International Association of Scientific Innovation and Research (IASIR)
(An Association Unifying the Sciences, Engineering, and Applied Research)

ISSN (Print): 2279-0047
ISSN (Online): 2279-0055

International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

Robust Image Encryption with SPIHT Compression

Ms. SITA YADAV
Computer Department,
D.Y.Patil Institute of Engineering and Technology,
University of Pune,
Pimpri, Pune.
INDIA

ABSTRACT: *Lossless Image information concealing is receiving interest from last few years. Because the hidden information cannot be extracted properly by exploitation previous ways owing to their fragile formula. This paper proposes a sturdy theme to demonstrate lossless image information concealing with compression. We have used the SPIHT compression formula with astrassen methodology. This paper is successful in lossless image information concealing and Compression, also the protection level exaggerated in hides of information and additionally in discover of cipher text. This paper offers the comparative study and implementation of lossless image information concealing with image compression and Security.*

KEYWORDS: *Lossless image information concealing, SPIHT, astrassen matrix method, secure information concealing*

I. Introduction

Data concealing could be a method to cover information into cover media such as pictures, audio clips or video streams. We mainly target information concealing into pictures. In this typical example of a stenographic application for covert communication, the receiver has no interest in the original cowl image before the message was embedded. We shall use digital pictures because the cowl objects in this paper within which we tend to engraft the hidden info. The challenge of exploitation steganography in cowl pictures is to hide as information with the sturdy amount least noticeable difference within the stegoimage. Steganography algorithms operate on primarily 3 sorts of pictures: Raw images (i.e., bmp for- mat), Palette based mostly pictures (i.e., GIF images) and JPEG pictures. Sometimes it's found that associate formula used to hide massive amounts of knowledge generally result in lower physical property (i.e., larger modification to the image appearance) and a additional sturdy formula result into fix embedding capability. The JPEG image generation first rotten the input image into variety of eight 8x 8 blocks. Then DWT of every block are computed and the resultant DWT constant matrix is quantal exploitation a standard division table. Finally the inverse DWT of quantal constant matrix are evaluated and also the final JPEG image is obtained when rounding error the values. We propose a changed SPIHT comleft handal to writing formula, which can cut back bits redundancy and scanning redundancy of traditional SPIHT. Once scanning the gathering of list of insignificant sets (LIS), if there are vital coefficients in their offspring, ancient SPIHT can generate three direct kid nodes of their offspring and verify the importance of kid nodes to make your mind up wherever to send these kid nodes, list of great pixels or list of insignificant pixels, and meantime it generates a L-type set to LIS. We choose pixels with the HIOP (Higher Intensity of Pixel) formula. As we tend to divided the image into blocks and verify higher Intensity color of constituent in every block and pixels. As we tend to divided the image into blocks and verify higher intensity color of constituent in every block and use astrassen multiplication in every block. We create additional dispersion in elite pixels. As a result, the security level exaggerated in hide of information and additionally in discover of cipher text. It is also, strive to not degrade image quality and as so much as attainable doesn't modification the image size. SPIHT deserves special attention as a result of it provides the following:

1. Smart image quality, high PSNR.
2. it's optimized for progressive image transmission.
3. Produces a completely embedded coded file.
4. Straightforward division formula.
5. Quick coding/decoding nearly even.
6. Has wide applications, utterly accommodative.
7. Is used for lossless compression.

8. Will code to actual bit rate or distortion.
9. Economical combination with error protection.

II. System Design for Proposed Method

Select one applied math amount as parameter. Contemplate associate image divide it into non overlapping blocks, allow us to contemplate the block size of 10*10 pixels. Block should be even in numbers. Create 2 sets say P and Q. Set 'P' can consist set element 'Pi' and 'Q' can have part 'Qi'. That is every set has thirty two pixels. For every block, calculate the difference worth 'a'. The distinction worth 'a' is outlined as the arithmetic average of distinction of gray scale values of constituent pairs inside the block. A combine is chosen because the horizontally neighboring pixels. The distinction worth 'a' is given as

$$N=50 \qquad (1)$$

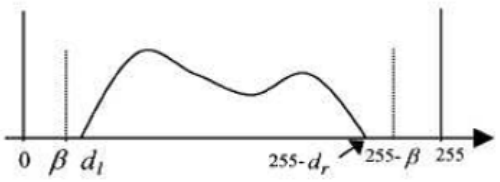
$$A= 1/n\sum_{i=0}^{n} Pi-Qi \qquad (2)$$

Where n is that the variety of constituent combine within the block. The difference worth 'a' is anticipated to be terribly near zero. Since the distinction worth relies on statistics of all pixel within the block, although the constituent within the block has small change when JPEG compression, this statistic value isn't simple to vary. Hence, this worth in and of itself robust against JPEG compression and alternative incidental alteration. Therefore, the distinction value is chosen because the sturdy amount for information embedding Note that the block size isn't necessary to be eight 10*10. It is the other even variety. However odd block size is not allowed as a result of the pattern doesn't perpetually contain couple of pixels. Since every block is employed to engraft one bit, the block size can therefore have an effect on information embedding capacity. Hence, larger the block size, the lower the info embedding capability. The hardiness of the embedded bits, on the opposite hand, is going to be stronger if the block size is larger. So the compromise between the info embedding capability and hardiness of hidden information want to be created in step with the precise application.

A. Category 1

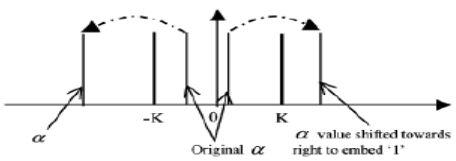
In this class the constituent grayscale values of the block are in the central a part of the bar graph as shown in Fig. 1. In this class, following 2 cases are thought of according to the distinction worth 'a'.

Figure 1 Block Bar Graph for Category - 1



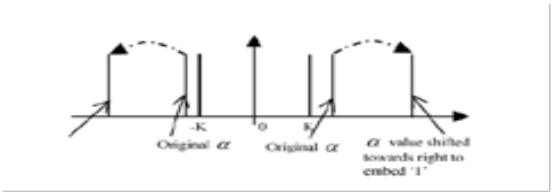
Case 1: The distinction worth 'a' is found between the threshold K and -K
1. If the embedding bit is "1" then the distinction worth 'a' is shifted by amount 'a' towards the proper hand aspect or left hand aspect reckoning on if 'a' is positive or negative.
2. If the embedding bit is"0" then the distinction value 'a' is unbroken intact there in block.

Figure 2 Embedding a trifle "1" for category-1 (Case – 1)



Case 2: The distinction worth 'a' is found on the far side the threshold K and -K.
1. If the embedding bit is "1" then the distinction worth 'a' is shifted by amount 'a' towards the proper hand aspect or left hand aspect reckoning on if 'a' is positive or negative.
2. If the embedding bit is "0" then the distinction worth 'a' is unbroken intact therein block.

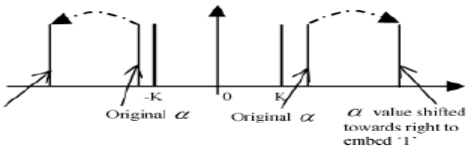
Figure 3 Embedding a trifle "1" for category-2 (case-1)



B. Category 2

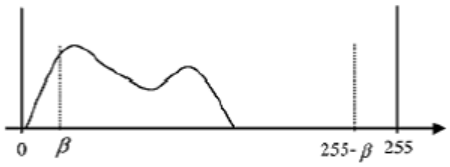
In this class the constituent grayscale values of the block are in the left aspect of the bar graph. During this class, following three cases are thought of in step with the distinction value 'a'.

Figure 4 Block bar graph for Category-2



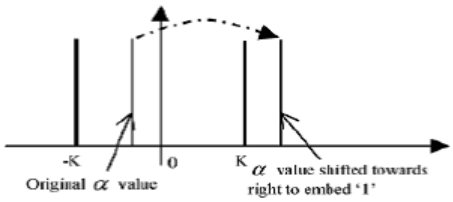
Case 1: The distinction worth 'a' is found between the threshold K and -K.
1. If the embedding bit is"1" then the distinction worth 'a' is shifted by amount 'a' towards the right hand aspect.
2. If the embedding bit is" 0" then the distinction worth 'a' is unbroken intact therein block.

Figure 5 Embedding a trifle" 1" for category-2 (case-1)



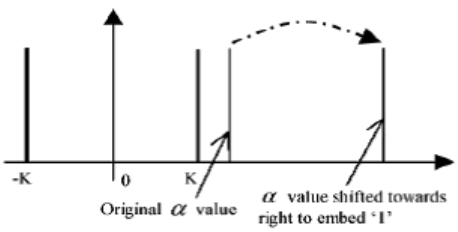
*Case 2:*The distinction worth 'a' is found on the far side the threshold K. in spite of whether or not the embedding bit is "0" or "1", it perpetually engraft bit "1" by shifting the distinction value 'a' towards hand aspect. Hence, it's going to introduce an error bit, that is then corrected exploitation error correction code.

Figure 6 Embedding a trifle "1" for category-2(case – 2)



Case 3: The distinction worth 'a' is found on the far side the threshold -K during this case, constituent grayscale values for the block never changes. Block perpetually engraft bit"0" regardless the embedding bit is" 0" or "1". Hence, it's going to introduce an error bit, that is then corrected exploitation error correction code.

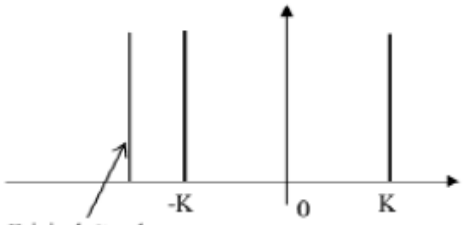
Figure 7 Embedding a trifle "0" for category-2 (case-3)



C. Category 3

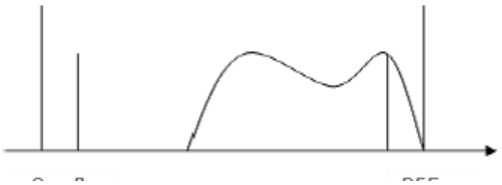
In this class the constituent grayscale values of the block are within the right aspect of the bar graph. Class three is comparable to class two except that the distribution of gray scale values of the block is near the bound. Hence, the data embedding formula is comparable to it of category 2 except shifting distinction worth 'a' to the left hand side. During this class, following 3 cases are thought of according to the distinction worth 'a'.

Figure 8 Block bar graph for class - three



Case 1: The distinction worth 'a' is found between the threshold K and -K one. If the embedding bit is"1" then the distinction worth 'a' is shifted by amount 'a' towards the left hand side of aspect. If the embedding bit is"0" then the distinction worth 'a' is unbroken intact in that block.

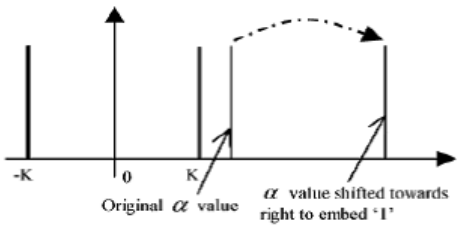
Figure 9 Embedding a trifle" 1" for category-3 (case-1)



Case 2: The distinction worth 'a' is found on the far side the threshold K. in spite of whether or not the embedding bit is "0" or "1",it perpetually engraft bit "1" by shifting the distinction value 'a' towards left hand aspect. Hence, it's going to introduce an error bit, that is then corrected exploitation error correction code.

Case 3: The distinction worth 'a' is found on the far side the threshold -K during this case, constituent grayscale values for the block never changes. Block perpetually engraft bit"0" regardless the embedding bit is"0" or"1". Hence, it's going to introduce an error bit, that is then corrected exploitation error correction code.

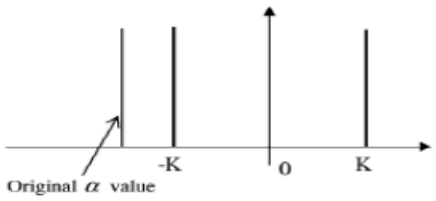
Figure 10 embedding a trifle"0" for category-2 (case-3)



D. Category 4

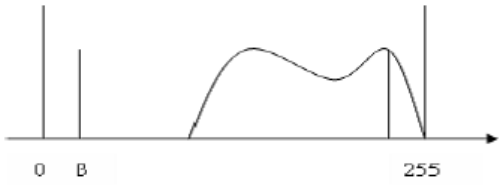
In this class the constituent grayscale values of the block are within the right aspect of the bar graph. Class three is comparable to class two except that the distribution of gray scale values of the block is near the bound. Hence, the data embedding formula is comparable to it of category 2 except shifting distinction worth 'a' to the left hand side. During this class, following 3 cases are thought of according to the distinction worth 'a'.

Figure 11 Block bar graph for class - three



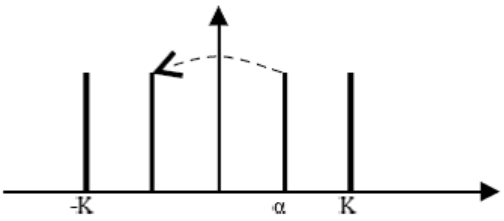
Case 1: The distinction worth 'a' is found between the threshold K and -K one. If the embedding bit is"1" then the distinction worth 'a' is shifted by amount 'a' towards the left hand side of aspect. If the embedding bit is"0" then the distinction worth 'a' is unbroken intact in that block.

Figure 12 Embedding a trifle "1" for category-3 (case-1)



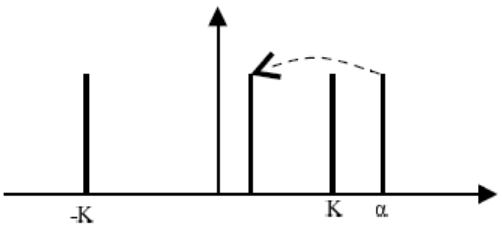
Case 2: The distinction worth 'a' is found on the far side the threshold K. in spite of whether or not the embedding bit is "0" or "1",it perpetually engraft bit "1" by shifting the distinction value 'a' towards left hand aspect. Hence, it's going to introduce an error bit, that is then corrected exploitation error correction code.

Figure 13 Embedding a trifle"1" for category-3 (case-2)



Case 3: The distinction worth 'a' is found on the far side the threshold -K during this case, constituent grayscale values for the block never changes. Block perpetually embeds bit"0" regardless the embedding bit is"0" or"1". Hence, it may introduce a slip bit that is then corrected exploitation ECC.

Figure 14 Embedding a trifle "0" for category-3 (case-3)



E. Information Embedding Pseudo Code

*L*et us contemplate the most important block size is twenty. The edge K is from two to five. The embedding level is double the K, which suggests is from four to ten. Error correction code is selected among 5 sorts of BCH codes delineated higher , then the pseudo code of the total formula is as following.

for BlockSize = SmallBlockSize to LargeBlockSize
for Embedding level B = Low to High
According the payload, select the suitable BCH code.
If no BCH is found, back to step 2.
Permutation
Set threshold K = half of Embedding level B
End
End

III. Mathematical Model

One of the most options of the planned method to writing methodology is that the ordering information isn't expressly trans left handed. Instead, it's supported the very fact that the execution path of any formula is outlined by the results of the comparisons on its branching points. So, if the encoder and decoder have constant algorithmic rule, then the decoder can duplicate the encoder's execution path if it receives the results of the magnitude comparisons, and the ordering info is recovered from the execution path. One vital reality utilized in the look of the algorithmic rule is that we tend to don't want type all coefficients. Actually, we want associate formula that merely selects the coefficients such

$$2^n \leq |c_i j| \leq 2^{n+1} \qquad (3)$$

with n decremented in each pass. Given n, if

$$|ci\ j| \leq 2 \qquad (4)$$

then we are saying that a constant is significant; otherwise it is called insignificant. The algorithmic rule divides the set of pixels into partitioning subsets 's' and performs the magnitude check. If the decoder receives a "no" to it answer(the set is insignificant), then it is aware of that all coefficients in 'T' are insignificant. If the solution is "yes" (the set is significant), then an exact rule shared by the encoder and also the decoder is employed to partition 'T' into new subsets 'T', and also the significance check is then applied to the new subsets. This set division method continues till the magnitude check is completed to any or all single coordinate important sub- sets so as to spot every significant constant. To cut back the quantity of magnitude comparisons set partitioning rule that uses associate expected ordering within the hierarchy outlined by the sub band pyramid. The target is to make new partitions such that subsets expected to be insignificant contain a large number of parts, and sub- sets expected to be important contain only 1 part. to form clear the link between magnitude comparisons and message bits, we tend to use the operate to point the importance of a set of coordinates T. Normally, most of associate image's energy is focused within the low frequency elements. Consequently, the variance decreases as we tend to move from the highest to rock bottom levels of the sub band pyramid. Furthermore, it's been discovered that there's a special self-similarity between sub bands, and also the coefficients are expected to be higher magnitude-ordered if we tend to move downward within the pyramid following constant spacial orientation. A tree structure of spacial orientation tree (SOT), defines the spacial relationship on the hierarchical pyramid.

A. Dynamic Programming and Serialization

The program logic follows the set theory method and divide and conquer method.

1) Consider the image 'X' divide image in block size of 10*10(non overlapping blocks).

2) Split it into cluster of 'pi' and 'qi'.

3) Calculate 'a' by using Equation 1.

4) Choose horizontal pixel pair 'Pi' and 'Qi'.

5) The proposed method used BCH (63, 7, 15) code.

The embedding capacity is computed as given below.

$$\sum pi = Pi \qquad (5)$$

$$\sum qi = Qi \tag{6}$$

$$TotalNo.ofBlocks = \frac{\mathrm{Im}\,ageSize}{BlockSize} \tag{7}$$

$$R = 63 \bmod (Totalno.ofBlocks) \tag{8}$$

$$\text{Embedding Capacity} = \frac{(TotalNo.ofBlock - R)}{63} \tag{9}$$

6) Peak-signal-to-noise (PSNR) is used to evaluate the visual quality of an embedded image. PSNR is defined by the following equation.

$$PSNR = 10 * \log_{10}\left(\frac{255^2}{MSE}\right)(db) \tag{10}$$

To calculate MSE, following equation is required,

$$MSE = \frac{1}{H * W}\sum_{i=1}^{H}\sum_{j=1}^{W}(I_{(i,j)} - I'_{(i,j)})\text{^}2 \tag{11}$$

Hardiness (bpp) means that the living bit rate within the unit bpp (bit per pixel), i.e, once a compressed image has a rate higher than or adequate to this bit rate the hidden data will be retrieved while not error.

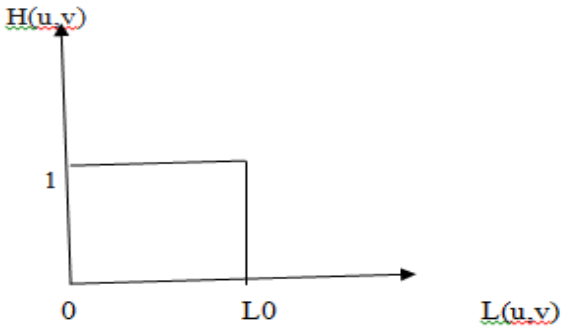
$$ROBUSTNESS = \frac{Compressed\ Image\ Size}{Original\ Image} \tag{12}$$

To use image compression by SPIHT use DWT mechanism. 1st convert the image into its riffle transform then transmits data concerning the wavelet constant. Decoder uses the received signal to reconstruct the riffle Associate an inverse rework to recover the image. Apply low pass filter and high pass filter in one dimension 0.5 the frequency range between filter. Analyze the try, low pass filter can get low frequency element and high pass filter can get high frequency element out of 4 bands LL, LH, HL, HH.

$$f(n) = \begin{cases} 0 & \text{if L(u,v) is less then L} \\ 1 & \text{if L(u,v) is grater then L} \end{cases}$$

L = Specific positive quantity
L(u,v) = Distance from point(u,v) to the origin of frequency plane

Figure 5. Low pass filter graph



For high pass filter equation 13 and equation 14 can be used as,

L_0 = Cut Of Frequency

$$L(u,v) = (Distance\ From\ The\ Origin\ [u^2 + v^2])^{\frac{1}{2}} \quad (13)$$

$$H(u,v) = \frac{1}{1 + [\frac{L_0}{L(u,v)}]^{2n}} \quad (14)$$

$$S_n(\Gamma) = \begin{cases} 1, & \max_{(i,j)\in\Gamma} \{|c_{ij}|\} \geq 2^n \\ 0, & \text{otherwise} \end{cases}$$

The above equations give the mathematical formulas involve in this paper work.

IV. MESSAGE EXTRACTION

In this section we will discuss the retrieving the message from the image freelance of the file format. Once a message has been retrieved it is to be regenerate in to the original message. This method will be done by reading the embedded knowledge from the file. The scan knowledge can be in bytes format. This may be done by extract the pixels of output image in one array. SPIHT codes the individual bits of the image riffle rework coefficients following a bit-plane sequence. Thus, it is capable of ill the image absolutely (every single bit of it) by writing all bits of the rework. It is quicker than the quality matrix operation algorithmic program and is useful in observe for giant matrices. Sensible implementations of Strassens algorithmic program switch to plain methods of matrix operation for little enough sub matrices, for which those algorithms are additional economical. The particular crossover purpose that Strassens algorithmic program is additional economical depends on the particular implementation and hardware. Earlier authors had calculable that Strassens algorithmic program is quicker for matrices with widths from thirty two to 128 for optimized implementations. It is attainable to urge all needed data by one break down of image. Thus the list of insignificant pixel, list of insignificant sets and list of serious pixels are created and then DWT and SPIHT can apply on that. We are ready to hide the info in image with strong amount and comparison between basic matrix operation and strassen matrix operation is given below. Strassen matrix multiplication will be done by divide and conquer method. The time analysis followed by it is

$$i^{\log n} \quad (15)$$

V. Conclusion

Here, the proposes the sturdy and lossless image data concealing theme with comparative study. That employs a sturdy statistical amount to mitigate the result of compression and small incidental alteration for information embedding. It utilizes totally different bit-embedding ways for groups of constituents with totally different pixel grayscale worth distributions. It employs error correction codes along with permutation theme. Consequently, it's with success avoided exploitation modulo-256 addition to realize losslessness, thus eliminating the annoying salt-and-pepper noise. Additionally this paper covers the compression with enhanced security in image information coding.

VI. References

[1] Z. Fang and Naixure Xiong,Interpolation-Based Direction-Adaptive Lifting DWT and Modified SPIHT for Image Compression in Multimedia Communications,IEEE Systems Journal, Vol.5, Dec 2011.

[2] Changhe Song, Yunsong Li, and Bormin Huang ,A GPU-Accelerated Wavelet Decompression System With SPIHT and Reed-Solomon Decoding for Satellite Images,IEEE Journal of selected topics in applied earth observations and remote sensing, VOL. 4, NO. 3, SEPTEMBER 2011 683

[3] T. Morkel, J.H.P. Eloff, ,An Overview of Image steganography ICSA research group, south africa,VOL. 4, NO. 3, Jan 2012K. Elissa, "Title of paper if known," unpublished.

[4] Zhichen Ni, Yun Q. Shi, Nirwan Ansari, Wei Su, Qibin Sun and Xiao Lin,"Robust Lossless Image Data Hiding Designed for Semi-Fragile Image Authentication," IEEE Trans, Vol 18, 497-509 (2008)

[5] Khosravi sara, A.D. Mah ,Steganography method based in HIOP IEEE Journal of global research in computer science, VOL.24,NO.1, Jan 2011

[6] D.B.Satre, ,Preserve robustness for image data hiding, IEEE conference on computer research and development, Dec 2010

[7] Michel Barret, Jean-Louis Gutzwiller, and Mohamed Hariti, Low-Complexity Hyperspectral Image Coding Using Exogenous Orthogonal Optimal Spectral Transform (OrthOST) and Degree-2, IEEE Transactions on geoscience and remote sensing, VOL. 49, NO. 5, MAY 2011 1557.

[8] Prof. Pal, IEEE transactions on image processing , Vol 20, No,1 Jan 2011,"An Improved Image Compression Algorithm Using Binary Space Partition Scheme and Geometric Wavelets"

[9] Ning Liu, Palak Amin, K. P. Subbalakshmi, Senior Member,IEEE "Security and Robustness Enhancement for Image Data hiding" IEEE Transactions on Multimedia.

[10] J. Tian,"Reversible data embedding using a difference expansion", IEEE Trans.

[11] Secure JPEG 2000 (JPSEC), ISO/IEC 15444-8:2007, Apr. 2007.