Maintaining the Confidentiality of Images: A New Perspective Based on Digital Watermarking

Yogendra Kushwaha[1], Narendra Kumar Gupta[2]
[1]M.Tech. Student, Department of Computer Science and Information Technology,
Shepherd School of Engineering and Technology, SHIATS Allahabad-211007, (U.P.), India
Kushwaha.yogendra@gmail.com
[2]Assistant Professor, Department of Computer Science and Information Technology,
Shepherd School of Engineering and Technology, SHIATS Allahabad-211007, (U.P.), India
Narendra_nkg04@rediffmail.com

Abstract **- Digital watermarking is a technique that secures important information by multimedia objects such as image, text or other digital objects. In this paper a chunk-based fragile watermarking technique based on partitioning around medoids (PAM) clustering algorithm has been proposed. This algorithm is superior enough to recover from the tapered region of an image. In the proposed algorithm, images are divided into chunks and each chunk has assigned 36 bits, consists of 32 recovery bits and 4 authentication bits. Mapping of these 36 bits for each chunk are pseudo randomly mapped with some other chunk by applying the same secret key. At the receiver side the extracted image is compared with authentication bits and from mapping bits, tampered chunk with extracted recovery bits provide significant information to again recover the host image. Performance of the proposed algorithm in comparison to Discrete Cosine Transform (DCT), Singular Value Decomposition (SVD) and Discrete Wavelet Transform (DWT) is checked over the experimental results obtained from PSNR, MSE and recovery time. It is found that the proposed algorithm shows its superiority over other state of the art algorithms for maintaining the confidentiality of an image.**

Keywords **- Chunk -based fragile watermarking, image alteration, image recovery, image authentication, image restoration.**

1. Introduction

Watermarking is the technique of altering a work to embed a message about that work. This technique contains two high level elements Embedder and Detector as shown in figure 1. The embedder takes two inputs. One is the payload which is to be embedded (e.g. either the watermark or secret message) and other is the cover work in which the payload is embedded. The output of embedder is presented as an input to the detector. Watermarking is applied in various fields to provide security. Some watermarking applications are broadcast monitoring, owner identification, transaction tracking, content authentication, copy control, device control and legacy enhancement.

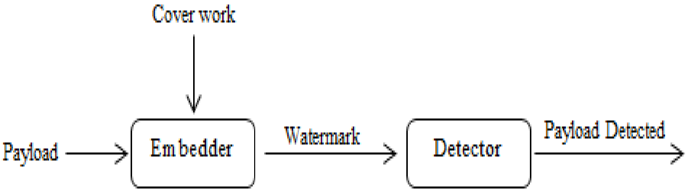


Figure 1: A generic watermarking system.

There are three categories of watermarking like fragile [1], semi fragile [2] and robust watermark [3].For guaranteeing the authenticity and information respectability the idea of fragile watermarking[1] [4] [5] [6]came in picture. A fragile watermark is an imprint that is promptly altered or demolished at the point when the host picture is adjusted through a linear or nonlinear transformation [2] .The sensitivity of fragile watermark to adjustment prompts their utilization in image authentication. Fragile watermarking can be isolated into two noteworthy classes i.e. Block wise fragile watermarking [7] and Pixel wise fragile watermarking [8] [9].

The fast extension of the web in the previous years has quickly expanded the accessibility of computerized information, for example, sound, pictures and recordings to the general population. As we have seen in the previous couple of years, the issue of ensuring sight and sound data turns out to be more and more vital and a ton of copyright proprietors are worried about securing any illicit duplication or any intentional change of their information or work. A few genuine work should be done keeping in mind the end goal to keep up the accessibility of media data at the same time, in the interim, the industry must concoct approaches to ensure licensed innovation of makers, wholesalers or basic proprietors of such information. This is an intriguing test and this is likely why so much consideration has been drawn toward the advancement of digital data protection plans.

We can expect a contextual investigation, assume someone has made a site and transferred some individual pictures on it, after some time he found that same pictures with some objectionable alteration are available in distinctive site. In this circumstance there will be strife that, which one is the right picture. This contention may be determined by fragile watermarking scheme by utilizing third trusted party. On the off chance that all pictures are watermarked utilizing fragile watermarking by trusted outsider and transferred to site then at the time of conflict, utilizing extraction algorithm we can realize what one is original picture.

Further the paper has been divided in to four sections: section 2 is dedicated for literature review. In which previous work on digital watermarking has been summarized. Proposed work is discussed in section 3. Experimental result and analysis of result is shown in section 4. Finally the paper is concluded in section 5 with future work.

II. Literature Review

The sudden increment in watermarking is in all likelihood because of the increase in worry over copyright security of substance and content validation of digital media. The web is an astounding appropriation framework for digital media since it is economical and delivery is verging on immediate. However the proprietor of the substance likewise sees a high risk of piracy.

The danger of piracy is exacerbated by the expansion of high-limit digital recording devices. Utilizing these recording devices and utilizing the web for conveyance privateers can effortlessly record modify and redistribute the copyright secured material without suitable compensation being paid to the genuine copyright proprietors. In this way content proprietors are enthusiastically looking for advancements that guarantee to secure their rights.

The principal innovation content proprietors swing to be cryptography. Cryptography is likely most regular strategy for securing digital media. In the event of cryptography it just protect the content in transmission, however once decrypted, the content has no further security. Therefore there is a significant requirement for an alternative to the cryptography: an innovation that can secure content even after it is decrypted. Watermarking has the potential to satisfy this need. Some literature work has been deeply reviewed which are as follows.

The fragile watermarking scheme [10] was proposed by P. Wong et.al. This approach is based on secret key and public key cryptography. In this work a different image is used as watermark instead of self-embedding technique. Another fragile watermarking approach [11] is proposed by Hongjie is a standard technique to detect alteration. The embedded watermark is generated with the help of discrete wavelet transform (DWT), and then the improved security watermark by embedding scrambling encryption into the least significant bit (LSB) of the host image. Fragile watermarking approach having restoration capability is proposed by Seng et al. [12]. Here cover work is converted in to wavelet domain by DWT and higher bits of watermark is embedded in to lower sub-band of wavelet and the lower order bits are embedded into another wavelet sub-band. The stego image is obtained by an inverse transform from the wavelet domain into the spatial domain. For authentication of a test image, a DWT is performed and watermark is removed from the wavelet sub-bands. Then comparison of the watermark is performed with a down-scale version of the test image. Then similarity check is performed between them if it found more than the threshold value, then the test image is classified as authentic. Otherwise, the affected region is highlighted and recovery of content is carried out with the extracted watermark information. Juan R. Hernández et al. [13] have applied the Discrete Cosine Transform (DCT) watermarking algorithm on digital images for copyright protection. They applied the probability as a measurement of the error in watermark decoding and detection of watermark. In 2010, Deepa Mathew K et al. [14] made a discussion that SVD based watermarking is very reliable to increase the robustness of an image. They showed that the SVD based watermarking is more secure and robust. For embedding watermark, they applied D and U components. The result obtained by SVD was good in accuracy, robustness and imperceptibility of the recovered watermarked image.

III. Proposed work

Proposed approach is based on partitioning around medoids (PAM) algorithm and that is a representative object based technique [15]. This proposed algorithm is made purely to work in spatial domain as shown in figure 2.PAM breaks the dataset into groups. It works to minimize the distance between the points that are found in the cluster and the centre point of the cluster. Some assumptions are taken for the proposed work. For example, assume a gray scale native image has dimension size m x n. Then the total number of pixels is denoted by N = m x n. Px_i is used to denote the gray scale value at each pixel of an image. Where Px_i is belongs to [0...255] and $i= 1, 2, 3 ....N$. This Px_i can be represented by 8 bits. The proposed work is divided into two parts i.e. watermark embedding and watermark extraction. Working model of the proposed work is shown in figure 2.

3.1 Watermark Embedding Process

This process has four different stages such as clustering of pixels, generation of recovery bit, generation of authentication bit and chunk mapping.

3.1.1 Clustering of pixels

1) Remove first 2 LSBs of all pixels for removing the gray scale value from [0,255] to [0, 15].Now 4 binary bits will be used to represent Px_i. This process can be understood as follows:

2) Now the image will be divided into number of chunks and the size of each chunk having 3 x 3 dimension. So each chunk will contain 9 gray scale values. Hence the total number of chunks will be N/9.

3) Each chunk containing 9 gray values will be taken as input to the PAM clustering algorithm and four different clusters will be created. If we have n dataset and c number of clusters is required then the PAM algorithm will be written as follows:

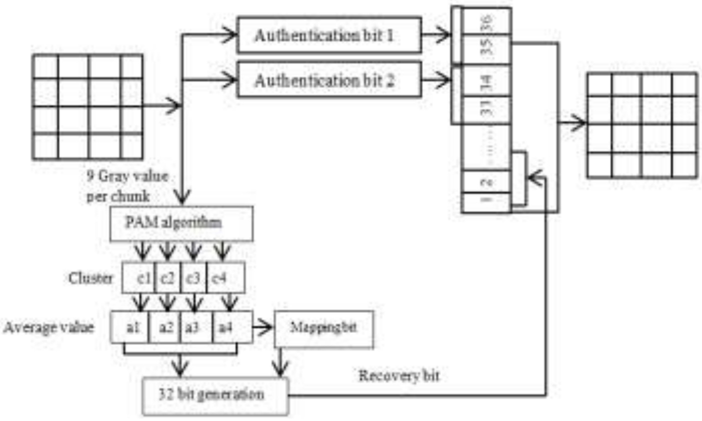


Figure 2: proposed Working model of the work.

X_{11}	X_{12}	X_{13}
X_{21}	X_{22}	X_{23}
X_{31}	X_{32}	X_{33}

1. Randomly choose k gray values in a chunk as the initial representative seeds.
2. Do
3. Initially take dataset to be clustered (Input: D = {d1,d2,...dn})
4. Randomly choose k number of clusters
5. Use flag to indicate D is the matrix of dissimilarity or not
6. Output: V = {v1,v2,...,vk} (vector of clusters medoids)
7. Assign every remaining gray value to the cluster with nearest representative seed.
8. Arbitrarily choose a non representative gray value.
9. Compute the total cost Ct of representative gray value with non representative gray value. Here cost is calculated by taking difference between both gray values.
10. If Ct< 0 then swap the representative gray value with non representative gray value.
11. Until no change is found.

4) At the end of clustering process we get 4 clusters for each chunk. Then we calculate the average for each cluster in round integer value. Now the averages are a1, a2, a3 and a4 then rearrange all average a1, a2, a3 and a4 in ascending order.

3.1.2 Generation of Recovery Bit

In this phase we generate 32 recovery bits for each chunk and these are named as vector V. Suppose that a1, a2, a3 and a4 are four averages for a chunk and ascending order of these are a1, a2, a3, and a4.

1) Perform mapping of average values to their respective two bit pattern, shown in table 1.

2) Start converting the max average value i.e. a4 into 4 bit binary form and put these values to the first four indexes of vector V.
3) Calculate the following
d1 = a4-a3, d2 = a4-a2
d3 = a4-a1, d4 = d3-d1
d5 = d3-d2
4) Here d1, d2, d3, d4, d5 represent the decimal value. Now convert d1, d2, d3, d4 and d5 into 2 bit binary form and put these values into the next 10 bit indexes of vector V. Up to this stage 14 index position of vector V have been covered. We need 18 more bits for recovery through calculation.

5) Now start mapping of 9 gray level values using two bit binary bits shown in table 1 used for mapping for their respective average values of those clusters in which they belongs to. We can understand this process with an example, given below:

Ch =

Suppose Ch is one of the chunks with 9 gray level values. Four clusters have been formed by applying the clustering algorithm (PAM) which is as follows:

C1 = {X_{11}, X_{12}, X_{22}, X_{23}, X_{21}}
C2 = {$X_{13,}$ X_{31}, X_{32}}
C3 = {X_{33}, X_{21}, X_{11}}
C4 = {X_{33}, X_{32}, X_{21}, X_{22}, X_{23}}

Since averages a1, a2, a3, a4 are calculated for their corresponding clusters C1, C2, C3 and C4 and arranged in the ascending order such as a1,a2,a3,a4.Mapping of average is done through table 1 like a4 is mapped with 00, a3 is mapped with 01, a2 is mapped with 10 and 11 is used for a1. If two bit binary mapping is used for any gray level value of any chunk (Ch) e.g. X_{11}, X_{13} etc. then it will be represented by 11 and 10 respectively. Using the same way we can get 18 bit sequences for 9 gray values. Finally this will be placed in the remaining 18 indexes of V. up to this stage 32 bits have been recovered and 4 bit are still remaining.

Table 1: Mapping Bit for Average Value of Four Clusters in Each Chunk.

Average	Bit value for Mapping
a1	11
a2	10
a3	01
a4	00

3.1.3 Bit Generation for authentication

For detection of alteration in each chunk, we have used four bits to achieve accuracy and validity. Also we have tried to reduce the complexity. These four bits are categorized into two parts i.e. authentication bit one and authentication bit two. The details of these bits are as follows:

3.1.3.1 Authentication Bit One

First two bits of four bits are used for this authentication. Relation among all most significant bits of Px_i is shown as follows:

$$\sum_{i=1\ldots16}\left(\sum_{v=7,6\ldots4}(b_{iv}\oplus b_{iv-1})\quad mod2\right)\quad mod2 \qquad (1)$$

In the above formula i shows the internal operations that are repeated again and again for all 9 gray values for each chunk. Then summation is done on all 9 gray values and then modulo 2 is applied on the summation.

3.1.3.2 *Authentication Bit Two*

Rest two bits of four bits are used for generating a binary matrix of size (N/9) x (N/9) randomly with the help of a secret key (36 bits). For maintaining the accuracy in detecting the alteration, each bit is utilized for its corresponding chunk. After successful generation of four authentication bits for each pixel, they will be placed at last four indexes of vector V. Finally we get 36 bits having 32 bits as recovery bits and 4 bits as authentication bits. We create a matrix for all chunks of an image of size 9 x 4. Total 36 bits are used as a secret key.

3.1.4 Mapping of Chunk

Mapping of chunk is very significant because improper mapping may cause of loss of bit information. Thirty six bit information of a chunk cannot be simply placed into the same chunk. This may create a typical situation after any alteration in bit information. In this situation we will unable to recollect the necessary pixel values if it is found that the essential bit pattern containing main information of 36 bit information is lost.

Since we have N/9 number of chunk matrix represented as Mt_i. We have exchanged the content of matrix Mt_i for one chunk with other matrix Mt_j having the information of other chunk by using the secret key. This whole process is applied for each matrix Mt_i where i represent the number of matrixes i.e. i = [1, 2… 9]. Now 36 bits are required to be inserted on first 4 least significant bit (LSBs) positions of nine pixels of mapping chunk and this should be applied on all corresponding matrixes. Now we can get the gray level of pixels from [0…15] to [0…255].

Finally we will get the watermarked image after performing all required manipulation over matrix Mt_i.

3.2 Watermark Extraction

The watermark extraction technique has two parts i.e. detection of image alteration and recovery of image. Through our extraction technique we can find the place where it is tried to tamper the image and also restore the original image with its corresponding originality at the receiver end.

3.2.1 Detection of Alteration in Image

Through alteration an attacker tries to alter some pixel values without changing the image size. The altered image seems like original image. But detection technique helps to detect the tampered location. The alteration detection algorithm is mentioned below:

1) At the receiver side, the matrix of size (N/9) x (N/9) is generated pseudo randomly by applying the same secret key which was used at the time of embedding the watermark.

2) Then extract 36 bit stream using the same secret key from each chunk. Start matching of all 36 bits which was permuted at the embedding time to its corresponding chunk.

3) Matrix Mt_i of size 9 x 4 is rearranged for each chunk by applying the same secret key and obtain the same image as it was used as the time of embedding watermark.

4) Calculate the authentication bit one and two for each chunk as it is shown in section 3.1.3. Start comparison of the authentication of calculated authentication bit with the corresponding extracted authentication bits. If there any mismatch found in the comparison then label the chunk as altered one.

3.2.2 Recovery of Chunk

This is a method through which altered chunk is located and restored with proper imperceptibility. The algorithm of restoring the altered chunk is as follows:

1) To recover the chunk, get the altered chunk and extract first 8 rows form the corresponding matrix Mt_i. Create a row vector V.

2) The decimal value of first 4 bits from a vector V is the highest average value. Here this value is a4. Decimal value of d1, d2, d3, d4, and d5 having 2 bit binary mapping from table 1. Now we will get 14 bits from vector V. We can also calculate a3, a2 and a1 as follows.

a3 = a4-d1, a2 = a4-d2
d3 = a4-a1, a1 = a4-d3

3) Follow table 1 for replacing all two consecutive binary bits with their corresponding average value for each vector V.

4) Now we get 9 gray level values that have ranges from [0 …. 15].

5) At the end of each gray value append four 0s as first four least significant bits to make a complete range of 0 to 255.

6) Create a matrix Mj of size 3 x 3 from 9 gray values.

7) Start replacing the altered chunks by their corresponding Mj matrix.

Finally recovered image is obtained.

IV. Experimental Setup and Result Analysis

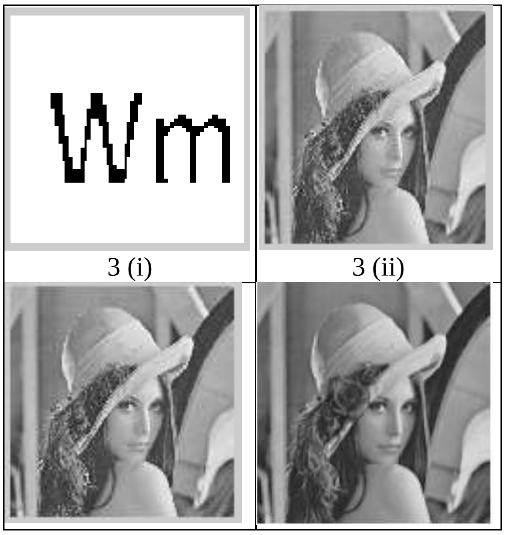
For this experiment, we have checked the performance of the proposed algorithm with other state of the art algorithms like DCT, DWT and SVD. In measuring the performance of each watermarking methods, we have considered the comparison between the received watermarked image and the original host image. For this comparison, Peak Signal to Noise Ratio (PSNR) [16] [17], Mean Square Error (MSE) [16] [17] and recovery time of extracting the original image is calculated for each algorithm to check the effectiveness of restoration process. The respective formula of PSNR and MSE are as follows.

$$PSNR= \quad 10log_{10}\frac{(MAX)^2}{MSE}$$

(2)

$$MSE=\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(I(i,j)-K(i,j))^2 \qquad (3)$$

The algorithm which has greater PSNR value, less MSE and less recovery time, that will be the best algorithm among all algorithms. Five types of attacks such as whitenoise, Gaussian filter, crop, rotate and no attack have been used to check the superiority of the algorithms. Result shows that the proposed algorithm gives better result on the basis of PSNR and MSE as compared to DCT, SVD. The recovery time obtained by proposed algorithm is better than DCT and SVD but comparable to DWT. Different images of dimension 256 x 256 have been used for examine the performance of the proposed algorithm and it is found that the proposed algorithm gives better result as compared to other state-of-the-art algorithms. But for showing the overall performance of the proposed one we have taken one image to show the overall working of the proposed algorithm, shown in figure 3. Figure 3, shows the different stages of image authentication and restoration process i.e. 3 (i) Shows the watermark image, 3 (ii) This is a cover image used for embedding watermark, 3 (iii) Shows the watermarked image (after embedding watermark), 3 (iv) This is the attacked image (altered image), 3 (v) This depicts the extracted watermark image.

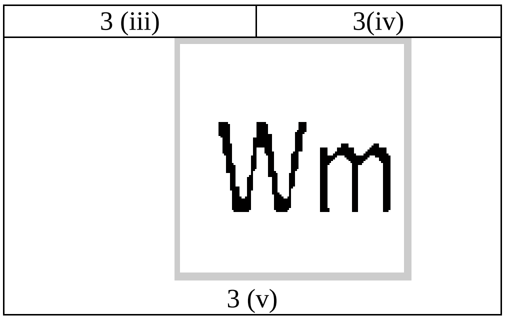


3 (i)

3 (ii)

3 (iii)

3(iv)



3 (v)

Figure 3: Shows different stages of watermarking of host image from embedding watermark to watermark extraction.

The experimental values of PSNR, MSE and recovery time by different algorithms are shown in table 2, 3 and 4 respectively. The results obtained by the proposed algorithm are better than other state of the art algorithms. These results obtained in the presence of different types of attacks which are mentioned earlier and also shown in the following tables 2, 3 and 4. In the following tables we have shown the best result found by all the algorithms for PSNR, MSE and recovery time. It is found that the proposed algorithm shows its superiority over other algorithms.

Table 2: PSNR of Different Algorithms on Different Attacks.

Attacks	Algorithms			
	SVD	Proposed	DCT	DWT
1—whitenoise	113.6751	109.4799	104.9039	2.9307
2--Gaussian filter	345.6166	784.9005	609.9850	371.5687
3—Crop	2.9332	2.9336	2.9266	2.9325
4—Rotate	2.9261	14.7468	14.2456	2.9273
5--No attack	2.9298	209.8763	157.0462	2.9317

Table 3 : MSE Of Different Algorithms on Different Attacks.

Attacks	Algorithms			
	SVD	Proposed	DCT	DWT
1—whitenoise	0.9968	0.8303	0.9133	1
2--Gaussian filter	0.9971	0.9890	0.9132	1
3—Crop	1	0.8631	1	1
4—Rotate	0.8631	0.0131	0.9167	0.8631
5--No attack	0.4902	1	1	0.4750

Table 4: Recovery Time of Different Algorithms on Different Attacks.

Attacks	Algorithms			
	SVD	Proposed	DCT	DWT
1—Whitenoise	5.1636	4.6644	5.0700	1.9032
2—Gaussian filter	5.6004	5.0388	5.2572	1.9500
3—Crop	15.0541	5.3820	17.4253	2.0592
4—Rotate	5.2104	4.8672	5.0232	2.1060
5--No attack	6.6300	4.4304	5.6004	1.7160

V. Conclusion and Future Work

In this paper, a new chunk-based fragile watermarking technique has been proposed for image authentication and restoration. The proposed technique works in spatial domain rather than other image restoration technique that works in spatial domain. Result shows the efficiency of proposed work in detecting the altered chunk and also restoring with enough accuracy from tempered regions with high imperceptibility. PAM, a object based clustering algorithm is used in making clusters of gray level values that helps in replacing other gray level values within a chunk. Comparison of the proposed method with other methods like DCT, DWT and SVD is done on the basis of PSNR, MSE and recovery time. It is found that the proposed one gives better result and shows its effectiveness and accuracy in image restoration.

Our future work is to optimize the performance of the proposed algorithm by adding some new factors that will be helpful to improve the accuracy and authenticity of the result. Also we will use this algorithm on different applications related to image authentication and restoration processes for checking the performance and getting the feedback. This feedback will be utilized for further improvement.

References

[1] Lin, Eugene T., and Edward J. Delp. "*A review of fragile image watermarks* ", Proceedings of the Multimedia and Security Workshop (ACM Multimedia'99) Multimedia Contents. 1999.

[2] Eugene T. Lin and Edward J. Delp, “A *review of fragile watermarking”,* Center for Education and Research in Information Assurance and Security, Purdue University, West Lafayette, IN 47907-2086.

[3] Jiri Fridrich and Miroslav Goljan, *“Images with Self-Correcting Capabilities”,* IEEE, 0-7803-5467-2/99, 1999.

[4] Mi-Ae Kim and Won-Hyung Lee*, “A Content-Based Fragile Watermarking Scheme for Image Authentication”,* Springer-VerlagBerlin Heidelberg, AWCC 2004, LNCS 3309, pp. 258-265, 2004.

[5] Shengbing CHE, Bin MA, Zuguo CHE, *“ An Adaptive and Fragile Image Watermarking Algorithm Based on Composite Chaotic Iterative Dynamic System”,* IEEE DOI 10.1109/IIH-MSP.2008.24,2008.

[6] Yusuk Lim, ChangshengXu, David Dagan Feng*, “Web based Image Authentication Using Invisible Fragile Watermark”,* Australian Computer Society, Inc. 2002.

[7] Hongjie He, JiashuZhang, Fan Chen*, “Block-wise Fragile Watermarking Scheme Based on Scramble Encryption”,* IEEE 978-1-4244-4105-1/07, 2007.

[8] Yong-Zhong He, Zhen Han, *“A Fragile Watermarking Scheme with Pixel-wise Alteration Localisation”*, IEEE, 978-1-4244-2179-4/08, 2008.

[9] Xinpeng Zhang and Shuozhong Wang, *“Fragile watermarking scheme using a hierarchical mechanism Elsevier”*,doi:10.1016/j.sigpro.2008.10.001.

[10] Ping Wah Wong and Nasir Memon, *“Secret and Public Key Image Watermarking Schemes for Image Authentication and Ownership Verification”,* IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 10, NO. 10, OCTOBER 2001.

[11] Hongjie He, JiaShu Zhang, and Heng-Ming Tai, *“A Wavelet-Based Fragile Watermarking Scheme for Secure Image Authentication”,* IWDW 2006 LNCS 4283, pp. 422432, 2006.

[12] Woo ChawSeng, Jiang Du, Binh Pham, *“Fragile Watermark with Self Authentication and Self Recovery”* Malaysian Journal of Computer Science, Vol. 22(1), 2009.

[13] Juan R. Hernández, Associate Member, IEEE, Martín Amado, and Fernando Pérez-González, Member, IEEE, *“DCT-Domain Watermarking Techniques for Still Images: Detector Performance Analysisand a New Structure”*, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 1, JANUARY 2000.

[14] Deepa Mathew K Karunya University Coimbatore, India, “*SVD based Image Watermarking Scheme”*, IJCA Special Issue on “Evolutionary Computation for Optimization Techniques” ECOT, 2010

[15] https://en.wikipedia.org/wiki/K-medoids.

[16] Shivani, S., Patel, A. K., Kamble, S., & Agarwal, S. (2011), *“Image Authentication and Restoration using Block-Wise Fragile Watermarking based on k-Medoids Clustering Approach”* In *2nd Int. Conf. Work. Emerg. Trends Technol* (pp. 44-51).

[17] Bisla, Nidi, and Prachi Chaudhary. *"Comparative Study of DWT and DWT-SVD Image Watermarking Techniques."* International Journal of Advanced Research in Computer Science and Software Engineering 3.6 (2013): 821-829.