Towards Robust Image Steganography

Eldho Thariyan

Student, Dept. of Dual Degree Computer Applications, Sree Narayana Guru Institute of Science and Technology, N.Paravur, Kerala, India

---

Abstract - *The security of a steganography system is defined by our ability to detect it. It is of nothing unexpected then that steganography and steganalysis both rely intensely upon the exactness and strength of our finders. This is especially true when real-world data is considered, due to its heterogeneity. The difficulty of such data manifests itself in a penalty that has periodically been reported to affect the performance of detectors built on binary classifiers; this is known as cover source mismatch. The current strong JPEG steganographic calculations are working with the side data about JPEG pressure quality factor of a misfortune channel Nonetheless, gigantic late exploratory outcomes uncover that the presentation of the present strong JPEG steganographic calculations is poor if the side data is obscure. At that point, another strong JPEG steganographic calculation is proposed dependent on the investigation result. A progression of tests are directed on 10,000 pictures from BOSS base picture library. The outcomes show that the proposed strategy can oppose JPEG pressure effectively with adequate protection from steganalysis measurable location. In this paper, we propose a novel picture steganography system that is vigorous to such channels. Specifically, we first acquire the channel compacted form (i.e., the channel yield) of the first picture. Different tests are directed to show the adequacy of the proposed system for picture steganography hearty to JPEG pressure.*

1. INTRODUCTION

DATA hiding is a technique of embedding secrets into the digital media imperceptibly, which can be categorized into watermarking and steganography according to different applications. Watermarking is the process of marking the digital media for copyright protection, Early image steganographic methods adjust the value of the pixel (or coefficient) either by following a specific statistical model or reducing the modification caused by data embedding The authors in uniformly spread out the changes over all the Discrete Cosine Transformation (DCT) coefficients to resist the statistical attacks. In, the directions of modification are fully exploited to achieve high embedding efficiency. In, the DCT coefficients are categorized into four bands with different embedding rates. The stego-images generated by these schemes can be easily detected using modern steganalysis tools. Recent works on image steganography focus on syndrome trellis coding (STC) based data embedding. In these schemes, different distortion functions are designed to measure the distortion caused due to the data embedding. The STC seeks a solution to minimize such distortion, which achieves relatively good performance in terms of resisting the steganalysis tools.

The 'best detector' is a loose term because our ability to detect steganalysis relies heavily on our knowledge of a great number of variables defining various practicalities of the problem. Unlike cryptography where Kirchhoff's principle is sufficient to encompass all of the relevant knowledge about the problem, in steganography and steganalysis similar conditions only work for covers reduced to purely theoretical constructs. This has not, however, stopped the field from trying to improve the practical tools. Much of this improvement has come from the side of steganography and the consensus is that the practical methods here are more advanced than those in steganalysis. In steganalysis, research efforts have largely been focused on improvements in feature representations. The first representations were based on structural information, but modern representations encompass more general statistics and are used as features in conjunction with machine learning classifiers. With regards to the words "strong steganography", it is normal to consider "vigorous watermark". The powerful watermarking innovation has been investigated for over two decades, and a few sorts of strong watermarking techniques, (for example, spread range tweak, highlight based calculation, and scale invariant component change) are proposed to diminish the impact of different lossy procedure on spread picture object. As a rule talking, the need motivation behind hearty watermarking is to ensure the presence of installed data against the different assaults, (for example, JPEG pressure, turn, trimming, down-inspecting, etc.), and the culmination of the inserted data is on the subsequent stage.

2. EXISTING SYSTEM

The past decade has witnessed a surge of research activity in multimedia information hiding, targeting applications such as steganography, digital rights management, and document authentication. Various works are accessible in the writing related with the Image Steganography. Early image steganographic methods adjust the value of the pixel (or coefficient) either by following a specific statistical model or reducing the modification caused by data embedding. The authors in uniformly spread out the changes over all the Discrete Cosine Transformation (DCT) coefficients to resist the statistical attacks. In, the directions of modification are fully exploited to achieve high embedding efficiency.

In recent years, with the development of mobile communication technology, many social Medias such as Facebook, WeChat, and Instagram transmit enormous images through intelligent mobile terminal. JPEG compression is always applied on the images of social Medias after considering the bandwidth, tariff, traffic and other restrictions of intelligent mobile terminal.

3. PROPOSED SYSTEM

We propose a structure for concealing enormous volumes of information in pictures while causing negligible perceptual corruption. The embedded data can be recovered successfully, without any errors, after operations such as decompression, additive noise, and image tampering. The proposed strategies can be utilized for applications that require high-volume inserting with heartiness against certain non-noxious assaults. The concealing strategies we propose are guided by the developing writing on the data hypothesis of information covering up. Specifically, we use a code on the hidden data that spans the entire set of candidate embedding coefficients, and that can correct both errors and erasures. The subset of these coefficients where the encoder doesn't implant can be treated as deletions at the encoder. Inclusions currently become mistakes, and cancellations become eradications (notwithstanding the deletions previously speculated effectively by the decoder, utilizing indistinguishable nearby measures from the encoder). While the main role of the code is to take care of the synchronization issue, it additionally gives strength to mistakes because of assaults.

Two strategies for applying neighborhood measures are thought of. The first is the square level Entropy Thresholding (ET) technique, which chooses whether or not to insert information in each square (regularly 8X8) of change coefficients, contingent upon the entropy, or vitality, inside that square. The second is the Selectively Embedding in Coefficients (SEC) method, which decides whether or not to embed data based on the magnitude of the coefficient. Reed-Solomon (RS) codes are a natural choice for the block-based ET scheme, while a “turbo-like" Repeat Accumulate (RA) code is employed for the SEC scheme. We can conceal high volumes of information under both JPEG and AWGN assaults. In addition, the shrouded information additionally endures wavelet pressure, picture resizing and picture altering assaults. Our purpose is to generate an intermediate image whose channel compressed version is exactly the same as the stego-image. To this end, we first obtain the stego-image by data embedding on the channel compressed original image using any of the existing JPEG steganographic schemes. Then, we propose a coefficient adjustment scheme to produce the intermediate image based on the stego-image and the original image. This scheme ensures that the channel compressed version of the intermediate image is exactly the same as the stego-image.

4. METHODOLOGY

We propose a framework for hiding large volumes of data in images while incurring minimal perceptual degradation. The embedded data can be recovered successfully, without any errors, after operations such as decompression, additive noise, and image tampering. The proposed methods can be employed for applications that require high-volume embedding with robustness against certain non-malicious attacks. The hiding methods we propose are guided by the growing literature on the information theory of data hiding. The key novelty of our approach is that our coding framework permits the use of local criteria to decide where to embed data. In order to robustly hide large volumes of data in images without causing significant perceptual degradation, hiding techniques must adapt to local characteristics within an image. Specifically, we use a code on the hidden data that spans the entire set of candidate embedding coefficients, and that can correct both errors and erasures. The subset of these coefficients in which the encoder does not embed can be treated as erasures at the encoder. Insertions now become errors, and deletions become erasures (in addition to the erasures already guessed correctly by the decoder, using the same local criteria as the encoder). While the primary purpose of the code is to solve the synchronization problem, it also provides robustness to errors due to attacks.

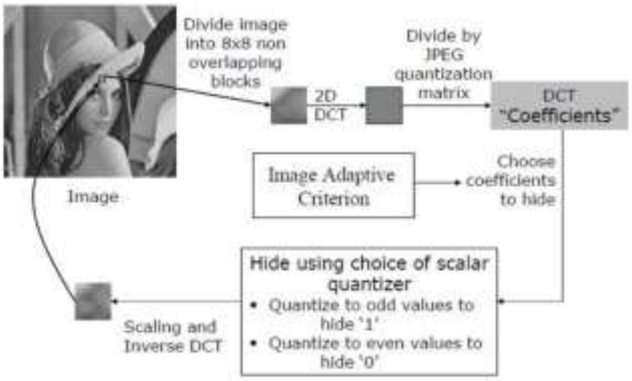
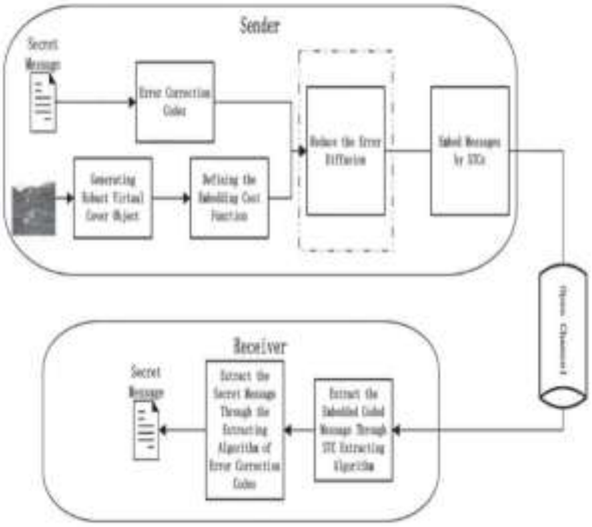

Divide image into 8x8 non overlapping blocks
2D DCT
Divide by JPEG quantization matrix
DCT "Coefficients"
Image
Image Adaptive Criterion
Choose coefficients to hide
Hide using choice of scalar quantizer
• Quantize to odd values to hide '1'
• Quantize to even values to hide '0'
Scaling and Inverse DCT

Image Adaptive Embedding

It is observed that the perceptual quality as well as the PSNR is better for the image with hidden data using local criteria. Note that though the PSNR is only marginally better, the actual perceptual quality is much better. This indicates that the local criteria must be used for robust and transparent high volume embedding.


Sender
Secret Message
Error Correction Codes
Generating Robust Virtual Cover Object
Defining the Embedding Cost Function
Reduce the Error Diffusion
Embed Message by STCs
Open Channel
Receiver
Secret Message
Extract the Secret Message Through the Extracting Algorithm of Error Correction Codes
Extract the Embedded Coded Message Through STC Extracting Algorithm

Frame Work

4.1 Design of Embedding Cost Function

After the robust virtual cover object is created, the popular minimal distortion model is used to concentrate the embedding modifications on the elements in the less distortion region to increase the performance on resisting detection. First, embedding cost function is defined to describe the distortion value caused by embedding message on cover object X

The cost function is designed based on the combination of the J-UNIWARD embedding cost function DF (D,D) of formula (1) and the robust virtual cover object. It is shown as follows:

$$\mathrm{DF}(x_j, y_j) = \begin{cases} 0, \quad x_j = y_j, \\ \rho_{ki}^{(1)}, \{x_j = 0 \& M_{ki} + \sigma_{ki} - D_k(i) = 1\} \\ \qquad \text{or}\{x_j = 1 \& D_k(i) - M_{ki} + \sigma_{ki} = 1\}, \\ \rho_{ki}^{(2)}, \{x_j = 0 \& M_{ki} + \sigma_{ki} - D_k(i) = 2\} \\ \qquad \text{or}\{x_j = 1 \& D_k(i) - M_{ki} + \sigma_{ki} = 2\}, \\ \text{wet\_cost}, \quad \text{else}, \end{cases}$$

where ρ(1) ki and ρ(2) ki , respectively, denote the cost values DF(Dk(i), ˜ Dk(i)) of modification magnitude 1 and 2 in theJ-UNIWARD embedding cost function, and wet cost denote the cost value that not advanced to be modified in the STCs embedding process(usually be set to a quite large value such as 108). Then, the robust virtual stegoobject Y is obtained after embedding secret message into X by STCs code, and the modification is mapped into image DCT coefficients Dk by formula (8) to obtain the DCT coefficients stego object.

4.2 Reduction of Error Diffusion

Based on the analysis result Section, the error probability can be reduced by shrinking the how of sub-matrix H. Because the height of H effects, the performance of steganographic embedding and computational complexity a lot, the value of h is often in the range h∈ 8,9,10,11,12 on embedding efficiency and complexity consideration.

Thus, we can increase the embedding rate of STCs by following steps:

I. Sort the elements of robust virtual cover object X computed from the same in-block position (i,j),i,j∈{1,2,...,8}of8×8DCTcoefficientsin arrowshot (X(i,j)),i,j∈1,2,...,8. II. Compute the required length nrd of virtual cover object based on the reduced embedding rate α. III. Select nrd elements from the set{sort(X(i,j)),i,j∈ {1,2,...,8}}in an order shared by the sender and receiver. The quantizing factor in JPEG is used to control the bit rate or the image quality. Here, the quantizing factor q1 is used to control the bit rate, and q2 is used to control the image quality. Besides, q1 is always smaller than q2, and the difference between them is used to control the hiding capacity. Fig. 7 shows the steps to take for embedding the secret information into a JPEG-compressed image.

4.3 Extracting Method

The extraction of secret information requires the original image, the stego-image, and the quantizing factors q1 and q2.

A. DCT Coefficient Identification The stego-image is a JPEG-compressed image. Thus, we can identify the selected DCT coefficients from the quantization table. The components whose value is 1 are the coefficients where we are to hide the secret information.

B. Generation of QET Apply loss JPEG compression to the original image with the quantizing factors q1 and q2. Then, dequantize the DCT blocks and calculate the difference between them to produce the QET.

C. Information Extraction To extract the secret information from the selected DCT coefficient (i, j), Eqn. (4) is used to obtain the embedded digit d. Then, we convert the digit d into a binary string whose length is E(i, j). Then we collect all the resulting bit strings

5. IMPLEMENTATION

There exists an implementation of this algorithm due to Kodovsky, but we propose a simulation of it which focuses only on PQ's embedding changes and leaves the coding strategy out for the reasons of computational efficiency, as it is essential to our experimental strategy.

Our algorithm, the simulated PQ, will be based on the idea introduced in Equation. Let f be the quality factor of the original JPEG images. Then we are looking for another quality factor f0 such that the set of coefficients that follow the relationship in equation (2.7) is maximized. For example, the JPEG images used for our experiments have quality factor f = 85. Following the argument from the previous section, we know that f0 ⇠ 70 would yield the maximum number of contributing coefficient modes. However in practice using the standard JPEG quantization table.

5.1 Perturbed Quantization

In nsF5 the minimum impact of a change is 1 as we decrease the absolute value of DCT coefficient by 1. PQ goes one step further with expected minimum impact of change being 0.5. To achieve this PQ requires side information and is therefore an instance of a content adaptive steganography algorithm. Its aim is to minimize the intrusiveness of embedding changes by perturbing the normal process of rounding which is a customary part of the JPEG's quantization step (step (4), Section 2.1). Two working modes can be differentiated. In the first mode PQ requires an original uncompressed (or lossless-format) image to calculate the side information. In the second, it uses an existing JPEG image for side-information and re-compresses it with a lower quality factor. Both variants produce a JPEG as an output stego image. Here we discuss the second but both modes follow a very similar algorithm.

For any two given quantization tables Q and Q0 one can find all DCT modes9 which satisfy the following equation:

$$kq_i = lq_i' + \frac{q_i'}{2}; \quad \text{s.t.} \quad k, l \in \mathbb{Z} \ \text{ and } \ \frac{q_i'}{g} \text{ is even}$$

where q0 I and qi are quantization steps at position i of tables Q and Q0 respectively and g is the greatest common divisor of q0 i and qi. For any image I with quantitation table Q and its re-compressed versionI0 with quantitation table Q0 all pairs of DCT coefficients modes qi,q 0 i satisfying the above equation are called contributing pairs.

During re-compression every odd coefficient which appears in a contributing mode will be rounded10 at quantitation with the rounding error of exactly 0.5. Different JPEG compression algorithms may perform rounding in different "directions". To embed into this position we simply choose the rounding direction such that the value of the least significant bit of the resulting (re-)quantized coefficient matches the required message bit. If the compression algorithm's rounding process was not deterministic, the expected impact of a change would be 0.5.

5.2 Embedding Rate

Unlike our implementation of nsF5, in this simulated PQ algorithm the true embedding rate will be different to our definition of embedding rate from Equation 2.4 as usable cover size here is almost always smaller than the cover size. sPQ's embedding rate is therefore measured in bits per usable DCT coefficient (bpuc).

```
procedure (simulated) PQ
    Set Q0 =2⇥Q
    I0 image I recompressed with Q0
    S indices of all odd conceits c of image I
    repeat
        Choose i from S at random without replacement
        if ci 2I and c0 i 2I0 satisfy equation (2.7) then
            Set c0 i = ci +0 .5 or c0 i = ci 0.5 . with probability of 0.5.
        end if
    until reached the length of payload
end procedure
```

5.3 Risk of JPEG compression library leak

It is known that a classifier's accuracy in an experimental environment can be artificially boosted by difference in the compression libraries that were used to create cover and stego images. In the case of sPQ this is not an issue because we are forced to re-compress to a lower quality factor to create both cover and stego images from a given recover for training and therefore the effects of the JPEG compression library will be the same for both classes. It is true, however, that if one was to inspect images created by such an algorithm in practice, the non-standard quantization table, which can be found from the JPEG headers or estimated, may itself serve as a leak and be treated as a warning sign that the image may have been tampered with using steganography embedding.

In the case of nsF5 no re-compression is performed because we are altering already quantized conceits and therefore no library leak will occur. This can be easily checked by performing a zero-operation embedding (i.e. just rewriting all DCTs with their original values) and comparing the output file to the original.

6. RESULT AND ANALYSIS

All steganography algorithms have to comply with a few basic requirements. The requirements are: Invisibility, Payload capacity, Robustness against statistical attacks, Robustness against image manipulation, Independent of file format and unsuspicious files. The following table compares least significant bit (LSB) insertion in BMP and in GIF files, JPEG compression steganography, and the patchwork approach and spread spectrum techniques, according to the above requirements:

	LSB in BMP	LSB in GIF	JPEG Compression	Patch work	Spread spectrum
Invisibility	High	Medium	High	High	High
Payload Capacity	High	Medium	Medium	Low	Medium
Robustness against statistical attacks	Medium	High	Medium	Low	Low
Robustness against image manipulation	Low	Low	Low	Medium	High
Independent of file Format	Low	Medium	High	Medium	High
Unsuspicious Files	Medium	High	Low	Medium	High

Table 1- Comparison of Steganography Algorithms

The levels at which the algorithms satisfy the requirements are defined as high, medium and low. A high level means that the algorithm completely satisfies the requirement, while a low level indicates that the algorithm has a weakness in this requirement. A medium level indicates that the requirement depends on outside influences, for example the cover image used. LSB in GIF images has the potential of hiding a large message, but only when the most suitable cover image has been chosen.

The process of embedding information during JPEG compression results in a stego image with a high level of invisibility, since the embedding takes place in the transform domain. JPEG is the most popular image file format on the Internet and the image sizes are small because of the compression, thus making it the least suspicious algorithm to use. However, the process of the compression is a very mathematical process, making it more difficult to implement. The JPEG file format can be used for most applications of steganography, but is especially suitable for images that have to be communicated over an open systems environment like the Internet.

7. FUTURE WORK

Digital Image Steganography system allows a user to securely transfer a text message by hiding it in a digital image file. 128 piece AES encryption is utilized to ensure the substance of the instant message regardless of whether its quality were to be distinguished. At present, no techniques are known for breaking this sort of encryption inside a sensible timeframe (i.e., two or three years). Furthermore, pressure is utilized to boost the space accessible in a picture.

To communicate something specific, a source text, a picture wherein the content ought to be installed, and a key are required. The key is utilized to help in encryption and to choose where the data ought to be covered up in the picture. A short book can be utilized as a key. To get a message, a source picture containing the data and the relating key are both required. The outcome will show up in the content tab in the wake of disentangling.

The common Internet-friendly format is offered. It is inherently more difficult to hide information in a JPEG image because that is exactly what the designers of JPEG wanted to avoid: the transmission of extra information.

8. CONCLUSIONS

A series of experiments are conducted and the results show that the proposed robust algorithm can achieve high extraction accuracy in various JPEG compression situations without knowing the quality factor of compression and can resist steganalysis statistical detection methods.

The meaning of Steganography is hiding information and the related technologies. There is a principal difference between Steganography and Encryption; however they can meet at some points too. They can be applied together, i.e. encrypted information can be hidden in addition. To hide something a covering medium is always needed. (Picture, sound track, text or even the structure of a file system, etc.) The covering medium must be redundant; otherwise the hidden information could be detected easily. We first obtain the stego-image by embedding data into the channel compressed original image using any of the existing JPEG stenographic schemes. According to the stego-image, we propose a coefficient adjustment scheme to slightly modify the original image to produce an intermediate image. On account of its easy to use interface, the application can likewise be utilized by any individual who needs to safely transmit private data. The main advantage of this program for individuals is that they do not have to have any knowledge about steganography or encryption. The visual method to encode the content, in addition to the visual key makes it simple for normal clients to explore inside the program.

REFERENCES

[1] N. Provos, “Defending Against Statistical Steganography,” Proc 10th USENEX Security Symposium 2005.

[2] N . Provos and P. Honeyman, “Hide and Seek: An introduction to Steganography,” IEEE Security & Privacy Journal 2003.

[3] Steven W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing.

[4] Katzenbeisser and Petitcolas , ”Information Hiding Techniques for Stenography and Digital watermarking” Artech House, Norwood, MA. 2000

[5] L. Reyzen And S. Russell, “More efficient provably secure Steganography” 2007.

[6] S.Lyu and H. Farid, “Steganography using higher order image statistics, “IEEE Trans. Inf. Forens. Secur. 2006.

[7] Venkatraman, s, Abraham, A. & Paprzycki M.” Significance of Steganography on Data Security “ , Proceedings of the International Conference on Information Technology : Coding and computing , 2004.

[8] Fridrich, J., Goljan M., and Hogea , D ; New Methodology for Breaking stenographic Techniques for JPEGs. “Electronic Imaging 2003”.

[9] L. Guo, J. Ni, and Y. Shi, “An efficient JPEG steganographic scheme using uniform embedding,” in Proceedings of the IEEE International Workshop on Information Forensics and Security, Costa Adeje, Tenerife,2012, pp.169–74.

[10] V. Holub, and J. Fridrich, “Low-complexity features for JPEG steganalysis using undecimated DCT,” IEEE Trans. Inf.ForensicsSec.,Vol.10,no.2,pp.219–28,Feb.2015.