Available online at http://www.ijims.com
ISSN: 2348 – 0343

A Secure Steganography Technique using DCT and Modified RC4

Jahfar C

Department of Computer Science, MCAS Vengara

Abstract

This paper proposed a novel steganography algorithm, which has used sufficient number of highest DCT coefficients of cover image to hide the crucial information. Security has achieved by randomly updating the DCT coefficient values based on the key value given by the user. For generating random sequences from the given key, I have modified Rivest Cipher 4 (RC4) algorithm. The visual quality of the image after performing steganography is the main evaluation criteria and we have used Peak Signal to Noise Ratio(PSNR) and Structural Similarity Index(SSIM) between stego image and cover image to evaluate the performance of proposed algorithm. My proposed algorithms are tested with standard thirty images and the results shows a major advancement in this area.

Keywords: steganography; data hiding; RC4; DCT; information hiding;

Introduction

Information hiding using steganography methods is an important research area, and its application extents to various fields of communication especially in military and medical image transmission, where security of data has more importance. Steganography refers to process of hiding crucial data into digital media without drawing any suspicion. The media with and without hidden data is known as stego and cover media respectively. Generally multimedia data such as videos and images are mainly used as a cover media to hide the data. Reason is that, multimedia data can easily modify without altering the visual quality that much. There are mainly two factors to be considered in steganography process, firstly the payload that we can hide to be more and second, the visual quality degradation due to steganography process should be minimum and imperceptible to the naked eye.

In this paper I proposed a new steganography method, that which hide data in any gray scale image. At the time of data embedding process, the user needs to give a private key value and the person who knows the key value can only extract the hidden information from stego image in future. The proposed data embedding algorithm will take a gray scale image as cover image, a sequence of characters along with private key value. The novelty of this method is that, proposed technique has used Discrete Cosine Transform (DCT) of the cover image to hide the data and the position of DCT coefficient value has find based modified RC4 algorithm.

LITERATURE REVIEW

Cryptography is often used to protect crucial information from unauthorized people by making them into an unreadable form before transmission. These kinds of illegible messages will lead to raise suspicion on a malicious thirty party people and hence lead to the destruction of such messages, even though they doesn’t have any kinds of benefits. Due to this only steganagraphy methods got great attention on the field of information security. Steganography produce a stego image, which looks like a normal one but consist of some hidden data which need to transfer securely. Detailed study of cryptography and steganography is discussed in [1,2].

Various algorithms are proposed for steganography in images and videos. In our study I have considered steganography technique in images only. Based on the domain, these algorithms can be classified into spatial domain or transform domain steganography. In spatial domain, the pixel intensity values are directly modifying, but the transform domain steganography algorithms convert the cover image into frequency domain and then frequency components will modify. Complexity of patial steganography techniques are less as compared to other, but transform domain steganography is more robust against various kinds of signal processing operations such as scaling, noise addition, compression etc.[3]

Least significant bit (LSB) based steganography is one of the most popular and traditional way of data hiding. A detailed analysis has done by K.Thangadurai *et. al* in [4]. The main isuue related with LSB watermarking is that, small distortions on the stego image will spoil the hidden data completely or partially. Researchers are also considers that, LSB steganography technique is less secure one, because an attacker can extract the LSB bits, to get back the hidden data. A modified LSb based steganography method has discussed in [5] by Nadeem Akhta et al. In this, some of the LSBs have been inverted and this will misguide the steganalysis process, hence recovery of hidden data by malicious people will be a tedious task.

Another interesting steganography technique proposed by Ren-Er, Yang, *et al*. under spatial domain category is in [6]. For providing better security for hidden data, before doing steganography the data stream will be encrypted using well known encryption technique called Data Encryption Standard (DES). Here the authors are focused to provide higher level security for hidden data and can be applied where security has attention than any. It is obvious that, the robustness of the proposed method will be less against the unintentional attacks. Because small changes in stego images will update the encrypted data and while decrypting the same, the small updating will lead to erroneous result.

Discrete wavelets transform (DWT) based steganography is discussed in [7]. Here the data will embedded into wavelet coefficients of the cover image. The random selection wavelets have done by graph modeling. The authors proved that they can embed high capacity of data with high robustness. Instead raster scans, the selection of random coefficients in the proposed technique make it more secure.

An integer DCT and affine transformation based steganography is proposed in [8]. Before or after LSB substitution in integer DCT coefficients, some affine transformations are executed on the image, which will maintain the Laplacian-shape-like distribution of DCT coefficients in histogram, and hence information bits can be extracted both completely and safely from stego images.

From the detailed literature review we have analyzed that transform domain steganography techniques are more efficient than spatial domain methods. Our proposed algorithms are based on DCT and the random selection of coefficients has achieved by modified RC4 algorithm [9]. Generally RC4 algorithm has been used for byte stream generation for encryption purpose. Instead of that, here we have modified it to select random DCT coefficients.

PROPOSED SYSTEM

A. *Algorithm-I (Data hiding)*

Data hiding algorithm receives a gray scale image $\mathbf{f_{m\ x\ n}}$, a sequence of characters $\mathbf{d_p}$, that we need to hide in the cover image and the security key $\mathbf{K_c}$. Both sender and receiver has already agreed upon the key, and it’s necessary for extracting the hidden data at any point of time. Data hiding algorithm described below.

```
Algorithm Data_hiding()
{
Input   : Gray scale image ($\mathbf{f_{m\ x\ n}}$), Data ($\mathbf{d_p}$), Key (Kc)
Output  : Stego image ($S_{\mathbf{m\ x\ n}}$ )
1. Find the 2-D DCT of the image $\mathbf{f_{m\ x\ n}}$, say $\mathbf{F_{m\ x\ n}}$.
2. Convert the matrix $\mathbf{F_{m\ x\ n}}$ in a column vector, say FC of size w. (w=m x n)
3. Apply an efficient sorting technique (heap sort) to sort the values of FC in descending order. Assume the discerningly sorted vector termed as $\mathbf{FS_w}$, say the $p^{th}$ DCT value is T.
4. Call the modified RC4 algorithm ( Algorithm-II) with $\mathbf{K_c}$. This will return an integer i , its value between 0 and p. Here p is the number of bytes to be hidden.
5. For each integer i received from Algorithm-II, update $i^{th}$ coefficient as $FC_i=FC_i+(\alpha \times d_j)$, only if $FC_i \geq T$, j=1,2,…,p
6. Convert the column vector $\mathbf{FC_i}$ into 2-D matrix and take the 2-D Inverse DCT. This will be the stego image $\mathbf{S_{m\ x\ n}}$.
7. Send the Stego image ( $\mathbf{S_{m\ x\ n}}$ ) and original image ( $\mathbf{f_{m\ x\ n}}$ ) to the intended recipient. Recipient has already agreed with a key value ( $\mathbf{K_c}$ ) that which sender has used.
}
```

Here, first we need to find the 2D-DCT of the image $\mathbf{f_{m\ x\ n}}$, and that DCT coefficients only we are using to embed the data. The 2 dimensional coefficient matrix $\mathbf{F_{m\ x\ n}}$ is converting into a column vector **FC** and then sorting the values in decreasing order. To sort **FC**, we have used heap sort to reduce the time complexity and hence it will improve time complexity of entire algorithm. The objective of sorting the coefficients is that, I need to get **p** number of highest DCT coefcient values. After sorting we can select the first **p** number of coefficients. Reason for selecting largest coefficients is that the distortion on the stego image Sm x n will be less as compared to updating small coefficient value. From the p number of selected coefficients, the embedding process will proceed by taking the ith coefficient value and updating it by adding the dj th character from the data sequence dp. The value of i is purely determined by algorithm-II. It’s obvious that algorithm-II will return an i value once only. To reduce the amount of change in DCT coefficient value we are multiplying dj with a scaling parameter α, it should be a value between 0 and 1. From trial and error we found that α=0.3 is giving better result. Higher α value will lead the increase the distortion on the stego image and it will lead to suspicious to third party, consequently will lead to destruction of stego image. Usage of less α is not reliable, it will lead to erroneous extraction of hidden data.

B. *Algorithm-II (Modified_RC4)*

This procedure is dedicated to generate random integers between 1 and p. For this purpose I have modified RC4 algorithm. The key value, K_{c} , and the size of data needed to be hidden, p , are the arguments to this procedure. Detailed algorithm described below.

```
Algorithm Modified_RC4()
{
Input : Key ($\mathbf{K_c}$), integer p
Output : Random position (rp) between 1 and p
Declare variables t, q, v, rep, M[p] , flag[p];
for t=1 : p
{
        M[t]=t;
}
for t=1 : p
{
        q=(q+M[t]+$K_c$[t mod c]) mod p;
        swap(M[t], M[q]);
}
rep=true ;
for v=1: p
{
        flag[v]==0 ;
}
```

```
Do while( rep )
{
        t=(t+1) mod p;
        q=(j+M[t] mod p;
        swap(M[t],M[q]);
        rp=M[M[t]+M[q]] mod p
        flag[rp]=true;
        count=0;
        for v=1: p
        {
                If flag[v]==true
                count=count+1;
        }
        if count <>p
                return rp;
        else
                rep=false;

}}//End of the algorithm
```

RC4 algorithm is often used for the purpose of data encryption. In encryption RC4 is used to generate random byte streams according to the given the key value to encrypt a specific data. The byte streams should be in a random order as possible as and the same sequence needs to be generated from the same key during decryption.

In My approach algorithm-II has adopted the core concept of RC4 byte stream generation and it will generate integer values between 1 and p, here p is the number of characters in the data that I need to be embedded. This random value returned by algorithm-II will be used by algorithm-I to select the DCT coefficients for modification.

C. *Data Extraction*

To extract the hidden data we need to take the stego image $\mathbf{S_{m \times n}}$ and original image $\mathbf{f_{m \times n}}$. Then find the 2D-DCT of both stego image and original image, and find out the coefficient positions **i** in the same order by which the insertion has done. Algorithm-II, will return the position **i**, in the DCT coefficient vector after sorting. While doing extraction also I need to pass integer **p** to the Algorithm-II for generating the correct sequence. The extraction process is the reverse operation of insertion and here we need the coefficient value of original cover image. Extraction of $\mathbf{j}^{th}$ character from $\mathbf{i}^{th}$ DCT coefficient value can be expressed as

$\mathbf{d}$j$\mathbf{=(FS_i – F_i)/ \alpha,}$ only **if** $FS_i \geq T$

$\mathbf{j=1,2,\ldots p}$ and $\mathbf{1 \leq i \leq p}$ **.**

$\mathbf{FS_i}$ is the coefficient of stego image and $\mathbf{F_i}$ is the corresponding coefficient of cover image. **T** will be $\mathbf{p}^{th}$ DCT coefficient after sorting in decreasing order. The index **i** has been determined by Algorithm-II and the key value.

RESULTS AND EXPERIMENTAL STUDY

The major focus of researchers working in the field of steganography is that, the amount of data that can be hidden need to increase without reducing the visual quality of the cover image. Major degradation in visual quality of stego images will spoil the objective steganography. The reason is that, malicious people are always concerned about encrypted images or visually degraded images and they have the great intuition that, these types of images consist of some hidden or crucial data. So they will try to extract the data or will destroy the same.

D. *Result Analysis*

Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) can be used as a parameter to measure the visual quality degradation on the cover image after embedding process.

PSNR: Ratio between the maximum possible power of the signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel (dB) scale.

$$\text{PSNR}= 10. \log_{10}(255^2 /E) \text{ dB}$$

Where E is mean square error (MSE), E can be defined as

$$E = \frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}[f(i,j) - f'(i,j)]\,2$$

f(i, j) is the pixel value of the cover image and f ′(i, j) is the pixel values of stego image.

SSIM: Structural Similarity Index is a method for measuring the similarity between two images. Structural Similarity Index is measuring based on a reference image here we will take the cover image as reference. SSIM is designed to improve on traditional methods like peak signal-to-noise ratio (PSNR) and mean squared error (MSE), which have proven to be inconsistent with human eye perception.

For the experimental study I took standard images from data set managed by University of British Columbia[10]. The data set is in RGB form and we have considered randomly selected thirty images including well know images such as lenna, airplane etc, and those are converted into gray scale image of size 512 by 512 pixels.

There will be 262144 pixels in every images and so the same number of DCT coefficients also. For embedding purpose, I choose the random sequence of characters with varying the size from 25 characters to 65 characters. At the time of experiment, I choose a random character sequence of length **l** and by using the proposed embedding technique the same sequence has embedded in all thirty images. The value of **l** has varied from 25 to 65. In all cases the value **α** has taken as **0.3**. **α** has fixed after trial and error.

The average PSNR obtained for original cover image stego image after data hiding process has shown in **Figure 1**. The result shows that, after adding 65 character length sequence also the PSNR value is greater than 46. The PSNR value mentioned in all over the paper is in decibel(dB).

The SSIM value obtained between cover image and stego image is shown in **Figure 2.** It clearly depicts that, the SSIM value will decrease when the number characters that which we are inserting into the cover image is increasing. After inserting 65 characters also the SSIM value obtained is more than 0.80. If two images does not have any structural dissimilarity SSIM value will be 1.

E. *Comparative Study*

My proposed method has compared against two well known steganography, LSB steganography [4] and DWT based steganography [7]. The PSNR value comparison is shown in **Table 1.**

F. *Experimental Results*

I have experimented thirty gray scale images. The size of data stream varies from 25 to 65 with an interval of 5. A result obtained for Lena image after embedding 25 characters and 65 characters respectively shows in **Figure 3(a) and Figure 3(b)** respectively. The characters are purely in random instead of any meaningful sentences. I have used key value 'RESEARCH' to do all experiments and it will not influence the PSNR or SSIM values of the stego image.

CONCLUSION

In this paper I have proposed a novel steganography algorithm, the experimental results shows that it will perform better than the well known steganography methods in use. Rather than this, I have give more importance to the security, and I modified RC4 algorithm to find the position to hide the data. This will resist brute force attack for data extraction by malicious people. PSNR and SSIM is used as the parameter to evaluate the proposed algorithm, and experiments are done in standard image set.

REFERENCES

[1] Bloisi, Domenico Daniele, and Luca Iocchi. "Image based steganography and cryptography." VISAPP (1). 2007.

[2] Raphael, A. Joseph, and V. Sundaram. "Cryptography and Steganography- A Survey." International Journal of Computer Technology and Applications 2.3 (2011).

[3] Sravanthi, Ms GS, et al. "A Spatial Domain Image Steganography Technique Based on Plane Bit Substitution Method." Global Journal of Computer Science and Technology Graphics & Vision, 12 (15) (2012).

[4] Thangadurai, K., and G. Sudha Devi. "An analysis of LSB based image steganography techniques." Computer Communication and Informatics (ICCCI), 2014 International Conference on. IEEE, 2014.

[5] Akhtar, Nadeem, Shahbaaz Khan, and Pragati Johri. "An improved inverted LSB image steganography." Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on. IEEE, 2014.

[6] Ren-Er, Yang, et al. "Image Steganography Combined with DES Encryption Pre-processing." Measuring Technology and Mechatronics Automation (ICMTMA), 2014 Sixth International Conference on. IEEE, 2014.

[7] Thanikaiselvan, V., and P. Arulmozhivarman. "High security image steganography using IWT and graph theory." Signal and Image Processing Applications (ICSIPA), 2013 IEEE International Conference on. IEEE, 2013.

[8] Song, Xianhua, Shen Wang, and Xiamu Niu. "An integer DCT and affine transformation based image steganography method." Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2012 Eighth International Conference on. IEEE, 2012.

[9]. Xie, Jian, and Xiaozhong Pan. "An improved RC4 stream cipher." Computer Application and System Modeling (ICCASM), 2010 International Conference on. Vol. 7. IEEE, 2010.

[10]. http://sipi.usc.edu/database/ (*accessed on Janvary-2015*)

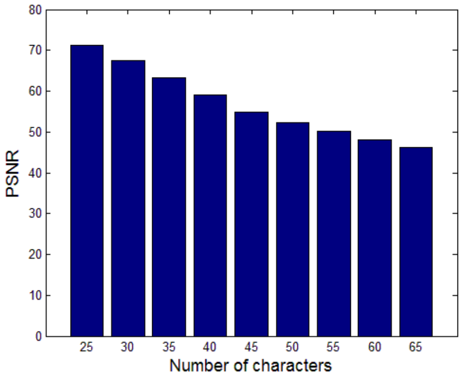


Figure 1. Number of characters Vs PSNR

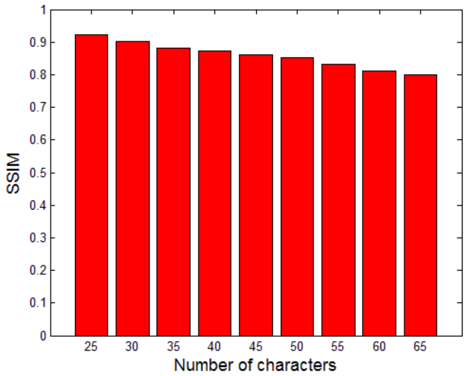
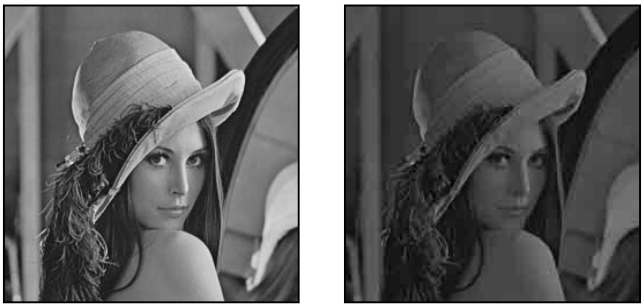


Figure 2. Number of characters Vs SSIM



(a) (b)

Figure 3. Stego images

Table 1. Comparison of PSNR values

Number of characters inserted	LSB	DWT	DCT (Proposed)
	PSNR		
25	56.21	63.32	71.32
30	52.26	61.21	67.52
35	49.28	60.02	63.22
40	46.77	54.87	59.12
45	45.31	50.21	54.81
50	42.19	48.45	52.27
55	40.01	45.22	50.23
60	38.99	42.76	48.15
65	36.22	40.01	46.21