ISSN: 0976-3031

Available Online at http://www.recentscientific.com

International Journal of Recent Scientific Research
Vol. 6, Issue, 6, pp.4800-4804, June, 2015

International Journal of Recent Scientific Research

RESEARCH ARTICLE

IMAGE STEGANOGRAPHY – AN OVERVIEW

Neelam Suryakant Chavan

Lecturer, Instrumentation, BGIT, Mumbai

ARTICLE INFO

Article History:

Received 2nd, May, 2015
Received in revised form 10th, May, 2015
Accepted 4th, June, 2015
Published online 28th, June, 2015

Key words:

Steganography, Imperceptibility, Robustness, Security, Cover, Entropy, Bit data hiding, information, Payload, Stego-image, attacker, data embedding, Cryptography, Transform Domair Compression.

ABSTRACT

There exist a wide range of protocols for hiding message in images. However, without leaving any apparent evidence of image alteration, security and robustness will be the key attributes of any particular technique. Many attacks to security constitute a first step towards performing attacks to robustness. In this paper, we demonstrate an algorithm to make the data embedding process as robust as possible. Starting form proper selection of images, blocks within the image and coefficients within the block this algorithm gives an idea of making the embedding process robust. Experimental results show an improvement as we follow this algorithm.

INTRODUCTION

Digital communication has become an integral part of infrastructure now a day. Lot of applications are Internet-based and in some cases it is desired that the communication be made secret. Two techniques are available to achieve this goal: first is cryptography, where the sender uses an encryption key to scramble the message, this scrambled message is transmitted through the insecure public channel, and the reconstruction of the original, unencrypted message is possible only if the receiver has the appropriate decryption key. Second method is Steganography, where the secret message is embedded in an image. Using this technology even the fact that a secret is being transmitted has to be secret [2].

Steganography is the art and science of communicating in a way, which hides the existence of the communication. Steganography or Stego as it is often referred to in the IT community, literally means, "covert writing" which is derived from the Greek language. In a digital world, Steganography and Cryptography are both intended to protect information from unwanted parties. Both Steganography and Cryptography are excellent means by which to accomplish this but neither technology alone is perfect and both can be broken. It is for this reason that most experts would suggest using both to add multiple layers of security. The algorithm demonstrated in this paper will give an idea of selection of proper image for embedding the information as well as making the data hiding process more robust.

Requirements of Steganographic data hiding

There are three requirements depending for the purpose of Steganographic data hiding:

1. ***Capacity (Pay Load)*:** It is an important factor when a lot of information is to be embedded into a cover image. For example the personal data and the diagnosis could be embedded into medical images. Another example could be embedding personal information into finger print image.
2. ***Imperceptibility*:** It is important when a secret communication occurs between two parties and the fact of a secret communication is kept to be secret. For example information exchange required for credit card transactions.
3. ***Robustness*:** Watermarking, fingerprinting and all copyright protecting applications demand robust steganographic method, i.e. where the embedded information cannot be removed without serious degradation of the image

Corresponding author:* **Neelam Suryakant Chavan
Lecturer, Instrumentation, BGIT, Mumbai

Selection of image for data hiding

When we consider digital image as a cover (image in which data is to be embedded), it is important that the Stego-image (image along with hidden information) does not contain any easily detectable attributes due to message embedding. A third party could use such attributes as an indication that a secret message is present.

Another important factor is the choice of the Cover-image. The selection is at the discretion of the sender who sends the message. The sender should avoid using Cover-images that would be easy to analyze for presence of secret messages. For example, one should not use popular charts, images with large areas of uniform color, images with only a few colors, and images with a unique semantic content. Scanned photographs or images obtained with a digital camera contain a high number of colors and are usually recommended and considered safe for Steganography. Some Steganography experts recommend grayscale images as the best cover-images [4].

Encoding techniques

The steganographic algorithms can be divided into two groups: Spatial/Time domain and Transform domain techniques. Least Significant Bits (LSB) modification techniques are easy way to embed information but they are highly vulnerable to even small cover modifications. An attacker can apply signal processing techniques to destroy the embedded information. Compression of images may create problems and there may be total loss of information. The transform domain methods operate in the Discrete Cosine Transform (DCT), Fourier Transform (FT) or Wavelet Transform (WT) domains of the host signal. Transform domain cover modification has advantages over spatial domain cover modification. It is robust to attacks such as compression, cropping and some image processing attacks and it is imperceptible to human sensory system therefore more undetectable.

Data embedding is done in the Transform domain, with a set of transform coefficients in the low and mid frequency bands selected as possible candidates for embedding (these are preserved better under compression attacks than high frequency coefficients)[3]. Most of the embedding methods require side information about the hiding locations to be sent to the decoder, which reduces the size of the payload. In contrast, the method prescribed in this paper uses property of an image block for selection of that block for embedding the information. The use of local criteria for deciding where to embed is found to be crucial for maintaining image quality under high volume embedding. The decoder does not have explicit knowledge of the locations where data is hidden, but employs the same criteria as the encoder to guess these locations. The distortion due to attacks may now lead to insertion errors. There may be synchronization problem between encoder and decoder, which has to be properly handled by proper coding of text and error correcting mechanisms.

Embedding Algorithm

Text Information processing

1. Read a text file from which information is to be embedded.
2. Encode the text information.
3. Add redundancy for error correction into the encoded information.
4. Make a series of bits information (BI) ready for embedding.

Image Processing

5. Read Cover image.
6. Divide image into 8 x 8 blocks.
7. Take DCT_2 of each block.
8. Calculate Entropy of each block.
9. Calculate Mean Value of Entropy (MVE) of all blocks.
10. Check number of blocks having Entropy greater than MVE and treat them as Valid Blocks (VB).
11. Dividethe DCT_2 coefficients by Quantization Matrix (QM)
12. Find the Valid (non-zero) Coefficients (VC) in each VB by reading them in zigzag manner.
13. If the total number of VC are less than number of BI then image is not suitable for embedding the information.
14. Embed BI in VC of all VB in zigzag order (except DC component). Range of coefficients in middle frequency may be chosen.
15. Multiply all quantized coefficients by QM.
16. Take $IDCT_2$ of all blocks.
17. Reconstruct the image as Stego-image.
18. Write / Transmit image.

Now let us see the processing of embedding in detail.

Transform Domain

While transforming domain, first cover image is divided into 8 x 8 blocks. These blocks are then converted to Discrete Cosine Transform (DCT_2) coefficient blocks using Discrete Cosine Transform.

Two-dimensional DCT of an M-by-N matrix A is defined as follows.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m-1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N-1 \end{cases} \quad (1)$$

The resultant 8 x 8 matrix of DCT_2 coefficients and the way in which it is scanned from coefficient_0 to Coefficient_63 and read while embedding the information is as shown in Fig. 1. Here coefficient_0 is called as DC coefficient. Frequency variations in horizontal and vertical direction are as shown inFig. 2.

Fig. 3 shows categorization of DCT2 coefficients as low frequency (F_L), middle frequency (F_M) and high frequency (F_H).

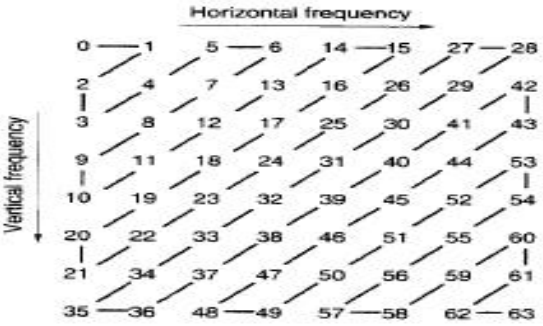


Fig.1 Zigzag Scan of DCT2 coefficients

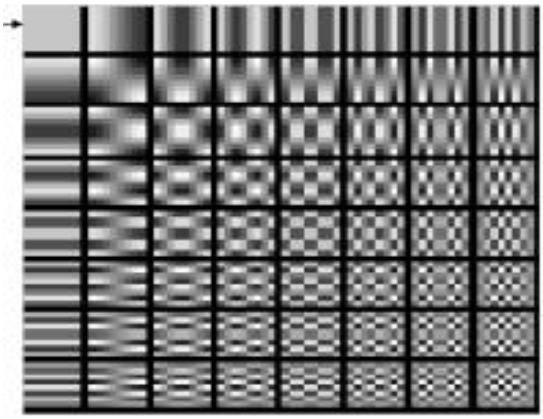


Fig. 2 Frequency Variation in DCT2

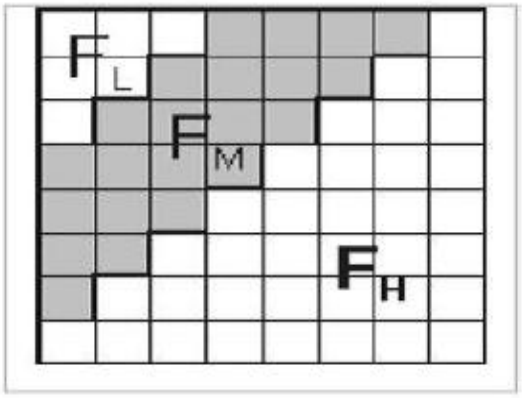


Fig. 3 Middle Frequency Coefficients

Level Entropy Thresholding (ET) is the method used for deciding whether or not to embed data in each block depending on the entropy, or energy, block can be selected for embedding the information [3].

The entropy (E) of the blocks is computed as follows:

$$E = \sum_{i=1}^{i=8} \sum_{j=1}^{j=8} \left| C_{i,j} \right|^2 \quad \text{- (2)}$$

Where C i, j are the DCT2coefficients in a block.

It is important to note that the DC (i = j = 1) coefficient is neither used for Entropy calculation nor for information embedding. This is because JPEG uses predictive coding for the DC coefficients and hence, any embedding induced distortion would not be limited a single 8 x 8 block. The blocks whose energy is greater than a predefined threshold are selected for information embedding. Process of ET is as shown in Fig.4 and Fig.5 for the images, 'Flower' and 'Finger' respectively.

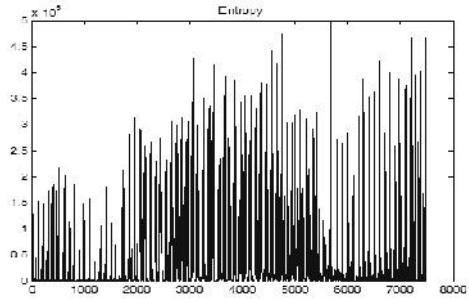
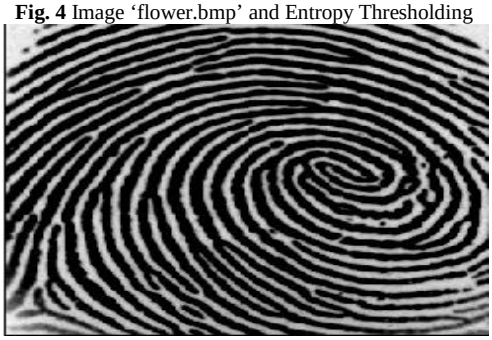


Fig. 4 Image 'flower.bmp' and Entropy Thresholding



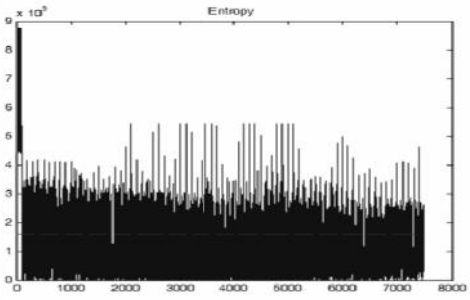


Fig. 5 Image 'finger.bmp' and Entropy Thresholding

Table 1

Image	ME	VB	PSNR
Flower	34073	1784	60.95
Finger	125216	3274	58.32

Table 1 shows our experimental result. It gives comparison of three typical images with respect to MVE, number of VB for embedding, MSE and Peak Signal to Noise Ratio (PSNR). Image flower' is having very low variations of pixel values,

therefore having minimum ME as well as less number of VB. On the other hand, image 'finger' is having very high variations of pixel values and therefore maximum ME as well as more number of VB. Table 1 clearly shows that image 'finger' is best suitable candidate for hiding information for the same value of data hiding capacity. The graphs of block entropy of all the blocks for all three images are as shown in Fig. 4, and Fig. 5 for images, 'flower', and ' finger' respectively.

Our further experimentation is on an image 'finger'. As we increase the value of ET the numbers of valid blocks of embedding are reducing with increase in PSNR. The results shown in Fig. 6. and Fig. 7. gives an idea about effect of increase in threshold value of number of valid blocks and PSNR. In order to increase PSNR one should increase the value of ET. However, it reduces the valid blocks and in tern pay load or embedding capacity. Wider band of quantized middle frequency coefficients in valid block and keeping the close eye on PSNR is the solution to increase the pay load.

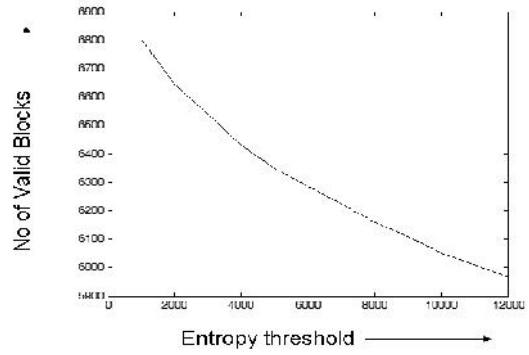


Fig. 6 Number of valid blocks image finger

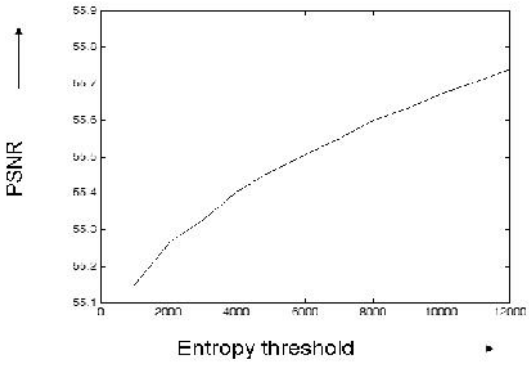


Fig. 7 PSNR for image 'finger'

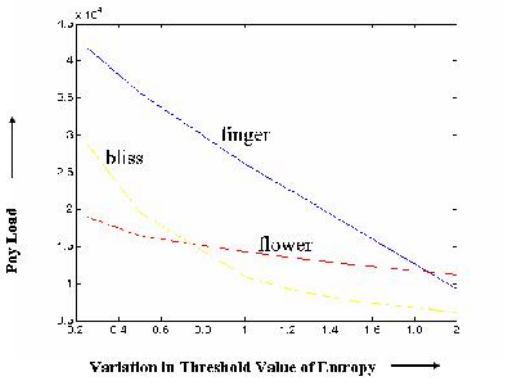


Fig. 8 Effect of ET Variation on VC

Pay load result shown in Fig. 8. gives an idea about selection of proper image for embedding. As image 'bliss' gives drastic decrease in pay load as we increase ET from 0.2 MVE to 2.0 MVE, hence not suitable for heavy pay load applications.

Quantization Matrix

Although in this paper we explain the technique on grayscale images, it can be extended to color images in a straightforward manner. We start with a short description of the JPEG compression algorithm. In JPEG compression, the image is first divided into disjoint blocks of 8×8 pixels. For each block (with integer pixel values in the range 0−255), DCT2 is calculated, producing 64 DCT coefficients. Let us denote the i^{th} DCT coefficient of the k^{th} block as $d_k(i)$, $0 \le i \le 64$, $k = 1,T$, where T is the total number of blocks in the image. In each block, all 64 coefficients are further quantized to integers $D_q(i)$ using the JPEG quantization matrix Q as shown in equation (3).

$$D_q(i) = integer\_round\left(\frac{d_k(i)}{Q(i)}\right) \quad \text{- (3)}$$

Embedding the information

For embedding the information we are selecting the blocks having entropy higher than the MVE for all blocks of the image. Within the block we are selecting the quantized coefficients having non-zero values, and are within the middle frequency range, excluding DC component and few low frequency components. For hiding data we are quantizing the values of the coefficients with the smallest possible value, so that image has less possible perceptual and statistical degradation (less MSE). However, at the same time we have to make sure that the change persists through the reverse process, which is essential for error free extraction of the embedded information

Limitations

Embedding in DCT domain is simply done by altering the DCT coefficients, for example by changing the least significant bit of each coefficient. One of the constraints of embedding in DCT domain is that many of the 64 coefficients are equal to zero, and changing too many zeros to non-zeros values will have an effect on the compression rate. That is why the number of bit one could embed in DCT domain, is less that the number of bits one could embed by the LSB method. Also the embedding capacity becomes dependent on the image type used in the case of DCT embedding, since depending on the texture of image the number of non-zero DCT coefficients will vary. Although changing the DCT coefficients will cause unnoticeable visual artifices, they do cause detectable statistical changes.

CONCLUSION

1. Entropy thresholding gives better perceptual quality of image and system becomes more secured as it avoid suspicious view of the attacker.
2. Instead of fixed DCT2 coefficient the image adaptive selection of blocks and coefficients will increase the

security. Even the decoder does not have explicit knowledge of the locations where data is hidden, but employs the same criteria (shared with the decoder) as the encoder to guess these locations.

3. Encryption of message before embedding gives additional security.
4. Added redundancy in information bits gives error free recovery of hidden data at the receiver.
5. Quantization matrix plays an important role in increasing the robustness and reducing the pay load.
6. In this work we have done practical observations as well as statically observations while choosing the parameters.

References

1. "High-Volume Data Hiding in Images: Introducing Perceptual Criteria into Quantization Based Embedding." K.Solanki, N.Jacobsen, S. Chanderasekaran, U.Madhow and B.S.Manjunath.
2. Anderson, R. J. – Petitcolas, F. A. P., "On the Limits of Steganography," *IEEE Journal of Selected Areas in Communications*," 16(4) pp. 474-481, May 1998. Special Issue on Copyright & Privacy Protection, ISSN 0733-8716.
3. Kaushal Solanki, Noah Jacoben, Upamany Madhow,B.S.ManjunathandShivkumar Chandrasekaran,
4. "Robust Image- Adaptive Data Hiding Using Erasureand ErrorCorrection," *IEEE Trans. Image Processing*, vol. 13, No. 12, pp. 1627–1639, Dec.2004.
5. Jessica Fridrich, Miroslav Goljan and Rui Du,"Steganalysis Based on JPEGCompatibility,"Proc. SPIE vol. 4518, p. 275-280, Multimedia systems and Application IV, Nov. 2001.

How to cite this article:
Neelam Suryakant Chavan., Image Steganography – An Overview. *International Journal of Recent Scientific Research Vol. 6, Issue, 6, pp.4800-4804, June, 2015*

****