To reach to the level of ambiguity, the algorithm will not use a predefined procedure to modify the samples but will decide, according to the environment, in this case the host file; as such it will modify indistinct pixel samples of image files, depending on their values and co – ordinate positions. Thus, some of the samples which algorithm determines they are suitable for modifying will modify and other samples may not change. This ambiguity in selecting pixel samples will thus increase security and robustness of the proposed algorithm.

B. Second Problem

A significant improvement in robustness against unintentional attacks -for example signal processing manipulation- will be obtained if an embedded message is able to resist distortions with high average power. To achieve this robustness the message could embed in deeper layers. But, selecting the layer and bits for hosting is critical because selecting higher layer will introduce distortion in pixel grey value. Embedding the message bits in deeper layers absolutely causes bigger error and it will decrease the quality of transparency. Thus, the algorithm which embeds the message bits in deeper layers should modify other bits intelligently to decrease the amount of this error and reserve the transparency.

Predictably, substitution techniques try to modify the bits of samples in accordance with a directive that is defined in algorithm. The target bits are definite, and the amount of resultant noise is not controlled. There may be some better techniques that try to adjust the amount of resultant noise in substitution techniques. These improved algorithms alter other bits else than target bit in sample to decrease the amount of resultant noise. A key idea of the improved algorithm is message bit embedding that causes minimal embedding distortion of the host image.

The basic idea of the proposed algorithm is embedding that cause minimal embedding distortion of the host image. In this approach the amount of resultant noise could be improved since the total noise will be less, when we are able to alter and adjust more samples. This can achieve more transparency and robustness.

PROPOSED APPROACH

Accordingly, there are two following solutions for mentioned problems:

1) The solution for first problem: Making more difficult discovering which bites are embedded by modifying the bits else than LSBs in samples, and selecting the samples to modify privately-not all samples.

2) The solution for second problem: Embedding the message bits in deeper layers and other bits alteration to decrease the amount of the error.

To integrate these two solutions, “embedding the message bits in deeper layers” that is a part of second solution also can satisfy “modifying the bits else than LSBs in samples” of second solution. In addition, when we try to satisfy “other bits alteration to decrease the amount of the error” of second solution, if we ignore the samples which are not adjustable, also “selecting not all samples” of first solution will be satisfied.

Thus, intelligent algorithm will try to embed the message bits in the deeper layers of samples and alter other bits to decrease the error and if alteration is not possible for any samples it

will ignore them. It is clear that the main part of this scenario is bit alteration that it should be done by intelligent algorithms which use genetic algorithms.

The algorithm at sender end and reciever end is :

Sender End

SENDER (Target Text Message, Cover Image)

This function will be used in the sender side to encrypt the TargetTextMessage inside the CoverImage.

Input: This function will take Target Text Message and Cover Image as input.

Output: It will output the encoded StegoImage.

Begin

Step 1. Start

Step 2. Read the Cover Image and Target Text Message.

Step 3. Generate the bit stream of 0's & 1's by taking each character of the Target Text Message one by one and representing their 8 bit binary representation from their corresponding ASCII code.

Step 4. Read the 1st pixel value of the Cover Image & 1st bit of the generated bit stream.

Step 5. Use a intelligent genetic function f(r, c),where r and c are the row number and column number of the last read pixel of the Cover Image, which outputs a integer value from 0 to 7,say 'pos'. The function may return NULL (to make sampling more robust) , in that case consider the next pixel of the host image.

Step 6. Change the 8 bit binary representation of the pixel value (say 'p') by updating (pos+1)th bit(LSB is the 1st bit) with the last read bit from the generated bit stream. Then keeping the last updated bit intact and change the other 7 bits, such that the difference between the updated pixel value and 'p' is minimized. Store the new generated pixel value in the corresponding position.

Step 7. Read the next pixel value of the Cover Image (row major order) & next bit of the generated bit stream. Loop Step 5 and 6 until the generated bit stream is exhausted.

Step 8. Store the Target Text Message length in the free space of image header.

Step 9. Store the Image as Stego-Image and send it to the receiver.

Step 10. Stop.

End

Reciever End

RECIEVER (Stego Image)

This function will be used in the receiver side to decrypt the TargetTextMessage from the StegoImage.

Input: This function will take StegoImage as input.

Output: It will output the decoded Target Text Message from StegoImage.

Begin

Step 1. Start

Step 2. Read the Stego Image. Obtain the size 'S' of the Target Text Message from the free space of the image header of the Stego Image. Set S1=S*8 ,i.e the size of the bit stream.

Step 3. Read the 1st pixel value of the Stego Image. Take an empty bit string say 'M'.

Step 4. Use the intelligent genetic function(used in the Sender end) f(r, c),where r and c are the row number and column number of the last read pixel of the Cover Image, which outputs a integer value from 0 to 7,say 'pos'. The function may return NULL (to make sampling more robust) , in that case consider the next pixel of the host image.

Step 5. Extract (pos+1)th bit(LSB is the 1st bit) of the corresponding 8 bit binary representation of the last read pixel value and concatenate with 'M'.

Step 6. Read the next pixel of the Cover Image (in row major order) .Loop Step 4 and 5 until S1 number of bits are extracted.

Step 7. Regenerate the Target Text Message 'msg' by taking 8 consecutive bits of 'M' at a time and obtaining the character with the ASCII code which is equivalent to the decimal value of the 8 consecutive bits under consideration.

Step 8. Display 'msg' as the decoded Target Text Message.

Step 9. Stop.

End

Genetic Algorithm Approach

Here we propose a new genetic algorithm approach to find the best position for data embedding and also optimize the quality of the steganographic image.

Each substitution matrix S is represented as a chromosome G; G = g0 g1 . . . gN−1 , where N = 2k and gene gi means that the gray value i in C (host image) will be replaced by the gray value gi . Note that there are N ! different chromosomes. In the generic algorithm, some chromosomes are specified as forming an initial population of the first generation. Then, the population of the next generation is created by the following operators, and sieved by a fitness function:

Reproduction: This randomly duplicates some chromosomes for the next generation.

Crossover: This randomly combines the left-hand side of one chromosome, with the right-hand side of another chromosome, to form a new chromosome. The new chromosome must be modified by replacing the repetitive genes with other genes, so that all genes are different, within each chromosome. In this operation the place of target bit embedded is not changed [15 -16].

Mutation: This randomly chooses a chromosome and exchanges any two genes to form a new chromosome. In this operation the place of target bit embedded is not changed.

Fitness function : A chromosome's fitness function is a maximisation or minimisation function in which the optimal value or a predefined cut – off value is attained through several itteration and learning. Here the fitness fuction should select the grey value with minimum deviation from the original grey value of the host image i.e. with maximum transperancy.

The algorithm executes in four main steps as defined below :

A. Alteration

At the first step, message bits substitute with the target bits of samples. Target bits are those bits which place at the layer that we want to alter. This is done by a simple substitution that does not need adjustability of result be measured.

B. Modification

In fact this step is the most important and essential part of algorithm. All results and achievements that we expect are depending on this step. Efficient and intelligent algorithms are useful here. In this stage algorithm tries to decrease the amount of error and improve the transparency. For doing this stage, two different algorithms will be used.

One of them that is more simple likes to ordinary techniques, but in aspect of perspicacity will be more efficient to modify the bits of samples (pixel grey value) better. Since transparency is simply the difference between original sample and modified sample, with a more intelligent algorithm, I will try to modify and adjust more bits and samples than some previous algorithms. If we can decrease the difference of them, transparency will be improved. There are two example of adjusting for expected intelligent algorithm below.

Sample bits are: 00101111 = 47

Target layer is 5, and message bit is 1

Without adjusting: 00111111 = 63 (difference is 16)

After adjusting: 00110000 = 48 (difference will be 1 for 1 bit embedding)

Sample bits are: 00100111 = 39

Target layers are 4&5, and message bits are 11

Without adjusting: 00111111 = 63 (difference is 24)

After adjusting: 00011111 = 31 (difference will be 8 for 2 bits embedding)

Another one is a Genetic Algorithm which the sample is like a chromosome and each bit of sample is like a gene. First generation or first parents consist of original sample and altered sampled. Fitness may be determined by a function which calculates the error. It is clear, the most transparent sample pattern should be measured fittest. It must be considered that in crossover and mutation the place of target bit should not be changed.

C. Verification

In fact this stage is quality controller. What the algorithm could do has been done, and now the outcome must be verified. If the difference between original sample and new sample is acceptable and reasonable, the new sample will be accepted; otherwise it will be rejected and original sample will be used in reconstructing the new stego image file instead of that.

D. Reconstruction

The last step is new image file (stego file) creation. This is done sample by sample. There are two states at the input of this step. Either modified sample is input or the original sample that is the same with host image file. It is why we can claim the algorithm does not alter all samples or predictable samples. That means whether which sample will be used and modified is depending on the status of samples (Environment) and the decision of intelligent algorithm.

4. Test Results

Test case 1 : lena.jpg

Host Image :

Embedded Message : DELHI—COMMONWEALTH—GAMES—2010

Stego Image :

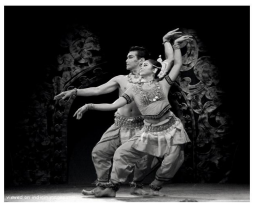
Retrived Message : DELHI—COMMONWEALTH—GAMES—2010

Test case 2 : dance.jpg

Host Image :

Embedded Message : DELHI—COMMONWEALTH—GAMES—2010

Stego Image :



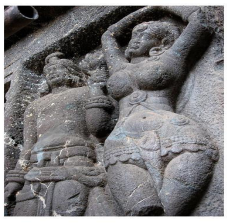
Retrived Message : DELHI—COMMONWEALTH—GAMES—2010

Test case 3 : statue.jpg

Host Image :

Embedded Message : DELHI—COMMONWEALTH—GAMES—2010

Stego Image :



Retrived Message : DELHI—COMMONWEALTH—GAMES—2010

COMPUTATIONAL COMPLEXITY ANALYSIS

In the Sender end, if 'n' is the total no. of bits in the corresponding bit stream of the message, then for inserting each bit it takes time O(n).The intelligent genetic algorithm is independent of 'n' but dependent only on the coordinate indexes of the Cover Image. So, for this genetic algorithm it is also accessed for all the bits of the message, which takes time nearly n*e(e is time for single execution of the genetic algorithm).As both are executed sequentially, it leads to a small linear complexity.

In the Receiver end, the positions for the bits are calculated from the same genetic function and then the message is regenerated sequentially from the extracted bit string, leading to a small linear time complexity.

CONCLUSIONS

A new approach is proposed to resolve two problems of substitution technique of image steganography. First problem is having low robustness against attacks which try to reveal the hidden message and second one is having low robustness against distortions with high average power. An intelligent algorithm will try to embed the message bits in the deeper layers of samples and alter other bits to decrease the error and if alteration is not possible for any samples it will ignore them. Using the proposed genetic algorithm, message bits could be embedded into multiple, vague and deeper layers to achieve higher capacity and robustness.

REFERENCES

[1] Provos N. and Honeyman, P., "Detecting Steganographic Content on the Internet", Center for Information Technology Integration, University of Michigan. Technical Report 01-11, 2001

[2] Sellars D., "An Introduction to Steganography", cs.uct.ac.za/courses/CS400W/NIS/papers99/ dsellars/stego.html

[3] Johnson N. and Jajodia S., "Exploring steganography: Seeing the unseen", Computer, 31, no 2:26-34, Feb. 1998.

[4] Fridrich J., Goljan M., and Hogea D., "Attacking the OutGuess," Proc. ACM Workshop Multimedia and Security2002, ACM Press, 2002.

[5] Provos N., Honeyman P., "Hide and Seek: An Introduction to Steganography", IEEE SECURITY & PRIVACY, MAY/JUNE 2003

[6] Westfeld A., and Pfitzmann A., "Attacks on Steganographic Systems". In: Pfitzmann A. (eds.): 3rd International Workshop. Lecture Notes in Computer Science, Vol.1768. Springer-Verlag, Berlin Heidelberg New York (2000)

[7] Westfeld, A. "Detecting Low Embedding Rates". 5 th Information Hiding Workshop, Netherlands, Oct. 7 .9, 2002

[8] Pik-Wah C., "Digital Video Watermarking Techniques for Secure Multimedia Creation and Delivery", A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Philosophy in Computer Science and Engineering, The Chinese University of Hong Kong, July, 2004

[9] Huang C. and Wu J., "A watermark optimization technique based on genetic algorithms", SPIE Electronic Imaging 2000 San Jose, Jan.2000.

[10] Steganography software for Windows, http://members.tripod.com/steganography/stego/ software.html

[11] Westfeld, A. "High Capacity Despite Better Steganalysis (F5–A Steganographic Algorithm)". Information Hiding. 4th International Workshop. LectureNotes in Computer Science, Vol.2137. Springer-Verlag,Berlin Heidelberg New York, 2001, pp. 289–302

[12] Provos N., "Defending Against Statistical Steganalysis", Proc.10th USENIX Security Symposium, Washington, 2001

[13] Westfeld A., Pfitzmann A., "Attacks on Steganographic Systems". In Proceedings of Information Hiding-Third International Workshop.Springer- Verlag, September 1999.

[14] Bassia P. and Pitas I., "Robust audio watermarking in the time domain", Findings report, Dept. of Informatics, University of Thessaloniki 1998.

[15] J. H. Holland, "Adaptation in natural and artificial systems", Ann Arbor, MI University of Michigan Press 1975.

[16] D. E. Goldberg, "The genetic algorithms in search, optimization, and machine learning", New York