www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2024, pp : 2154-2161

e-ISSN : 2583-1062
Impact Factor : 7.001

A DIFFERENTIAL-CHANNEL CODING APPROACH TO DIGITAL IMAGE PROTECTION AND SELF-RECOVERY USING FINGER PRINT AND FINGER VEIN

S. K. Anusha[1], A. Yesu Raja[2]

[1]Research Scholar, (Reg.no21113092282007), Department of computer science, Muslim Arts College, Thiruvithancode, Affiliated in ManonmaniamSundaranar University, Tirunelveli, India.

Email: anuarshina@gmail.com

[2]Assistant professor, Department of computer science, Muslim Arts College, Thiruvithancode, Affiliated in ManonmaniamSundaranar University, Tirunelveli, India.

Email: a_yesuraja@yahoo.co.in

ABSTRACT

Watermarking algorithms have been widely applied to the field of image forensics recently. One of these very forensic applications is the protection of images against tampering. For this purpose a watermarking algorithm fulfilling two purposes in case of image tampering: 1) detecting the tampered area of the received image and 2) recovering the lost information in the tampered zones is proposed. This project is aimed at showing that having the tampering location known image tampering can be modeled and dealt with as an erasure error. Therefore an appropriate design of channel code can protect the reference bits against tampering. In watermark embedding phase, the original image is source coded and the output bit stream is protected using appropriate channel encoder. For image recovery, erasure locations detected by check bits help channel erasure decoder to retrieve the original source encoded image. In this system a new technique called Differential Channel Coding Approach (DCCA) for secure image transmission is proposed, which watermarks a secret image into a meaningful target image with the different size from the target image. Based on this technique a given large-volume secret image is automatically transformed into a secret-fragment-visible image of the same size. The proposed system also decrease the time required to recover the image when complex patterns exists in the image.

1. INTRODUCTION

Modern digital technology has made it possible to manipulate multi-dimensional signals with systems that range from simple digital circuits to advanced parallel computers. The goal of this manipulation can be divided into three categories

1. Image Processing (image in -> image out)
2. Image Analysis (image in -> measurements out)
3. Image Understanding (image in -> high- level description out)

An image may be considered to contain sub-images sometimes referred to as regions-of- interest, ROIs, or simply regions. This concept reflects the fact that images frequently contain collections of objects each of which can be the basis for a region. In a sophisticated image processing system it should be possible to apply specific image processing operations to selected regions. Thus one part of an image might be processed to suppress motion blur while another part might be processed to improve color rendition.

Digital imaging has been rapidly developing in last two decades, and digital multimedia products are utilized in countless applications nowadays. As a consequence of this expansive development, popular and low-cost access to image editing applications challenges the integrity of digital images. On the other hand, sophisticated techniques are required to guarantee the integrity of an image or protect it against malicious modifications. One common approach is to use the hash of the original image. The receiver declares the image as unaltered if the hash output is the same as the one transmitted from the original image [14]–[15]. Image integrity verification through hash requires a secure channel that must be reused for each image transmission. Since such a channel might be unavailable, a more applicable approach is to embed the verification data into image itself, which is referred to as fragile watermarking.

More recent methods in the field of tampering detection achieve the perfect 100% localization using watermarks robust against wide variety of attacks . On the other hand, watermarking algorithms with the purpose of error concealment aim to restore information in the previously-detected tampered parts [20],[21].

Digital Watermarking can be used for image authentication research, an original image is divided into smaller blocks, and inserts a watermark into each and every block. After that modification concept is used to detect the watermark in the blocks. A high localization of tampering detection is achieved by applying a random permutation process where every embedded watermark bit verifies random image positions instead of a local image block. Thereby the resolution of tampering detection is significantly improved in comparison to existing solutions while keeping the payload low. The

proposed scheme doesn't embed the watermark locally but distributes it into the suitable embedding wavelet coefficients, avoiding embedding in smooth regions. Therefore, the scheme is intrinsically secure to block-based local attacks and retains high fidelity of the watermarked image.

Fragile watermarks can be used for both authentication of the received image and localization of tampered zone in case of malicious modifications (tampering localization), and recovering the image information in the lost area (error concealment). Inceptive fragile watermarking techniques aim only to verify the integrity of image or locate the tampered area with limited robustness against image processing modifications.

This self-recovery watermarking trend, initiated by [9],[11] has recently attracted growing interest. The problem of image self-recovery has been approached in numerous ways. In [28], conventional error control coding schemes are adopted for localization and restoration. Several methods embed a rep- resentation of an original image into itself for the sake of self-recovery. In [9],[11] discrete cosine transform (DCT) coefficients or reduced color-depth version of the host image is embedded in the least significant bits (LSB) of the original image. This representation of the original image can also be the first few DCT coefficients of each block [3],[4],[19] a binary image generated from the difference between the host image and its chaotic pattern [21], the hash of the original image [22], watermark derived from approximation coeffi- cients of its wavelet transform [29], a vector quantized or halftone [12] version of the original image. Fragile water- marks may also be designed for specific purposes, such as binary images, JPEG compressed images, colored image , [8][25][26] compression-resistant [9] or cropping- resistant applications [10].

The check bits support the receiver in locating the tampered blocks. The receiver knows the exact location of erroneous bits. Tampering is modeled as an erasure error in this way. Thus, an RS channel erasure decoder for image recovery at the receiver is required. The lengths of the channel encoder input and output blocks are long to achieve the best performance. Setting up the RS channel codes over GF (2t +1) instead of G(2t) is another suggestion of this paper which greatly simplifies the complexity of channel encoder and decoder implementation. It is shown that our watermarking scheme which replaces only two LSB of an image, efficiently recovers the tampering up to 33% without leaving any noticeable distortion. If the implemented algorithm is using 3 LSB, it totally outperforms the state-of-the-art methods using the same three LSB for watermarking. The proposed scheme is implemented for two certain sets of parameters; it can be flexibly adapted to different applications with different purpose.

The problem of image self-recovery is about finding an appropriate trade-off between these three parameters: the watermarked image quality, content recovery quality, and tolerable tampering rate (TTR). The size of watermark determines the amount of imposed distortion and the quality of the watermarked image.

In this image self-recovery algorithm using these two key ideas: i) Modeling image representation and reference bit generation as a source coding problem; ii) Modeling the tampering as an erasure channel while handling it with proper channel coding. The location of tampered areas being identified through check bits, tampering can be modeled as an erasure channel, where the locations of occurring errors are known to the receiver. Erasure modeling of tampering has been recently offered and exploited in [2] where the authors apply fountain codes to deal with it. It should be added that when one block is marked as tampered, all its carrying reference bits are missed. We would suggest Reed-Solomon (RS) [23] codes with large encoding blocks and over large Galva fields to solve the erasure problem. Moreover, we treat the challenge of finding some representation of the original image as a source coding problem. We apply the wavelet transform and set partitioning in hierarchical transforms (SPIHT) source encoding method [1] to efficiently compress the original image. Therefore, the watermark consists of three parts in our algorithm: source code bits, channel code parity bits and check bits. Source code bits which act as the reference bits are the bit stream of the SPIHT-compressed original image at a desired rate.

In order to survive tampering erasure, the reference bits are channel coded to produce channel code bits. Check bits are used at the receiver to determine the erasure location for the channel erasure decoder. The output of channel decoder is source decoded to find the compressed version of the original image. This work shows that by choosing appropriate parameters for source and channel encoding, our algorithm outperforms existing methods in the same watermark payload of three bits per pixel (bpp). Nevertheless, since the watermark artifacts are significant for embedding in three LSB, we would recommend two-LSB version of our algorithm and show that its performance is still remarkable.

This paper proceeds as follows. Section II briefly reviews some of the state-of-the-art self-embedding schemes. Section III presents watermark Embedding in general, while its components are explained in details in the subsequent sections. Section IV introduces SPIHT as our chosen source coding method. The RS channel coding is investigated in Section V as our choice for channel encoding to combat the channel erasure. Experimental results are presented and discussed in Section VI, and Section VII concludes the paper.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2024, pp : 2154-2161
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

2. RELATED WORKS

In self-embedding the main task is image quality during transfer of image, and image watermarking for proprietorship. Recovery fractal codes of the image blocks are done after embedding of image using Least Significant Bit technique to avoid packet loss. Reverse process is done at receiver end where fractal codes obtain at receiver end is convert into original image then extraction of watermark is done. Make a matrix of the same dimension of the image then fill the matrix correspond to the pixel value of the image at the cell in the matrix.

Now convert pixel values in binary form, this is required for the watermarking. As LSB technique is used to pixel value representation should be binary instead of decimal. In LSB Embedding watermark information which is binary form is replace by least significant bit of the pixel value at edge position. Embedding message is read for file and generate corresponding ASCII value. In Make Image Block, image is divided into same size of blocks where size can be 3X3 in this case 3 means number of pixel in the row and column. Generate Fractal Code Different combination for block of image are pass into function $(F_n - XOR(B_{m,}\ X_{n,m,}F_n)$ where n represent number of blocks to send in network, while m represent number of image block. Obtained fractal codes are send in network. It has been observed that for every six block of image corresponding eight block is generate by different combination of blocks.

In Hash Value Checking blocks obtained from the network may get corrupt to check this hash value is generated from the block and checked. Proprietorship of image is base on the validate by watermark so watermark is extract from the image. Here LSB bits present of the edge pixels are extract for watermark construction.

The sparse DCT coefficients of the image blocks are then under sampled using a pseudorandom matrix satisfying the restricted isometry property (RIP) required for the compressive sensing and sparse processing. The resulting projected values are then non-uniformly quantized and embedded as the watermarked bits. The reference data lost due to tampering is recovered at the receiver either using a comprehensive sensing or compositive reconstruction approach depending on whether the amount of the surviving reference data is below or above a certain limit, respectively. The reference data in [45] is generated by the least square quantization of the DCT coefficients. This information is then channel coded with the rate λ and embedded as the watermark data.

Therefore, the λ parameter determines the trade-off between the quality of the restored image and TTR for a certain embedding capacity. The higher λ values mean lower channel code protection and hence lower TTR. However, in this case the embedding capacity is dedicated more to the reference data and the lost data will be recovered with a higher quality while the tampering rate is below TTR. On the other hand, the embedding capacity is rather dedicated to the channel coding parity bits than reference bits for smaller λ cases, in which the restoration is possible with low quality for the tampering rates up to higher TTRs.

3. PROPOSED SCHEME

a. Watermark Embedding

Consider the original image I represented as 8-bit gray-scale pixel values. These eight bits are divided into four parts: The most significant bits (MSB) that will not change at the watermark embedding phase, check bits, source code bits, and channel code parity bits, denoted by nm, nh, ns and np, respectively. The nm MSB bits of each pixel remain unchanged during watermark embedding and will be used later for hash generation and image reconstruction. The remaining bits are used for the purpose of watermark embedding.

Assume the number of image pixels are N = N1 × N2, where N1 and N2 stand for numbers of rows and columns of the original image. We compress the original image into Ns = N × ns bits using proper source coding algorithm (SPIHT here). A channel coding algorithm (Reed-Solomon code here) of rate R = ns/nc is applied to permuted com- pressed image bit stream, where nc = ns +np. Channel code yields Nc = N ×nc bits in total. These bits are permuted and spread over the whole image, which means every pixel will host ns source code bits and np channel code parity bits. The permutations before and after channel coding are generated using keys k1 and k2, both derived from a secret key K, which is known to both embedding phase (transmitter end) and image reconstruction phase (receiver end), to guarantee the security of our algorithm. The original image is also divided into blocks of size B×B, thus each block will host bc = nc × B2 channel code bits. These bc bits originally belonged to some other blocks, whose rows and indices are turned into a binary stream of brc bits called position bits. These brc position bits along with bm = nm ×B2 MSB bits of each block are used as input to a hash generator algorithm (MD5 here), to produce bh = nh×B2 hash bits. A random binary key of length bh fixed over the whole image is generated at the embedding phase. This key is XORed with hash bits to generate bh check bits. These bh check bits along with bc channel code bits of each block are spread over the block which results in replacing last nw = nc + nh least significant bits of each pixel of the original image, where nw is the number of LSB per pixel used for watermark embedding. After having all blocks processed, watermarked image is produced. To summarize, nm MSB of each pixel are preserved and nw = 8−nm LSB are replaced with watermark bits

www.ijprems.com
editor@ijprems.com
INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2024, pp : 2154-2161
e-ISSN : 2583-1062
Impact Factor : 7.001

during embedding process. These nw bits consist of ns source code bits, np channel code parity bits, and nh check bits. nw is not necessarily an integer. For instance, one may use two or three LSB bits in each block for watermark insertion alternatively. In this case, we have nw = 2.5. For the sake of simplicity, we assume integer nw (nm) hereafter. In the case that nw LSB of each pixel is used for the sake of watermark insertion, our algorithm is called nw-LSB. Block diagram of watermark embedding for 2-LSB algorithm is shown in Fig. 1. In this case, nw, ns, np and nh are equal to 2, 1, 0.5 and 0.5, respectively.

b. DCCA Approach

In watermark embedding phase, the original image is source coded and the output bit stream is protected using appropriate channel encoder. For image recovery, erasure locations detected by check bits help, channel erasure decoder to retrieve the original source encoded image. For secure transmission, a new technique called Differential Channel Coding Approach (DCCA) is proposed, which watermarks a secret image into a meaningful target image with the different size from the target image. Based on this technique a given large-volume secret image is automatically transformed into a secret-fragment-visible image of the same size. The result shows that the time required to recover the image is less than the existing approaches when complex patterns exists in the image.

If the size of the secret image is dS is different from that of the target image T, change the size of dS to be identical to that of T; and divide the secret image dS into n tile images {dS1, dS2, . . . , dSn} as well as the target image T into n target blocks {B1, B2, . . . , Bn} with each *Ti* or *Bi* being of size *NT*.

Compute the means and the standard deviations of each tile image *Ti* and each target block *Bj* for the three color channels according and compute accordingly the average standard deviations for *Ti* and *Bj* , respectively, for $i = 1$ through n and $j = 1$ through n.

Sort the tile images in the set dS tile = {T1, T2, . . . ,Tn} and the target blocks in the set dS target = {B1,B2, . . . , Bn} according to the computed average standard deviation values of the blocks; map in order the blocks in the sorted Stile to those in the sorted S target in a 1-to-1 manner. Create an identical image dF by fitting the tile images into the corresponding target blocks according to size of the target image.

For secret image dS in identical image dF, construct a bit stream Mi for recovering dS including the bit-segments which encode the data items 1) the index of the corresponding target block Bji; 2) the means of Ti and Bji and the related standard deviation quotients of all three color channels; 4) the bit sequence for overflows/underflows.

Concatenate the bit streams Mi of all dS in dF in a raster-scan order to form a total bit stream Mt ; use the secret key K to encrypt Mt into another bitstream Mt ; and embed M t into dF by the reversible contrast mapping scheme.

C. SPIHT Algorithm For Image Compression

Set Partitioning In Hierarchical Trees (SPIHT) encoding is applied as source encoder in the proposed method. SPIHT is an embedded compression algorithm, in which one can truncate its output bit stream at the desired rate and come to a certain reconstruction of the original image. The more output rate exploited, the better quality of reconstruction is achievable.

The SPIHT method is not a simple extension of traditional methods for image compression, and represents an important advance in the field. The SPIHT is an efficient image coding method using the wavelet transform. Recently, image-coding using the wavelet transform has attracted great attention. Among the many coding algorithms, the embedded zero tree wavelet coding by Shapiro and its improved version, the set partitioning in hierarchical trees by Said and Pearlman have been very successful. Compared with JPEG the current standard for still image compression, the EZW and the SPIHT are more efficient and reduce the blocking artifact. The algorithm sorts the rounded multi-resolution wavelet transform coefficients according to their magnitudes and transmits them based on significant bit order. The sorting order must be available to the decoder as well. A sophisticated sorting method is required to decrease the "bit-budget" used for sorting pass. SPIHT exploits the self-similarities across different sub bands of wavelet transform. Beside the low computational complexity, the fact that SPIHT is an embedded compression algorithm with adaptive output rate makes it suitable for our application in which we may need to exploit different compression rates to satisfy different purposes.

Our algorithm truncates the SPIHT output at the rate of ns bits per pixel. Channel coding is applied to source encoder output bit stream to protect it against tampering. The maximum achievable peak signal to noise ratio (PSNR) of our reconstruction algorithm happens when channel code has worked perfectly and retrieved all source encoded bits, and equals the PSNR of SPIHT for original image.

d. RS- Channel Coding

The source encoder outcome must be protected through some channel codes. The tampered blocks will be recognized using check bits. It is worthy that their information is available to channel decoder. Considering the source-channel code

www.ijprems.com
editor@ijprems.com
INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2024, pp : 2154-2161
e-ISSN : 2583-1062
Impact Factor : 7.001

design and having error locations available, tampering can be modeled and treated as an erasure error, where the locations of error are known to decoder. An erasure decoder uses the channel coded data in preserved blocks and the locations of erasure must be implemented for the purpose of image recovery. When a block is recognized as tampered, all its bc embedded channel code bits are assumed to be erased.

RS codes with large code words over large fields tackle this challenge effectively. The use of RS codes with large code words, several bits are congregated to one symbol, resulting in limited number of code words affected by image tampering. Image is compressed at the rate of ns bpp, and nc = ns +np bpp is dedicated to channel code, thus the rate of used RS(n,k) must be R = k/n = ns/nc. Suppose that we want to construct our RS(n,k) code over GF(2t), that is, the code words can be represented by t bits. Limit our choice of t on those that divide bc, i.e bc =kct for some positive integer kc. The bc dedicated channel code bits are actually binary representation of kc output code words of channel code. For each tampered-recognized block, we make sure only kc code words are affected (erased). From this point of view t = bc is the best choice that ensures us every tampered block affects only one code word.

Implementing RS codes over GF(2bc) for large bc might be overcomplicated and practically not reliable. An elegant tradeoff between implementation complexity and erasure recovery performance of the code must be regarded in this phase i.e, t must be chosen in a way that it divides bc and is not too large for the construction of a feasible code. After choosing proper t, we need to find the length of channel code input and output blocks, k and n, respectively. At this point, source code at the rate of ns bpp has led to Ns = N ×ns source encoded bits, or Ns/t code words. Total bit-budget of nc = ns +np bits per pixel is also dedicated to channel code outputs, resulting in Nc/t output code words. The Shannon theory reveals that the channel code exhibits its best performance when the length of coding block is the greatest possible.

The ideal case is where we encode the whole image into one block of channel code, that is, where RS(Nc/t, Ns/t) is feasible over GF(2t). This is possible by means of puncturing a mother RS code , only when Nc/t < 2t. In this case, we choose n = Nc/t and k = Ns/t, else we encode the whole image by i iterations of channel encoder in which i =(Nc/t/2t). In this situation k and n take appropriate values about Nc/t/i and Ns/t/i, respectively. RS codes are directly feasible for every (n,k) where n divides 2t − 1 (maximum order of the field members). Encoder and decoders are implemented by puncturing a mother RS(m,k) code where m = min ˆ m (ˆ m|2t −1 andˆ m > n). (1) Implementation of conventional RS(n,k) encoder and erasure decoder over GF(2t) might be complicated for the case of large t. But in some cases, we can simplify our encoding-erasure decoding problem, in particular when 2t +1 is a prime number.

This opportunity is available only in case of GF(2t +1) in which the length of DFT's used is a power of two. The only especial point about GF(2t+1) that should be considered is that since it represent the output code words using t bits, the symbol 2t must be avoided in the output of decoder. It can be done by simply choosing appropriate k1 secret key (permutation control at the input of channel encoder) or negligibly modifying the contents of the original image. Encoder and erasure decoder algorithms over GF(216 + 1) are efficiently implemented using FFT, which are described in appendices A and B, respectively. At the end of this section, to investigate the tampering protection performance of RS codes. RS(n,k) is capable of correcting n − k erasures . In the case that the whole image is encoded using one block of RS(n,k) = RS(Nc/t, Ns/t), then k erasure out of n code words is tolerable at the decoder. The tolerable tampering rate (TTR) of the channel code is equal to: TTR(ns,nc)= n−k n = 1− ns nc . It declares that via dedicating ns and nc bpp for source encoder and channel encoder outputs, the proposed method will be capable of recovering the tampered area of image if its size does not exceed the fraction of 1−ns/nc of the total image size. The source encoder output bits will be perfectly retrieved and tampered zones will be recovered with maximum possible PSNR of the source encoder algorithm.

4. EXPERIMENTAL RESULTS

A Binary finger print watermarked using our proposed method. The original Binary finger print is shown in Fig. 1(a). Fig. 1(b) shows the watermarked image generated by 2-LSB version of our algorithm. As mentioned, the PSNR of watermarked image generated by 2-LSB version of our algorithm equals 43.58 dB, which is far beyond the HVS threshold of noticeable distortion. State-of-the-art tampering protection algorithms usually use three least significant bits for watermark insertion. This embedding approach degrades the PSNR of watermarked image down to 37.9 dB, which is not suitable for smooth areas.

Set the block size B = 8. The first parameter to choose is the number of LSB used for watermark insertion, nw. This parameter directly affects the visual quality of the watermarked image. For integer nw, if we replace the last nw LSB of pixels with watermark bits, the average energy of distortion imposed by watermarking measured by MSE (Mean Square Error) equals:

$$\text{MSE}_{\text{nw}} = \frac{4^{nw}-1}{6}$$

Therefore, the average peak signal to noise ratio (PSNR) is calculated as:

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2024, pp : 2154-2161
e-ISSN : 2583-1062
Impact Factor : 7.001
www.ijprems.com
editor@ijprems.com

$PSNR_{(nw)} = 10 \log_{10} \left(\frac{255^2}{MSE_{nw}}\right)$

The tampering protection performance of our algorithm is also investigated in practice. Both "low-rate" and "high-rate" tampering scenarios are applied to Binary finger vein. Fig.2 depicts the result of low-rate tampering protection. In this case, 2-LSB version of our algorithm has been applied. This watermark is supposed to tolerate tampering rate of up to 33%. The original and watermarked images are the same as Figs. 1(a) and 1(b).

1(a) 1(b)

Fig.1(a) original Finger vein image 1(b) Watermarked image using 2LSB.

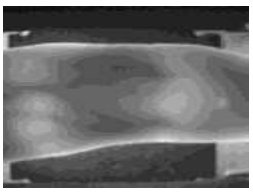

Fig. 2: Secret Finger vein image

The values derived for PSNR of watermarked image are constant and independent of the chosen host image, in spite of the reconstruction PSNR which varies depending on the selected cover image.

5. CONCLUSION

This work, introduced a watermarking scheme to protect images against tampering. The watermark bit-budget falls into three parts, check bits, source encoder output bits, and channel encoder parity bits. The original image is source coded using SPIHT with DCCA compression algorithm. The source encoder output bit stream is channel coded using RS code of a required rate and over appropriate field. Since image tampering affects a burst of bits, the RS codes over large Galva fields are wise choices. On the other hand, check bits support the receiver in locating the tampered blocks. Therefore, the receiver knows the exact location of erroneous bits. Tampering is modeled as an erasure error in this way. It need an RS channel erasure decoder for image recovery at the receiver. The lengths of the channel encoder input and output blocks are also taken as long as possible to achieve the best performance. This watermarking scheme which replaces only two LSB of an image, efficiently recovers the tampering up to 40% without leaving any noticeable distortion. The peak signal to noise ratio is comparatively high, obtaining better image recovery.

In Future, various improvements in SPIHT with DCCA algorithm can be made in the areas of speed with high PSNR, resilience and memory requirement. The algorithm can be implement using another LSB bits to obtain better PSNR, to increase the protection of images for security. In future one can embed video using same approach for optimizing resource utilization.

6. REFERENCE

[1] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. Circuits Syst. Video Technol., vol. 6, no. 3, pp. 243–250, (Jun. 1996).

[2] P. Korus and A. Dziech, "Efficient method for content reconstruction with self-embedding," IEEE Trans. Image Process., vol. 22, no. 3, pp. 1134–1147, (Mar. 2013).

[3] Z. Qian, G. Feng, X. Zhang, and S. Wang, "Image self-embedding with high-quality restoration capability," Digital Signal Process., vol. 21, no. 2, pp. 278–286,(2011).

[4] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Self-embedding watermark with flexible restoration quality," Multimedia Tools Appl., vol. 54, no. 2, pp. 385–395, (2011).

[5] X. Zhang and S. Wang, "Fragile watermarking scheme using a hierarichical mechanism," Signal Process., vol. 89, no. 4, pp. 675–679, (2009).

[6] T.-Y. Lee and S. D. Lin, "Dual watermark for image tamper detection and recovery," Pattern Recognit., vol. 41, no. 11, pp. 3497–3506, (2008).

[7] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference sharing mech- anism for watermark self-embedding," IEEE Trans. Image Pro+cess., vol. 20, no. 2, pp. 485–495, (Feb. 2011).
[8] X. Zhu, A. T. S. Ho, and P. Marziliano, "Image authentication and restoration using irregular sampling for traffic enforcement applications," in Proc. 1st Int. Conf. Innov. Comput., Inf. Control (ICICIC), vol. 3, pp. 62–65.(Aug./Sep. 2006).
[9] C.-Y. Lin and S.-F. Chang, "SARI: Self-authentication-and-recovery image watermarking system," in Proc. 9th ACM Int. Conf. Multimedia (MULTIMEDIA), pp. 628–629,(2001).
[10] S. Bravo-Solorio, C.-T. Li, and A. K. Nandi, "Watermarking method with exact self-propagating restoration capabilities," in Proc. IEEE Int. Workshop Inf. Forensics Security (WIFS), pp. 217–222, (Dec. 2012).
[11] N. Wang and C.-H. Kim, "Tamper detection and self-recovery algo- rithm of color image based on robust embedding of dual visual watermarks using DWT-SVD," in Proc. 9th Int. Symp. Commun. Inf. Technol. (ISCIT), pp. 157–162,(Sep. 2009).
[12] A. Cheddad, J. Condell, K. Curran, and P. McKevitt, "A secure and improved self-embedding algorithm to combat digital document forgery," Signal Process., vol. 89, no. 12, pp. 2324–2332, (2009).
[13] C.-S. Lu, S.-K. Huang, C.-J. Sze, and H.-Y. M. Liao, "Cocktail water- marking for digital image protection," IEEE Trans. Multimedia, vol. 2, no. 4, pp. 209–224, (Dec. 2000).
[14] M. Tagliasacchi, G. Valenzise, and S. Tubaro, "Hash-based identification of sparse image tampering," IEEE Trans. Image Process., vol. 18, no. 11, pp. 2491–2504, (Nov. 2009).
[15] A. Swaminathan, Y. Mao, and M. Wu, "Robust and secure image hashing," IEEE Trans. Inf. Forensics Security, vol. 1, no. 2, pp. 215–230, (Jun. 2006).
[16] K.K.Amutha , N.A.Vidya Mol, "An Efficient Active Content Reconstruction Based on Adaptive Pixel Permutation," International Conference on Humming Bird (1st March 2014)
[17] Awanish Kr Kaushik , "A Novel Approach for Digital Watermarking of an Image Using DFT," International Journal of Electronics and Computer Science Engineering ISSN-2277-1956
[18] Zhenxing Qian , Lili Zhao , "A Flexible Scheme of Self Recovery for Digital Image Protection ," IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 3,(November 2012).
[19] Bali Sharma, S. V. Pandit, "Image Self Embedding and Watermarking using Least Significant Bit," Volume 5, Issue 6, International Journal of Advanced Research in Computer Science and Software Engineering,(June 2015).
[20] X. Zhang and S. Wang, "Fragile watermarking with error-free restoration capability," IEEE Trans. Multimedia, vol. 10, no. 8, pp. 1490–1499, (Dec. 2008).
[21] X. Zhang and S. Wang, "Statistical fragile watermarking capable of locating individual tampered pixels," IEEE Signal Process. Lett., vol. 14, no. 10, pp. 727–730, (Oct. 2007).
[22] Chao-Ming Wu, Yan-Shuo Shih, "A Simple Image Tamper Detection and Recovery Based on Fragile Watermark with One Parity Section and Two Restoration Sections,"
[23] Optics and Photonics Journal, 2013, 3, 103-107 Published Online (8June 2013).--
[24] S. B. Wicker, Reed–Solomon Codes and Their Applications. Piscataway, NJ, USA: IEEE Press,(1994).
[25] Luo, T., Jiang, G., Wang, X. et al.," Stereo image watermarking scheme for authentication with self-recovery capability using inter-view reference sharing," Multimedia Tools and Applications, Volume 73, Issue 3, pp 1077–1102,(December 2014)
[26] Clara Cruz-Ramos, Rogelio Reyes-Reyes, Mariko Nakano-Miyatake, Héctor Pérez-Meana," Image Authentication Scheme Based on Self-embedding Watermarking," Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications Volume 5856 of the series
[27] M. U. Celik, G. Sharma, E. Saber, and A. M. Tekalp, "Hierarchical watermarking for secure image authentication with localization," IEEE Trans. Image Process., vol. 11, no. 6, pp. 585–595,(Jun. 2002).
[28] J. Fridrich and M. Goljan, "Images with self-correcting capabilities," in Proc. Int. Conf. Image Process. (ICIP), vol. 3. 1999, pp. 792–796.
[29] J. Lee and C. S. Won, "Authentication and correction of digital watermarking images," Electron. Lett., vol. 35, no. 11, pp. 886–887, (1999).
[30] R. Chamlawi, A. Khan, and I. Usman, "Authentication and recovery of images using multiple watermarks," Comput. Elect. Eng., vol. 36, no. 3, pp. 578–584, (2010).
[31] S. B. Wicker, Reed–Solomon Codes and Their Applications. Piscataway, NJ, USA: IEEE Press, (1994).
[32] V.P.Ananthi1, P. Balasubramaniam and P.

[33] Raveendran, "Impulse noise detection technique based on fuzzy set", IET jounals the institution of Engineering and technology, vol. 12, issue 1, pp. 12-21, 2018.
[34] Karen Panetta, Long Bao and Sos Agaian, "A new unified impulse noise removal algorithm using a new reference sequence–to-sequence similarity detector", IEEE Journal ,Vol. 6 , issue 1, pp. 37225 - 37236, year 2018.
[35] Ugur Erkan, Dang Ngoc Hoang Thanh, Le
[36] Minh hieu An and Serdar Enginoglu, "An
[37] iterative mean filter for image denoising", IEEE
[38] Access, vol. 7, pp. 167847-167859, 2019.
[39] Xiaoqin Zhang Jingjing Zheng, Di Wang and Li
[40] Zhao, "Exemplar based denoising a unified low rank recovery framework", IEEE journal early access article, pp. 1, 2019.
[41] Min Liu, Xinggan Zhang and Lan Tang,"Weighted t-Schatten-p norm minimization for real color image denoising", IEEE Jrnl, vol.8, pp. 150350-150359, 2020.