INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

FPGA IMPLEMENTATION OF STEGANOGRAPHY ALGORITHM USING SECRET SHARING APPROACH FOR DIGITAL IMAGES

[1]P.Ayyanar. ME (APPLIED ELECTRONICS), Kamaraj College of Engineering and Technology

[2]J.Augustin Jacob., ME., (Ph.d) Kamaraj College of Engineering and Technology

[3]R.Karthik Kumar., ME (APPLIED ELECTRONICS), Kamaraj College of Engineering and Technology

[1]ayya.kvp65@gmail.com, [3]mail2rkarthikkumar@gmail.com

ABSTRACT

Steganography has been considered as the solution for illicit interception and unauthorized copying of digital media. Steganography hides secret data in a host medium and conveys the hidden data. The intended recipient can then extract the secret data from the stego image. The hidden data should make a visually imperceptible change to the stego image when viewed by an unintended recipient. A remedy to achieve the imperceptibility of the stego image has been proposed, and is a concept called secret sharing. In a secret sharing approach, the sending party sends N stego images to one or multiple recipients and allows a receiving party to extract the secret data only when all N stego images are available. The (N,1) is a steganography secret sharing approach that utilizes N+1 cover images. Both the sending party as well as the receiving party shares the N covers images through a secure channel. The remaining one cover image is converted to gray coded stego image at the sending party and is to be sent through an unsecure channel. The conversion is performed based on operations through the N+1 cover image pixels and secret data. In the Phase-I, stego object is generated by embedding the text message into cover image using spatial domain technique and is implemented in FPGA.

Keywords: Steganography; FPGA;Stego; imperceptible

1. INTRODUCTION

The security of steganographic communication between two principles lies in the inability of an eavesdropper to distinguish cover objects from stego images. A system can be said to be insecure if an eavesdropper can suspect the presence of the secret communication. The system must first be a ‘perfectly secure’ steganographic systems, where even a computationally unbounded observer cannot detect the presence of a secret message exchange. The second thing is that, it might be difficult to construct secure schemes usable in practice. Thirdly, they all require the knowledge of the probability distribution of normal covers; although it might be possible in certain cases to compute this probability, it will in general be infeasible to obtain.

2 .STEGANOGRAPHY

The word 'Steganography' is of Greek origin and means 'covered, or hidden writing'. The art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. A steganographic message (the plaintext) is often first encrypted by some traditional means, and then a cover-text is modified in some way to contain the encrypted message (cipher-text), resulting in stego text. For example, the letter size, spacing, typeface, or other characteristics of a cover-text can be manipulated to carry the hidden message; only the recipient (who must know the technique used) can recover the message and then decrypt it.

2.1 Digital Rights and Copyright Marking

One of the driving forces behind the increased use of copyright marking is the growth of the Internet which has allowed images, audio, video, etc to become available in digital form. Though this provides an additional way to distribute material to consumers it has also made it far easier for copies of copyrighted material to be made and distributed. In the past, pirating music, for example, used to require some form of physical exchange. Copyright marking is seen as a partial solution to these problems. The mark can be embedded in any legal versions and will therefore be present in any copies made. This helps the copyright owner to identify who has an illegal copy.

2.2 Embedding and Detecting

The first step in embedding and hiding information is to pass both the secret message and the cover message into the encoder. Inside the encoder, one or several protocols will be implemented to embed the secret information into the cover message. The type of protocol will depend on what information you are trying to embed and what you are embedding it in. For example, you will use an image protocol to embed information inside images.

A key is often needed in the embedding process. This can be in the form of a public or private key so you can encode the secret message with your private key and the recipient can decode it using your public key. In embedding the information this way, you can reduce the chance of a third party attacker getting hold of the stego object and decoding it to find out the secret information. In general the embedding process inserts a mark, M, in an object.

In the decoding process, the stego object is fed in to the system. The public or private key that can decode the original key that is used inside the encoding process is also needed so that the secret information can be decoded. Depending on the encoding technique, sometimes the original cover object is also needed in the decoding process. Otherwise, there may be no way of extracting the secret information from the stego object. After the decoding process is completed, the secret information embedded in the stego object can then be extracted and viewed. The generic decoding process again requires a key.

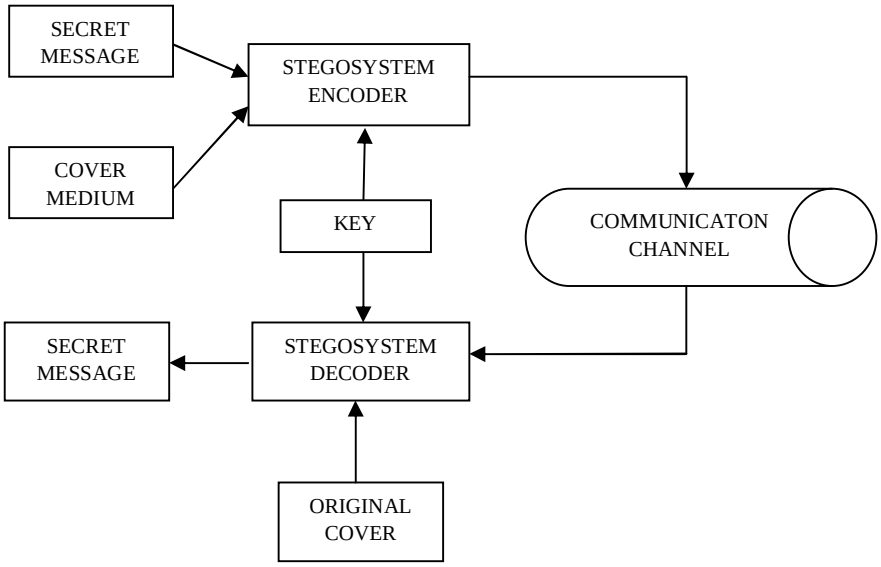

Figure 1. Steganography system

2.3 Requirements of Hiding Information Digitally

There are many different protocols and embedding techniques that enable us to hide data in a given object. However, all of the protocols and techniques must satisfy a number of requirements so that steganography can be applied correctly. The following is a list of main requirements that steganography techniques must satisfy:

The integrity of the hidden information after it has been embedded inside the stego object must be correct. The secret message must not change in any way, such as additional information being added, loss of information or changes to the secret information after it has been hidden. If secret information is changed during steganography, it would defeat the whole point of the process.

The stego object must remain unchanged or almost unchanged to the naked eye. If the stego object changes significantly and can be noticed, a third party may see that information is being hidden and therefore could attempt to extract or to destroy it.

2.4 Steganography Terms

Carrier File – A file in which a secret is to be hidden.

Steganalysis – The process of detecting hidden information inside a file.

Stego-image – The image in which the information is hidden.

Redundant Bits – Pieces of information inside a file which can be overwritten or altered without damaging the file.

Payload – The information which is to be concealed.

2.5 Classification of Steganography

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. Figure 1.2 shows the classification of steganography.

3. Image Format

It appends the secret message found in the text file 'Message.txt' into the JPEG image file 'Cover.jpg' and produces the stego-image 'Stego.jpg'. The idea behind this is to abuse the recognition of EOF (End of file). In other words, the message is packed and inserted after the EOF tag. When Stego.jpg is viewed using any photo editing application, the latter will just display the picture ignoring anything coming after the EOF tag. However, when opened in Notepad for example, our message reveals itself after displaying some data as shown in Figure 1.3. The embedded message does not impair the image quality. Neither image histograms nor visual perception can detect any difference between the two images due to the secret message being hidden after the EOF tag.

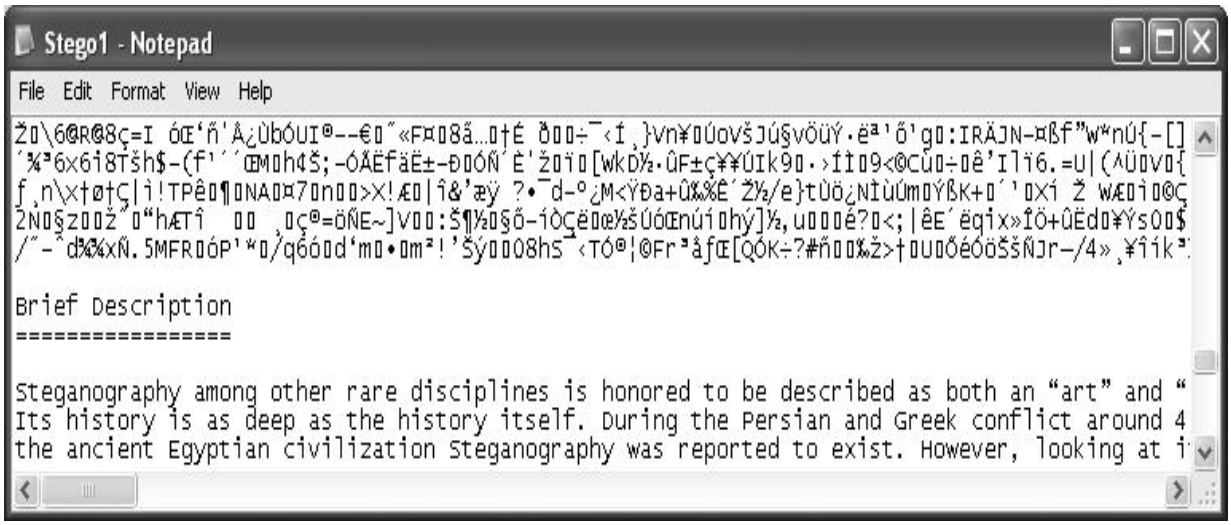

Figure 2. Stego image in Notepad

Unfortunately, this simple technique would not resist any kind of editing to the Stego-image nor any attacks by steganalysis experts. Note that the format of the inserted message remains intact

4. Adaptive Steganography

Adaptive Steganography known as 'Statistics-aware embedding', 'Masking' or 'Model-Based'. This method takes statistical global features of the image before attempting to interact with its LSB/DCT coefficients. The statistics will dictate where to make the changes. It is characterized by a random adaptive selection of pixels depending on the cover image and the selection of pixels in a block with large local STD (Standard Deviation). The latter is meant to

avoid areas of uniform colour (smooth areas). This behavior makes adaptive steganography seek images with existing or deliberately added noise and images that demonstrate colour complexity.

The model-based method (MB1), generates a stego-image based on a given distribution model, using a generalized Cauchy distribution, that results in the minimum distortion. Due to the lack of a perfect model, this steganographic algorithm can be broken using the first-order statistics. Moreover, it can also be detected by the difference of 'blockiness' between a stego-image and its estimated image reliably. The discovery of 'blockiness' produce an enhanced version called MB2, a model based with de-blocking.

'*A Block Complexity based Data Embedding*' (ABCDE). Embedding is performed by replacing selected suitable pixel data of noisy blocks in an image with another noisy block obtained by converting data to be embedded. This suitability is identified by two complexity measures to properly discriminate complex blocks from simple ones; which are run-length irregularity and border noisiness. The ABCDE method introduced a large embedding capacity; however, certain control parameters had to be configured manually, e.g., finding an appropriate section length for sectioning a stream of resource blocks and finding the threshold value that controls identification of complex blocks. These requirements render the method unsuitable for automatic processes.

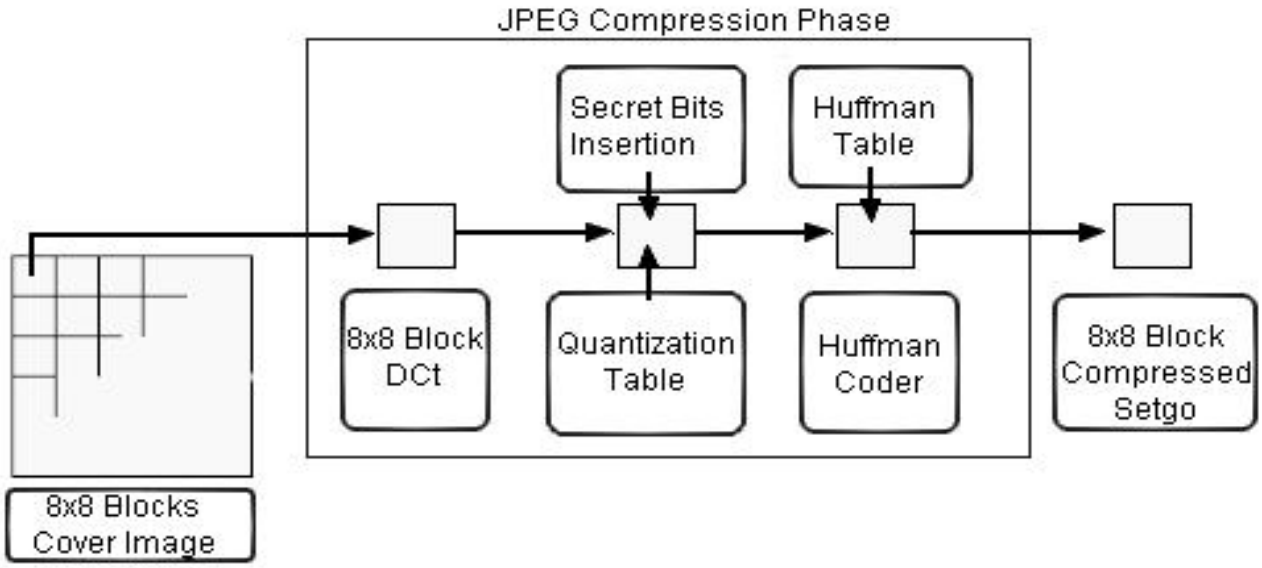

Figure 3. Data flow diagram of embedding in the frequency domain

The next step is the quantization 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

5. RESULTS AND DISCUSSION

Figure 5.1 Original Image

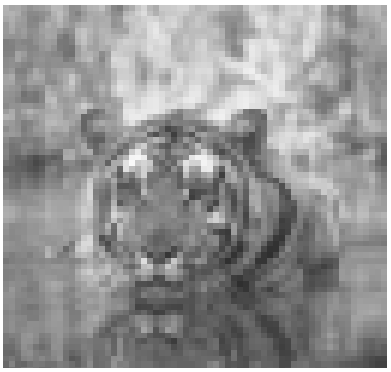

Figure 5.2 Original gray scale image

Figure 5.3 Stego Image

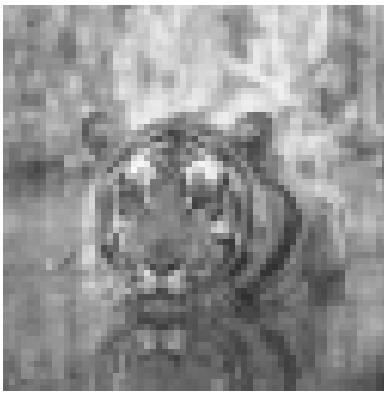

Figure 5.4 Stego gray scale image

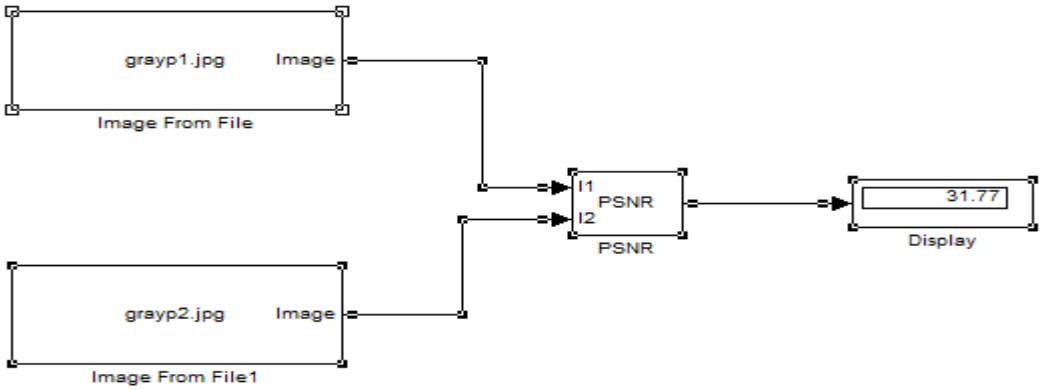

Figure 5.5 Simulink Block Diagram to Determine PSNR Value

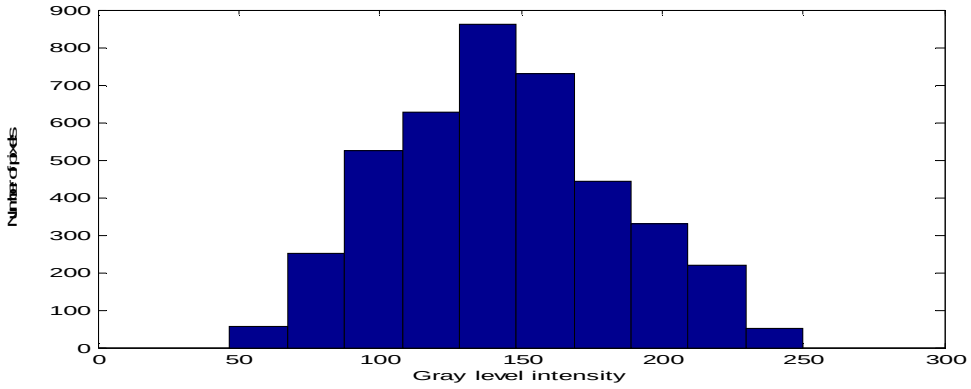

Figure 5.6 Histogram of Original Image

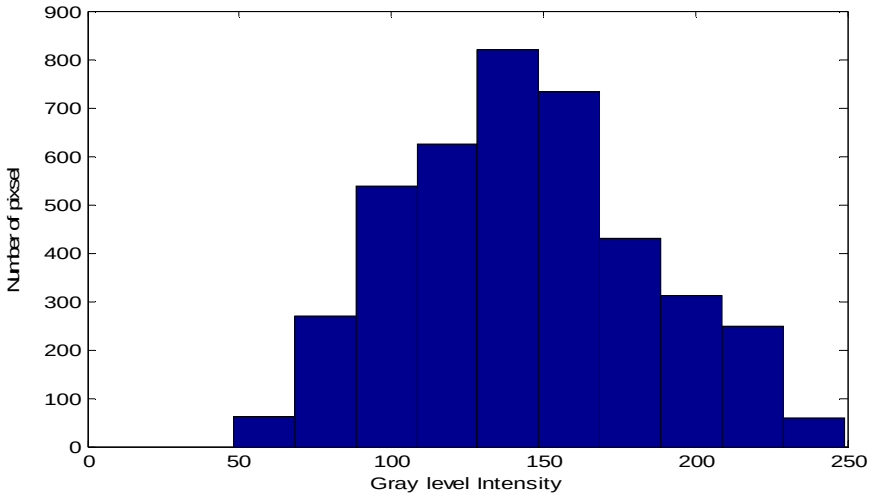

Figure 5.7 Histogram of Stego Image

REFERENCES

1. Chang C. C., and R. J. Hwang (1998), 'Sharing Secret Images using Shadow Codebooks', Information Sciences, 111(1-4): pp 335-345.
2. Chang C.-C., T. D. Kieu, and Y.-C. Chou (2008), 'A High Payload Steganographic Scheme Based on (7, 4) Hamming Code for Digital Images', Proc. of the 2008 International Symposium on Electronic Commerce and Security, pp.16-21.
3. Feng J. B., H. C. Wu, C. S. Tsai, and Y. P. Chu (2005), 'A New MultiSecret Images Sharing Scheme Using Largrange's Interpolation', Journal of Systems and Software, 76(3), pp.327- 339.
4. Kim C., E.-J. Yoon, Y.-S. Hong, and H. 1. Kim (2009), 'Secret Sharing Scheme Using Gray Code based on Steganography' Journal of the Institute of Electronics Engineers of Korea, 46(1), pp.96-102.
5. Lin C. C., and W. H. Tsai (2004), 'Secret Image Sharing with Steganography and Authentication', Journal of Systems and Software, 73(3), pp.405-414.

6. Miche Y., P. Bas, A. Lendasse, and C. Jutten (2009), 'Reliable Steganalysis Using a Minimum Set of Samples and Features', EURASIP Journal on Information Security, DOI:IO.1155/2009/901381.
7. Provos N., and P. Honeyman (2003), 'Hide and Seek: An Introduction to Steganography', IEEE Security and Privacy, 1(3), pp. 32-44.
8. Sallee P. (2004), 'Model-Based Steganography' LNCS, Vol. 2939, pp. 254-260.
9. Sharmir A. (1979), 'How to Share a Secret', Communications of the ACM, 22(11), pp.612-613.
10. Solanki K., A. Sarkar, and B. S. manjunath (2008), 'YASS: Yet Another Steganographic Scheme That Resists Blind Steganalysis', LNCS, Vol. 4567, pp. 16-31.
11. Thien C. C., and J. C. Lin (2002), 'Secret Image Sharing', Computers and Graphics, 26(1), pp.765-770.
12. Westfeld A. (2001), 'F5-A Steganographic Algorithm: High Capacity Despite Better Steganalysis' LNCS, Vol. 2137, pp. 289-302.
13. Westfeld A., and A. Pfitzmann (1999), 'Attacks on Steganographic Systems', LNCS, Vol. 1768, pp. 61-76.
14. Zhang W., S. Wang, and X. Zhang (2007), 'Improving Embedding Efficiency of Covering Codes for Applications in Steganogrphy', IEEE Communications Letters, 11(8), pp.680-682.