JPEG Compression Steganography & Crypography Using Image-Adaptation Technique

Meenu Kumari
BVUCOE/IT Dept, Pune, India
Email: kumari.meenu90@gmail.com

Prof. A. Khare and Pallavi Khare
BVUCOE/IT Dept, Pune, India
SSSIST/E&TC Dept, Bhopal, India
Email: khareakhil@gmail.com

***Abstract*—In any communication, security is the most important issue in today's world. Lots of data security and data hiding algorithms have been developed in the last decade, which worked as motivation for our research. In this paper, named "JPEG Compression Steganography & Cryptography using Image-Adaptation Technique", we have designed a system that will allow an average user to securely transfer text messages by hiding them in a digital image file using the local characteristics within an image. This paper is a combination of steganography and encryption algorithms, which provides a strong backbone for its security. The proposed system not only hides large volume of data within an image, but also limits the perceivable distortion that might occur in an image while processing it. This software has an advantage over other information security software because the hidden text is in the form of images, which are not obvious text information carriers. The paper contains several challenges that make it interesting to develop. The central task is to research available steganography and encryption algorithms to pick the one the offer the best combination of strong encryption, usability and performance. The main advantage of this project is a simple, powerful and user-friendly GUI that plays a very large role in the success of the application.**

***Index Terms*—Steganography, Cryptography, Compression, JPEG, DCT, Local Criteria, Image-Adaptation, Huffman coding, ET, SEC scheme**

I. INTRODUCTION

In simple words, Steganography can be defined as the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information.

Though the concept of steganography and [1]cryptography are the same, but still steganography differs from cryptography. Cryptography [24] focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated. The strength of steganography can thus be amplified by combining it with cryptography.

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy far greater than necessary for the object's use and display. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding.

Given the proliferation of digital images, especially on the Internet, and given the large amount of redundant bits present in the digital representation of an image, images are the most popular cover objects for steganography. In the domain of digital images many different image file format exit, most of them for specific applications. For these different image file formats, different steganographic algorithms exist. Among all these file formats, the JPEG file format is the most popular image file format on the Internet, because of the small size of the images.

II. OVERVIEW

When working with larger images of greater bit depth, the images tend to become too large to transmit over a standard Internet connection. In order to display an image in a reasonable amount of time, techniques must be incorporated to reduce the image's file size. These techniques make use of mathematical formulas to analyze and condense image data, resulting in smaller file sizes. This process is called compression [3]. In images there are two types of compression: lossy and lossless compression [3]. Compression plays a very important role in choosing which steganographic algorithm to use. Lossy compression techniques result in smaller image file sizes, but it increases the possibility that the embedded message may be partly lost due to the fact that excess image data will be removed. Lossless compression

[1] Manuscript submitted on May 15, 2010; revised May 17, 2010; accepted May 31, 2010

doi:10.4304/jait.1.3.141-145

though, keeps the original digital image intact without the chance of lost, although is does not compress the image to such a small file size.

To compress an image into JPEG format, the RGB colour representation is first converted to a YUV representation. In this representation the Y component corresponds to the luminance (or brightness) and the U and V components stand for chrominance (or color). According to research the human eye is more sensitive to changes in the brightness (luminance) of a pixel than to changes in its color. This fact is exploited by the JPEG compression by down sampling the color data to reduce the size of the file. The color components (U and V) are halved in horizontal and vertical directions, thus decreasing the file size by a factor of 2.

The next step is the actual transformation of the image. For JPEG [18], the Discrete Cosine Transform (DCT) [18] is used, but similar transforms are for example the Discrete Fourier Transform (DFT). These mathematical transforms convert the pixels in such a way as to give the effect of "spreading" the location of the pixel values over part of the image. The DCT transforms [18] a signal from an image representation into a frequency representation, by grouping the pixels into 8 × 8 pixel blocks and transforming the pixel blocks into 64 DCT coefficients each. A modification of a single DCT coefficient will affect all 64 image pixels in that block.

The next step is the quantization [18] phase of the compression. Here another biological property of the human eye is exploited: The human eye is fairly good at spotting small differences in brightness over a relatively large area, but not so good as to distinguish between different strengths in high frequency brightness. This means that the strength of higher frequencies can be diminished, without changing the appearance of the image. JPEG does this by dividing all the values in a block by a quantization coefficient. The results are rounded to integer values and the coefficients are encoded using Huffman coding to further reduce the size.

Originally it was thought that steganography would not be possible to use with JPEG images, since they use lossy compression [3] which results in parts of the image data being altered. One of the major characteristics of steganography is the fact that information is hidden in the redundant bits of an object and since redundant bits are left out when using JPEG it was feared that the hidden message would be destroyed. Even if one could somehow keep the message intact it would be difficult to embed the message without the changes being noticeable because of the harsh compression applied. However, properties of the compression algorithm have been exploited in order to develop a steganographic algorithm for JPEGs.

One of these properties of JPEG is exploited to make the changes to the image invisible to the human eye. During the DCT transformation phase of the compression algorithm, rounding errors occur in the coefficient data that are not noticeable. Although this property is what classifies the algorithm as being lossy, this property can also be used to hide messages.

It is neither feasible nor possible to embed information in an image that uses lossy compression, since the compression would destroy all information in the process. Thus it is important to recognize that the JPEG compression algorithm is actually divided into lossy and lossless stages [3]. The DCT and the quantization phase form part of the lossy stage, while the Huffman encoding used to further compress the data is lossless. Steganography can take place between these two stages. Using the same principles of LSB insertion the message can be embedded into the least significant bits of the coefficients before applying the Huffman encoding. By embedding the information at this stage, in the transform domain, it is extremely difficult to detect, since it is not in the visual domain.

III. PROPOSED SYSTEM

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 [22].

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. The main ingredients of our embedding methodology are as follows.

(a) As is well accepted, 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)

(b) A novel feature of our method is that, from the candidate set of transform coefficients, the encoder employs local criteria to select which subset of coefficients it will actually embed data in. In example images, the use of local criteria for deciding where to embed is found to be crucial to maintaining image quality under high volume embedding.

(c) For each of the selected coefficients, the data to be embedded indexes the choice of a scalar quantizer for that coefficient. We motivate this by information theoretic analysis.

(d) 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 (the decoder guessing that a coefficient has embedded data, when it actually does not) and deletion errors (the decoder guessing that a coefficient does not have

embedded data, when it actually does). In principle, this can lead to desynchronization of the encoder and decoder.

(e) An elegant solution based on erasures and errors correcting codes is provided to the synchronization problem caused by the use of local criteria.

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.

Two methods for applying local criteria are considered. The first is the block-level Entropy Thresholding (ET) method, which decides whether or not to embed data in each block (typically 8X8) of transform coefficients, depending on the entropy, or energy, within that block. 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 [24] 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 are able to hide high volumes of data under both JPEG and AWGN attacks [24]. Moreover, the hidden data also survives wavelet compression, image resizing and image tampering attacks.

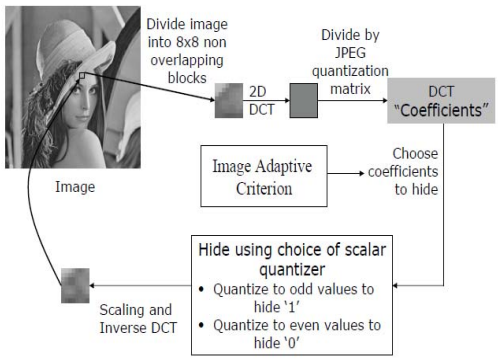

Figure 1. Image-adaptive embedding methodology

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.

Although we do not use specific perceptual models, we refer to our criteria as `perceptual' because our goal in using local adaptation is to limit perceivable distortion. Figure 1 shows a high-level block diagram of the hiding methods presented. Both the embedding methods, the entropy thresholding (ET) scheme, and the selectively embedding in coefficients (SEC) scheme, are based on joint photographic experts group (JPEG) compression standard. As seen in the Figure 1, the techniques involve taking 2D discrete cosine transform (DCT) of non-overlapping 8X8 blocks, followed by embedding in selected DCT coefficients.

Coding for Insertions & Deletions:

We noted that use of image-adaptive criteria is necessary when hiding large volumes of data into images. A threshold is used to determine whether to embed in a block (ET scheme) or in a coefficient (SEC scheme). More advanced image-adaptive schemes would exploit the human visual system (HVS) models to determine where to embed information. Distortion due to attack may cause an insertion (decoder guessing that there is hidden data where there is no data) or a deletion (decoder guessing that there is no data where there was data hidden). There could also be decoding error, where the decoder makes a mistake in correctly decoding the bit embedded. While the decoding errors can be countered using simple error correction codes, insertions and deletions can potentially cause catastrophic loss of synchronization between encoder and decoder.

In the ET scheme, insertions and deletions are observed when the attack quality factor is mismatched with the design quality factor for JPEG attack. However, for the SEC scheme, there are no insertions or deletions for most of the images for JPEG attacks with quantization interval smaller than or equal to the design interval. This is because no hidden coefficient with magnitude $\leq$ t can be ambiguously decoded to t+1 due to JPEG quantization with an interval smaller than the design one. Both the ET and SEC schemes have insertions/deletions under other attacks.

Coding Framework:

The coding framework employs the idea of erasures at the encoder. The bit stream to be hidden is coded, using a low rate code, assuming that all host coefficients that meet the global criteria will actually be employed for hiding. A code symbol is erased at the encoder if the local perceptual criterion for the block or coefficient is not met. Since we code over entire space of coefficients that lie in a designated low-frequency band, long codewords can be constructed to achieve very good correction ability. A maximum distance separable (MDS) code [24], such as Reed Solomon (RS) code, does not incur any penalty for erasures at the encoder. Turbo-like codes, which operate very close to capacity, incur only a minor overhead due to erasures at the encoder. Figure 3.4 shows how the sequence is decoded in the presence of attacks. As it is seen, insertions become errors, and deletions become additional erasures. It should be noted that a deletion, which causes an erasure, is about half as costly as an insertion, which causes an error. Hence, it is desirable that the data-hiding scheme [4] be adjusted in such a manner that there are only a few insertions. Thus, using a good erasures and errors correcting code, one can deal with insertions/deletions without a significant decline in original embedding rate. Reed-Solomon codes have been used for ET scheme and Repeat Accumulate codes have been used for the SEC scheme.

IV. RESULT ANALYSIS

All steganographic 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, the patchwork approach and spread spectrum techniques, according to the above requirements:

TABLE I.
COMPARISON OF IMAGE STEGANOGRAPHY ALGORITHMS

	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	Low	Low	Medium	High	High
Robustness against image manipulation	Low	Low	Medium	High	Medium
Independent of file format	Low	Low	Low	High	High
Unsuspicious files	Low	Low	High	High	High

* - Depends on cover image used

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 ideal, in other words a perfect; steganographic algorithm would have a high level in every requirement. Unfortunately in the algorithms that are evaluated here, there is not one algorithm that satisfies all of the requirements. Thus a trade-off will exist in most cases, depending on which requirements are more important for the specific application.

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.

V. CONCLUSION AND SCOPE FOR FUTURE WORK

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. The technology of hiding should match the nature of the medium. The hidden information should not be lost, if the carrying medium is edited, modified, formatted, re-sized, compressed or printed. That's a difficult task to realize.The application is primarily intended to be used to inconspicuously hide confidential and proprietary information by anyone seeking to hide information. This software has an advantage over other information security systems because the hidden text are in the form of image, which is not obvious text information carriers.

Because of its user-friendly interface, the application can also be used by anyone who wants to securely transmit private information. The main advantage of this program for individuals is that they do not have to have any knowledge about steganography or encryption. The visual way to encode the text, plus the visual key makes it easy for average users to navigate within the program.

Digital Image Steganography system allows an average user to securely transfer text messages by hiding them in a digital image file. A combination of Steganography and encryption algorithms provides a strong backbone for its security. Digital Image Steganography system features innovative techniques for hiding text in a digital image file or even using it as a key to the encryption**.**

Digital Image Steganography [2] system allows a user to securely transfer a text message by hiding it in a digital image file. 128 bit AES encryption is used to protect the content of the text message even if its presence were to be detected. Currently, no methods are known for breaking this kind of encryption within a reasonable period of time (i.e., a couple of years). Additionally, compression is used to maximize the space available in an image.

To send a message, a source text, an image in which the text should be embedded, and a key are needed. The key is used to aid in encryption and to decide where the information should be hidden in the image. A short text can be used as a key. To receive a message, a source image containing the information and the corresponding

key are both required. The result will appear in the text tab after decoding.

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 that doesn't affect the appearance of the image.

ACKNOWLEDGEMENT

The work on this paper was supported by the Bharati Vidyapeeth University & College of Engineering, Pune. The views and conclusions contained herein are those of the authors and the paper contains the original work of the authors. We took help from many books, papers and other materials.

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 watermaking" 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] http:/ aakash.ece.ucsb.edu./ data hiding / stegdemo.aspx.Ucsb data hiding online demonstration . Released on Mar .09,2005.

[10] Mitsugu Iwanmoto and Hirosuke Yamamoto, "The Optimal n-out-of-n Visual Secret Sharing Scheme for GrayScale Images", IEICE Trans. Fundamentals, vol.E85-A, No.10, October 2002, pp. 2238-2247.

[11] Doron Shaked, Nur Arad, Andrew Fitzhugh, Irwin Sobel, "Color Diffusion: Error Diffusion for Color Halftones", HP Laboratories Israel, May 1999.

[12] Z.Zhou, G.R.Arce, and G.Di Crescenzo, "Halftone Visual Cryptography", IEEE Tans. On Image Processing,vol.15, No.8, August 2006, pp. 2441-2453.

[13] M.Naor and A.Shamir, "Visual Cryptography", in Proceedings of Eurocrypt 1994, lecture notes in computer science, 1994, vol.950, pp. 1-12.

[14] Robert Ulichney, "The void-and-cluster method for dither array generation", IS&T/SPIE Symposium on Electronic Imaging and Science, San Jose, CA, 1993, vol.1913, pp.332-343.

[15] E.R.Verheul and H.C.A. Van Tilborg, "Constructions and properties of k out of n visual secret sharing scheme", Designs, Codes, and Cryptography, vol.1, no.2, 1997, pp.179-196.

[16] Daniel L.Lau, Robert Ulichney, Gonzalo R.Arce, "Fundamental Characteristics of Halftone Textures: Blue-Noise and Green-Noise", Image Systems Laboratory, HP Laboratories Cambridge, March 2003.

[17] C.Yang and C.Laih, "New colored visual secret sharing schemes", Designs, Codes and Cryptography, vol.20, 2000, pp.325-335.

[18] Jain, Anil K., "Fundamentals of Digital Image Processing", Prentice-Hall of India, 1989

[19] C.Chang, C.Tsai, and T.Chen, "A new scheme for sharing secret color images in computer network", in Proc. of International Conference on Parallel and Distributed Systems, 2000, pp. 21-27.

[20] R.L.Alder, B.P.Kitchens, M.Martens, "The mathematics of halftoning", IBM J. Res. & Dev. Vol.47 No.1, Jan. 2003, pp. 5-15.

[21] R.Lukac, K.N.Plantaniotis, B.Smolka, "A new approach to color image secret sharing", EUSIPCO 2004, pp.1493-1496.

[22] H.Ancin, Anoop K.Bhattacharjya, Joseph Shu, "Improving void-and-cluster for better halftone uniformity",International Conference on Digital Printing Technoogies.

[23] D. Hankerson, P. D. Johnson, and G. A. Harris, "Introduction to Information Theory and Data Compression".

[24] Ranjan Bose, "Information Theory Coding and Cryptography".

Meenu Kumari- Completed B.E. in Information Technology from Sanjivani Educational Society & College of Engineering, Kopargaon, Pune University in 2005. Persuing M.Tech. IT from Bharati Vidyapeeth University College of Engineering, Pune. Presented one national comference on Image Compression. Published research paper in one e-journal & one international journals. Submitted research paper in other national & international journals for publication.

Prof. A. Khare- Completed B.E. and M.E. from Bhopal. Currently working as Assistant Professor, in Bharati Vidyapeeth University College of Engineering, Information Technology Department, Pune. Presented many national & international conferences & journals.

Pallavi Khare- Research student of SSSIST, E&TC Department, Bhopal, India.