# Image Authentication and Restoration using Block-Wise Fragile Watermarking based on k-Medoids Clustering Approach

Shivendra Shivani
CSED
Motilal Nehru National Institute of Technology, India

Sushila Kamble
CSED
Motilal Nehru National Institute of Technology, India

Anoop Kumar Patel
CSED
Motilal Nehru National Institute of Technology, India

Suneeta Agarwal
CSED
Motilal Nehru National Institute of Technology, India

## ABSTRACT

This paper proposes a block-wise fragile watermarking scheme based on k-medoids clustering approach. Proposed scheme is effective enough to reconstruct the tampered extensive content of an image. According to the suggested algorithm, first of all image is divided into the blocks and forty eight bits are calculated for each block, which consist of forty five Recovery bits and three Authentication bits namely Union bit, Affiliation bit and Spectrum bit. Authentication bits for a particular block are calculated by extracting five MSBs from each pixel of that block and apply them, on some predefined hash functions. Forty five Recovery bits are calculated using the means of derived clusters and its corresponding mapping bits. These Forty eight bits for each block are pseudo- randomly mapped with some other block using secret key. At the receiver end by comparing extracted and further calculated Authentication bits , one can be acquainted with the tampered block and extracted Recovery bits from mapping blocks gives adequate information to recover extensive content of host image.

## General Terms

Algorithms, Security, Watermarking

## Keywords

Block-wise fragile watermarking, Tamper detection, Image Recovery, Image Alteration.

## 1. INTRODUCTION

The simplicity in which digital data like image, audio, video can be altered, has produced a requirement for techniques that decide the integrity of digital information. Authentication and Recovery of tamper localization of a digital data are two critical requirements. Hence watermarking technique is widely used to protect these kinds of multimedia information. This technique is broadly categorized into three classes viz. fragile, semi fragile and robust watermarking. The concept of fragile watermarking came in existence because of ensuring the legitimacy and data integrity especially when it is utilized as an evidence of court or in medical diagnosis.

A fragile watermark is a mark that is readily altered or destroyed when the host image is modified through a linear or nonlinear transformation [4]. The fragile watermarking is used in image authentication because of its sensitivity against alteration. Fragile watermarking again can be classified into two category viz. Block wise fragile Watermarking [2][5][6][7][8][10][13][14] and Pixel wise fragile watermarking[9][11][12][15]. In block-wise fragile watermarking, host image is divided into small blocks and watermark information is derived from the vital content of block of the host image. In case of image alteration, the tampered block and watermark contained in that block will mismatch and by this inequality one can easily identify the tampered block. A block wise fragile watermarking proposed by *Hongjie, et al*. is standard technique which is based on scramble encryption [6]. According to their approach the watermark, derived from a block is randomly distributed on to the LSB of the whole image. This scheme is suitable to identify the tempered block but lacks image recovery. *J Fridrich and M Goljan* have also proposed the Block wise technique which is based on self embedding in [5]. They have proposed two methods to keep the image integrity. The first technique is based on quantization of block wise Discrete Cosine Transform (DCT) coefficient and represented as 64 or 128 bits. These bits are used to replace one or two Least Significant bit (LSB) of another block. In the second method a new image is produced by reducing gray level of the original image. These reduced gray levels are cyclic shifted and embedded in to pixel difference. If some modification is done on watermarked image then the quantized DCT coefficient and the new reduced gray level image can be used to reconstruct the principal content of the tampered area. Block-wise fragile watermarking has some limitations like within a particular block some pixels are really altered and some are not which is undesirable. Hence notion of pixel-wise fragile watermarking came into picture in which watermark information is derived from gray value of pixels and further embedded into image itself. Any alteration in gray value of pixel will be responsible for wrong value of watermark in further calculation at receiver side hence one can easily recognize altered pixel with high precision. A pixel wise fragile watermarking scheme is suggested by *Y Lim et al.* in [15]. In this technique seven Most

Significant Bit (MSB) of a gray value is given as an input to the hash function. Using a secret key and hash value we calculate a single value either 0 or 1 for each pixel and this value is embedded in the first LSB of corresponding pixel. Any change in gray value of pixel will return wrong hash value and altered pixel can be identified easily. *X Zhang and S Wang* have proposed a statistical watermarking technique for accurately localizing tampered pixel in [11] .They have calculated a set of tailor-made authentication data for every pixel with some additional test data and embedded into the host image. On the receiver side by examining the pixels and their respective authentication data, one can reveal the exact pattern of the content alteration. These all proposals are good enough to identify the exact position for tamper but lacks in restoration of alteration. Hence some more proposals are also there which are used to recover the image content. *X Zhang and S Wang* have proposed fragile watermarking with error free restoration capability [10]. This technique is based on tailor made watermark consisting of reference bits and check bits. These bits are embedded into the host image using lossless data hiding method. On the receiver side the check bits are calculated and compared with extracted check bits. This can detect the tampering of the image. After getting the alteration spot the reference bit extracted from other blocks are used to exactly recover the original image. Another self recovery watermarking technique is proposed by *Hong-Jie He et.al* in [7]. This proposed scheme embeds the encrypted feature comprising 6-bit recovery data and 2-bit key-based data of the image block into the LSB of its mapping block. The validity of a test block is determined by comparing the number of inconsistent blocks in the 3×3 block-neighborhood of the test block with that of its mapping block. The 3×3 block-neighborhood is also used to recover the tampered blocks whose feature hidden in another block is corrupted. A standard approach in transform domain is also given by *X. Zhao et.al* in [13]. In this paper the watermark bits are embedded into the middle frequency region of each block after applying Slant transform (SLT) of the host image. The host image is further compressed and then embedded into the LSBs of the watermarked image for subsequent self-restoration. The tampered regions of the watermarked image can be detected and localized by extracting the embedded watermark to compare with the original watermark for authentication. Localized tampered regions are self-recovered by extracting the LSBs of the watermarked image.

A detailed overview of our proposed algorithm is explained in section 2. Section 3 provides experimental results and their analysis. The proposed algorithm is concluded in section 4 followed by the references.

## 2. PROPOSED APPROACH

Proposed algorithm is based on k-Medoids clustering scheme which is representative object based technique. This algorithm is purely in spatial domain as illustrated in figure 1. Consider a gray scale host image I which has dimension m × n. Then N represents number of pixels $N = m \times n$. So the gray scale value at each pixel of the image is denoted by $P_i$ where $P_i$ Є (0..255), $i = 1,2,3.....N$. $P_i$ can be denoted by 8 bits. So each single bits of $P_i$ is denoted by $b(P_i, 7)$, $b(P_i, 6)$, $b(P_i\ , 5)$….. $b(P_i\ , 0)$. Then the individual bit of any pixel $P_i$ can be represented in binary form by following equation

$$\boldsymbol{b(P_i, u) = \left\lfloor \frac{P_i}{2^u} \right\rfloor \, mod\, 2 \text{ where } u = 0, 1, 2 \ldots 7} \qquad \mathbf{(1)}$$

## 2.1 Watermark Embedding

Watermark embedding process can be classified into four different phases viz. clustering of pixels, Recovery bit generation, Authentication bit generation and Block mapping.

### *2.1.1 Clustering of Pixels*

**Step1-** First of all remove first 3 LSBs of all pixels to reduce the gray scale value from [0,255] to [0, 31]. Now each $P_i$ will take 5 binary bits to represent it.

**Step2-** Divide the image I into number of blocks having dimension 4×4. Hence each block will contain 16 gray scale values. So the total number of blocks will be N/16.

**Step3-** Every block which contains 16 gray values will be inputted to k-medoids clustering algorithm to make three different clusters.

Suppose we have n dataset and we have to make k clusters then k-medoids algorithm is as follows.

(1) Randomly choose k gray values in a block as the initial representative seeds.
(2) repeat
(3) Assign every remaining gray value to the cluster with nearest representative seed.
(4) Arbitrarily choose a non representative gray value.
(5) Compute the total cost S of representative gray value with non representative gray value. Here cost is calculated by taking difference between both gray values.
(6) If S < 0 then swap the representative gray value with non representative gray value.
(7) Until no change.

**Step4-** Finally we get 3 clusters for each block now calculates the mean for each cluster as a round integer. Let means are $m_1$, $m_2$, $m_3$ then rearrange all mean $m_1$, $m_2$, and $m_3$ in descending order.

Here first phase of watermark embedding is completed and second is as follows

### *2.1.2 Recovery bit generation*

For each block we generate 45 recovery bits. These bits are formed as a vector V. Suppose three means for a block are $m_1$, $m_2$ and $m_3$ where descending order of mean is $m_3 > m_2 > m_1$.

.**Step5-** Map the mean values with their corresponding two bit pattern as shown in table.

**Step6-** Convert the highest mean that is $m_3$ here, into 5 bit binary form and put on first five indexes of vector V.

**Step7-** Now calculate the following

$$\boldsymbol{D_1 = m_3 - m_2}$$
$$\boldsymbol{D_2 = m_3 - m_1}$$
$$\boldsymbol{D_3 = D_2 - D_1}$$

**Step8-** Convert $D_1$ and $D_3$ into 4 bit binary form and put them into next eight indexes of vector V. Now we have occupied 13 position of vector V and we have to calculate 32 more bits for recovery.

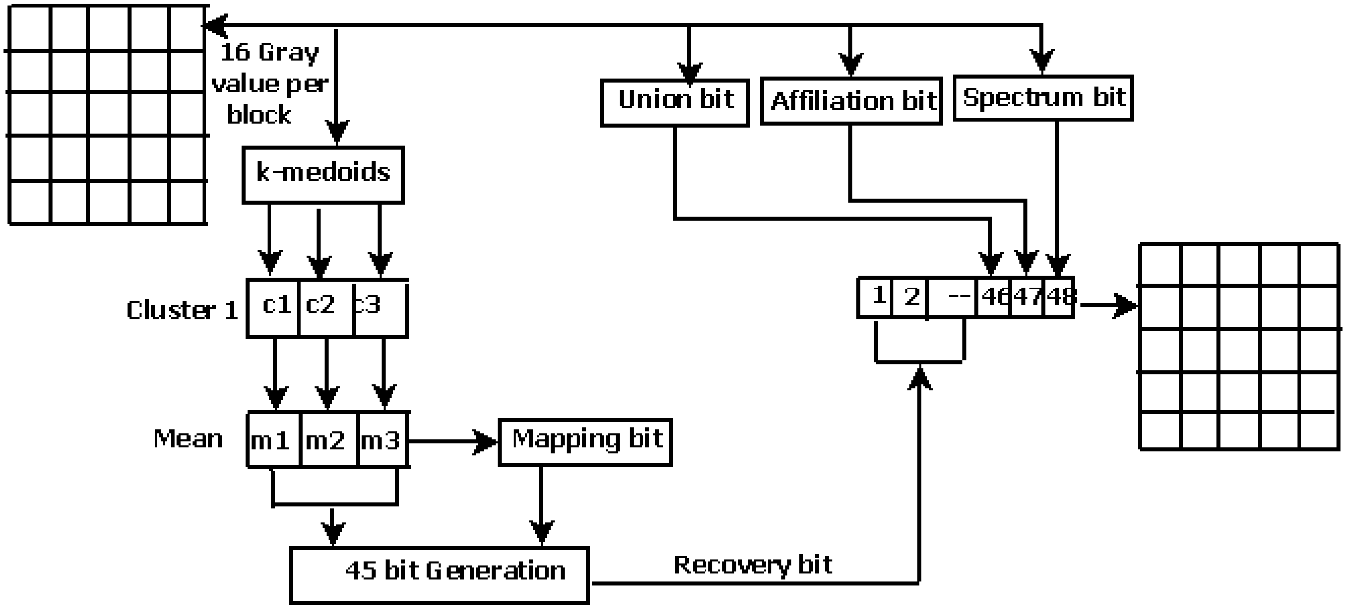


**Fig 1: Block diagram for watermark embedding.**

**Step9-** Map the all 16 gray level values with the two bit mapping binary bits for their corresponding mean values of those clusters in which they belong. For example

$$\mathbf{B} = \begin{array}{cccc} Y_{11} & Y_{12} & Y_{13} & Y_{14} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} \end{array}$$

Let B is one of the block having 16 gray level values and by using clustering algorithm we have calculated three clusters $C_1$, $C_2$, and $C_3$ which has following values

$\mathbf{C_1 = \{ Y_{11} , Y_{12} ,Y_{33} , Y_{34} , Y_{31} ,Y_{32}\}}$

$\mathbf{C_2 = \{ Y_{13} ,Y_{14} ,Y_{41} , Y_{42} \}}$

$\mathbf{C_3 = \{ Y_{43} , Y_{44} ,Y_{21} , Y_{22} , Y_{23} , Y_{24}\}}$

Mean of $C_1$, $C_2$, and $C_3$ are denoted by $m_1$, $m_2$, $m_3$ respectively and after arranging in descending order we get $m_3$, $m_2$, $m_1$. According to table 1, $m_3$ is mapped by 01 similarly $m_2$ is mapped by 10 and $m_1$ is mapped by 11. If we have to put two bit binary mapping bit for $Y_{11,}$ then it will be denoted by 11 because it belongs to the cluster $C_1$ and mapping bit for $C_1$ is 11. So by this way we will get 32 bit sequence for 16 gray values and it will be placed in next 32 indexes of vector V systematically. Now we have occupied 45 bits of V and we have 3 remaining bits.

### *2.1.3 Authentication bit generation*

For each block, we are using three bits for alteration detection for more accuracy. These three bits are called Union bit, Affiliation bit and Spectrum bit. These names depend on their generation procedure. Following operation is done for each block

#### *2.1.3.1 Union bit generation*

Consider $P_i$ is any pixel of block B. So we 5 MSBs of $P_i$ represented as $b_a$ where $a$ Є (3..7) for union bit calculation. Similarly $b^r_a$ and $b^c_a$ are binary value of corresponding row and column value in spatial image plane.

Now calculate the following

$$A^{s1} = \underset{a=7,6..3}{Ex\text{-}OR}(b^r_{a-3} , b_a) \qquad (2)$$

$$A^{s2} = \underset{a=7,6..3}{Ex\text{-}OR}(b^c_{a-3}, b_a) \qquad (3)$$

Where, $A^{s1}$ represents bitwise Ex-OR operation between row value and pixel value whereas $A^{s2}$ represents bitwise Ex-OR operation between column value and pixel value.

Hence Union bit calculation for a block will be followed as.

$$\boldsymbol{Union\ bit} = \sum_{i=1,.16} \Big( \sum_{j=1,2..5} (A^{s1}_{ij} \wedge A^{s2}_{ij})\ \boldsymbol{mod}\ 2 \Big)\ \boldsymbol{mod}\ 2 \quad (4)$$

#### *2.1.3.2 Affiliation bit generation*

The second LSB of $P_i$ is called as Affiliation bit. According to its name, it will show the relation among all MSBs of $P_i$. Calculation for this bit will be as follows

$$\boldsymbol{Aff\ bit} = \sum_{i=1...16} \Big( \sum_{v=7,6...4} (b_{iv} \oplus b_{iv-1} )\ \boldsymbol{mod}\ 2 \Big)\ \boldsymbol{mod}\ 2 \ (5)$$

Here variable i shows that internal operation will be repeated for all 16 gray values of a block and then modulo 2 will be done on the summation of all 16 output.

### *2.1.3.3 Spectrum bit generation*

For calculating spectrum bit a binary matrix of size $\frac{N}{16} \times \frac{N}{16}$ is generated pseudo-randomly with the help of a secret key. Each bit is used for its corresponding block.

**Step10-** After generating three authentications bits namely Union, Affiliation and Spectrum bit for each pixel we put them on last 3 indexes of vector V respectively.

Now we got 48 bits including forty five Recovery bits and three Authentication bits for each block as shown in figure 2, now we have to map 48 bit information each block into another location.

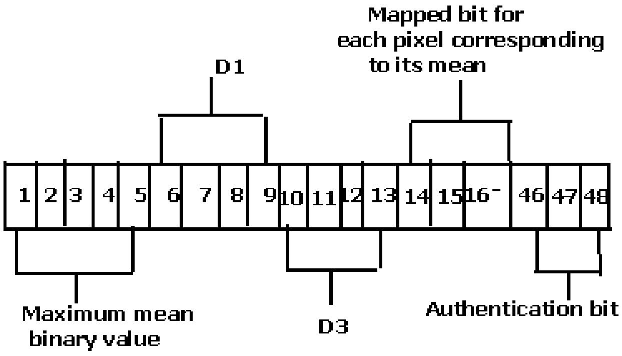

Mapped bit for each pixel corresponding to its mean
D1
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
46
47
48
Maximum mean binary value
D3
Authentication bit

**Fig 2: Forty eight bit vector with all position values.**

**Step11-** Create a matrix M for all blocks having size 16 x 3 using forty eight bits and permute it using a secret key.

### *2.1.4 Block mapping*

We cannot simply put forty eight bit information of one block into same block because, after any alteration we will not be able to recover the extensive pixel values if essential bit pattern from forty eight bit information which carries main information (till 13th ) is lost.

**Step12-** Since there will be N/16 number of block matrix which is named as $M_i$. Hence using secret key pseudo randomly exchanges the content of the matrix $M_i$ for one block with other matrix $M_j$, which has information of other block according to the figure 3. This step will be done for each matrix $M_i$ where i = [1, 2...16]. These forty eight bits must be inserted on the first three Least Significant Bit positions of sixteen pixels of mapping block, in one by one manner from the corresponding matrix M. Now the pixel's gray level range will be increased from [0...31] to [0...255].

After all manipulation of matrix $M_i$ whatever image we get that will be watermarked image. Assuming that the original distribution of 3 LSBs is uniform, the average energy of distribution caused by watermarking on each pixel can be calculated as

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left(I(i,j) - K(i,j)\right)^2 \qquad (6)$$

Where MSE is mean square value which is for $m \times n$ two monochrome image I and K in which one of the image is original host image and another one is watermarked image. Now the PSNR is defined as

$$\text{PSNR} = 10\log_{10}\frac{(MAX)^2}{MSE} \qquad (7)$$

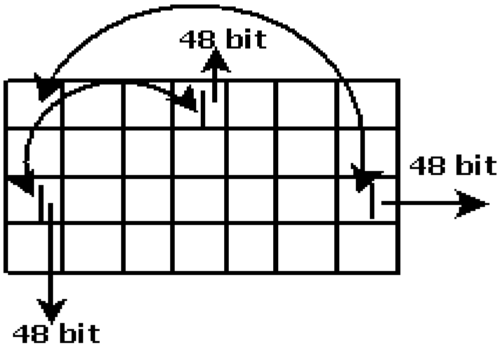


**Fig 3: Pseudo randomly permutation of Block's 48 bits.**

**Table 1. Mapping bit for mean value of 3 clusters in a block**

| Mean in Descending order | Two bit binary mapping |
|---|---|
| $m_3$ | 01 |
| $m_2$ | 10 |
| $m_1$ | 11 |

## 2.2 Effect of Tampering

Suppose an attacker alters the gray scale value of pixel $P_i$ which belongs to the block $B_j$ then we see that at what probability three alteration bits namely union, affiliation and spectrum will alter.

### *2.2.1 Effect on union bit*

Total number of possible alteration for any pixel $P_i$ due to 5 MSBs is

$$\alpha_1 = \sum_{k=1,2..5} {}^5C_k \qquad (8)$$

Since alteration in $P_i$ will not be detected by union bit if the $b^r_{(a-3)}$ and $b^c_{(a-3)}$ which are binary bits of corresponding row and column of altered pixels are even time similar. Hence the total number of set of MSBs which affects the union bit is

$$\boldsymbol{\alpha_2 = \sum_{1\le w \le k} \binom{k}{w} . 2^{5-k}} \qquad \textbf{(9)}$$

Hence the probability for altering the union bit is

$$\mathbf{Pr}(\boldsymbol{u}) = \frac{\boldsymbol{\alpha_2}}{\boldsymbol{\alpha_1}} \qquad \textbf{(10)}$$

Now the probability for detection of altered block using union bit is

$$\boldsymbol{Pr_b(u)} = \frac{\sum_{k=1,3,5..15} {}^{16}C_{k}}{\sum_{l=1,2,3..16} {}^{16}C_{l}} \times \mathbf{Pr}(\boldsymbol{u}) \qquad \textbf{(11)}$$

### *2.2.2 Effect on affiliation bit*

Alteration in $P_i$ will be detected by affiliation bit if and only if the altered pixel contains exactly one altered boundary bit of MSB (either 3rd or 7th). Hence the total number of pixels which actually affects the affiliation bit is

$$\boldsymbol{\beta = \sum_{k=1,2..5} {}^{5}C_{k} - \frac{3}{2}\sum_{l=1,2,,4} {}^{4}C_{l}} \qquad \textbf{(12)}$$

Therefore the probability of alteration in affiliation bit due to any change in $P_i$ is

$$\mathbf{Pr}\ (\boldsymbol{a}) = \frac{\boldsymbol{\beta}}{\boldsymbol{\alpha_1}} \qquad \textbf{(13)}$$

Now the probability for detection of altered block using affiliation bit is

$$\boldsymbol{Pr_b(a)} = \frac{\sum_{k=1,3,5..15} {}^{16}C_{k}}{\sum_{l=1,2,3..16} {}^{16}C_{l}} \times \mathbf{Pr}(\boldsymbol{a}) \qquad \textbf{(14)}$$

### *2.2.3 Effect on spectrum bit*

Since spectrum bit is not derived by MSBs of $P_i$ for a given block. Hence there will be no effect on the spectrum bit due to change in $P_i$. It will only be detected if there will be change in spectrum bit only. The significance of this bit is authentication verification of embedder and extractor.

## 2.3 Watermark Extraction

Proposed algorithm is using blind approach for image alteration detection and recovery. It means at the receiver end there will be only one altered image and on the basis of our extraction algorithm we find the tamper location as well as restore it with good imperceptibility.

### *2.3.1 Alteration Detection*

Suppose any attacker has altered some pixel values without changing image size. Then at receiver end it will be desirable to detect tampered location. Hence alteration detection algorithm is as follows.

**Step1-** First of all, receiver generates the pseudo random matrix of size $\frac{N}{16} \times \frac{N}{16}$ using same secret key which was used at the time of watermark embedding.

**Step2-** Extract forty eight bit stream from each block and using the secret key, match the forty eight bit stream with its corresponding block which was permuted at the time of embedding.

**Step3-** Rearrange the matrix $M_i$ of size 16 x 3 for each block using the secret key and make it, as it was the time of embedding.

**Step4-** Calculate the Union bit and Affiliation bit for each block by the help of equation 4 and 5. Now compare the calculated Union, Affiliation and spectrum bits with the corresponding extracted Union, Affiliation and Spectrum bit, if mismatch found then mark that block as altered one.

### *2.3.2 Block Recovery*

Once we locate the altered block we need to restore it in such a way so that proper imperceptibility is maintained. So for restoring the block, algorithm is as follows.

**Step5-** For each altered block**,** extract first fifteen rows from the corresponding matrix $M_i$ and convert it into the form of row vector V.

**Step6-** The decimal of the first five bits from the vector V is highest mean value, for above example it will be $m_3$ . Decimal of another four bit is $D_1$ and next four bits is $D_3$ . Now we calculate $m_2$ and $m_1$ using following way

$$\boldsymbol{m_2 = m_3 - D_1}$$

$$\boldsymbol{D_2 = D_3 - D_1}$$

$$\boldsymbol{m_1 = m_3 - D_2}$$

**Step7-** According to the table 1, replace all two consecutive binary bits from 14th to 45th position for each V, with their corresponding mean value. After that we get 16 gray level value which ranges from 0 to 31.

**Step8-** Now appends three 0s as first three LSB at the end of each gray value to make the range from 0 to 255. Then make a matrix $M_r$ of size 4 x 4 from that 16 value.

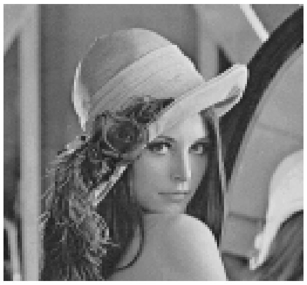
**Step9-** Replace the altered block by its corresponding $M_r$ matrix. This procedure will be done for all altered blocks.

Finally we get the recovered image and using equation 6 and 7 we can check the effectiveness of restoration.

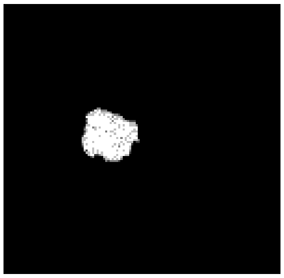
## 3. EXPERIMENTAL RESULTS

We have taken four test images as host image, all are having dimension 256×256 and each original host image is watermarked using our watermark embedding technique. Here we have considered four types of live applications for proposed algorithm as shown in figure 4. According to the given figures, series (a) of all figure shows watermarked image, series (b) shows the altered image, series (c) shows the alteration detection where white region shows the alteration detected block and series (d) is recovered image using extraction algorithm. First experiment is shown in figure 4 I(i) . Here we have taken object addition type attack where additional rose is added on the hat of lena shown in figure 4 I(b). Using proposed algorithm we have detected altered region shown 4 I(c) and recovered it also shown in 4 I(d). Second experiment is a very good application of suggested algorithm, sometime number plat of vehicle is a important evidence in court which may be subjected for alteration. Here in figure 4 II(a) and 4 II(b) number plat is altered from 16 to 18 but using this technique we have detected the tampered region as well as recovered it. Third experiment shows the object removal in which one person in given figure is removed shown in figure 4 III(a) and 4 III(b). Using proposed algorithm we have recovered the lost portion of the image shown in figure III (d). Forth type of attack is writing some undesirable text on image which is shown in figure 4 IV(a), 4 IV (b) but by the help of this scheme we have detected it as well as recovered it with high precision shown in figure 4 IV(c) and 4 IV (d) . Proposed algorithm is also tested on many images from standard image database and found a very satisfactory result.

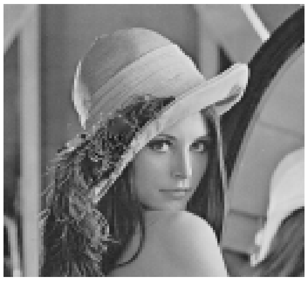
I (a)



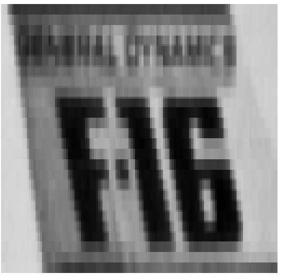
I (b)



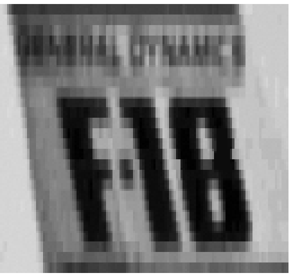
I (c)



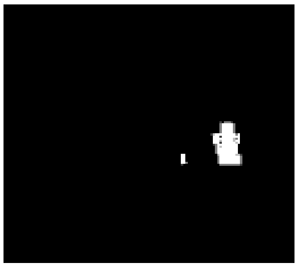
I (d)



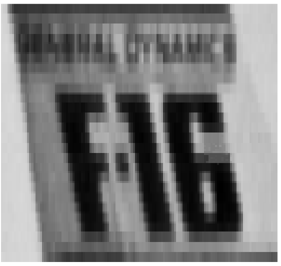
II (a)



II (b)

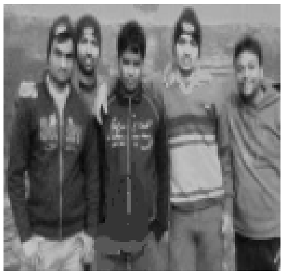


II (c)

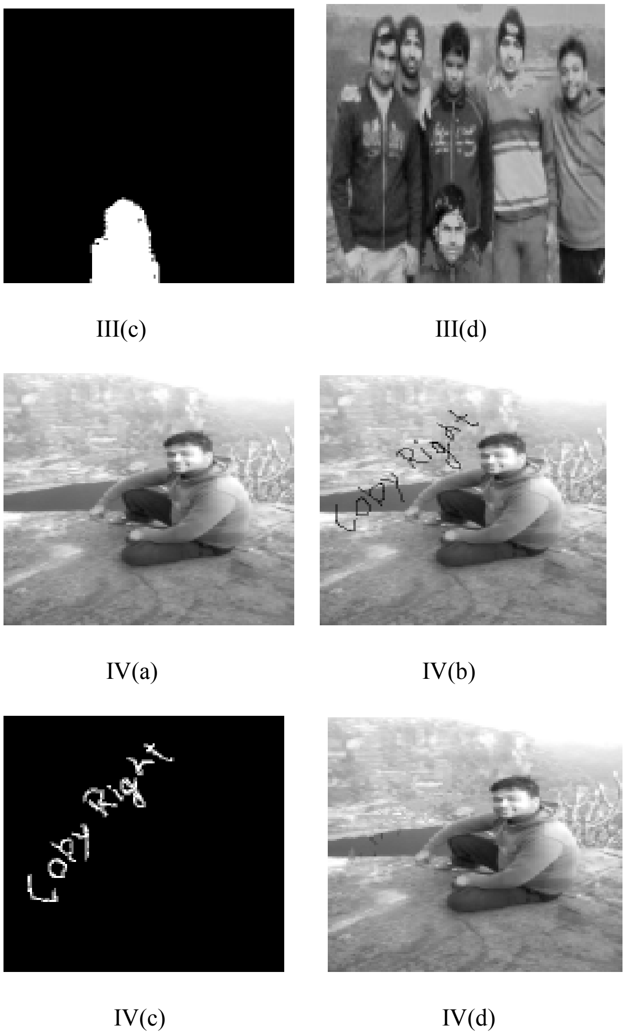


II (d)

III(a)



III(b)



III(c) III(d)

IV(a) IV(b)

IV(c) IV(d)

**Fig 4: Embedding, Alteration, Detection and recovery of watermarked image.**

Some vital information about the performance of algorithm is shown in Table 3, where Peak Signal to Noise Ratio (PSNR) of watermarked as well as recovered image is given.

**Table2 Essential information observed during watermark embedding and extraction**

| Image | PSNR<br>(Embedding) | Altered Block | Detected % | PSNR<br>(Recover) |
|---|---|---|---|---|
| Lena | 40.2 dB | 235 | 95% | 39.1 dB |
| Number Plate | 41.5 dB | 101 | 98% | 40.4 dB |
| Group | 39.7 dB | 356 | 96% | 38.2 dB |
| Person | 40.8 dB | 97 | 94% | 39.2 dB |

# 4. CONCLUSION

This paper proposes an image authentication and restoration approach using block-wise fragile watermarking which is based on k-medoids clustering technique. Many of the approaches for image restoration, proposed earlier, were in frequency domain but suggested scheme is utterly in spatial domain. Experimental results and table 2 show the efficiency of proposed scheme which is not only good enough to perceive the altered block with high accuracy but also able to restore those tampered blocks with good imperceptibility. K-medoids is a representative object based clustering technique which ensures that the gray level value which is used to replace other gray value within a block, belongs to the same block. Hence we take the benefit of this property of k-medoids. Clustering for each block is done by using a common characteristic hence for each element within a single cluster has same characteristics.

# 5. REFERENCES

[1] A. Criminisi, P. Perez, and K. Toyama. Region _llingand object removal by exemplar-based image inpainting. *IEEE Trans. Image Process*., vol. 13, no. 9, pp. 1200-1212, Sep. 2004.

[2] Anthony T. S. Ho, Xunzhan Zhu, Jun Shen, and PinaMarziliano. Fragile Watermarking Based on Encoding of the Zeros of the -Transform *IEEE Transactions On Information Forensic and Security,* VOL. 3, NO. 3, September 2008

[3] Anthony T. S. Ho, Xunzhan Zhu,lilian h. tang,Digital watermarking authentication and restoration for Chinese calligraphy image,*IEEE*, 1-4244-0882-2/07,2007

[4] 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.

[5] Jiri Fridrich and Miroslav Goljan. Images with Self-Correcting Capabilities. *IEEE*, 0-7803-546 2/99,1999.

[6] Hongjie He, Jiashu Zhang , Fan Chen. Block-wise Fragile Watermarking Scheme Based on Scramble Encryption. *IEEE* 978-1-4244-4105-1/07,2007.

[7] Hong-Jie He, Jia-Shu Zhang, and Heng-Ming Tai, Self-recovery Fragile Watermarking Using Block-Neighborhood Tampering Characterization, *Springer-Verlag Berlin Heidelberg*, IH 2009, LNCS 5806, pp. 132–145, 2009

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

[9] 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

[10] Xinpeng Zhang and Shuozhong Wang. Fragile Watermarking With Error-Free Restoration Capability, *IEEE transaction of multimedia*, VOL. 10,

NO. 8, December 2008.

[11] Xinpeng Zhang and Shuozhong Wang. Statistical Fragile Watermarking Capable of Locating Individual Tampered Pixels, *IEEE Signal processing letters*, VOL. 14, NO.10, October 2007.

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

[13] X. Zhao, A.T.S. Ho, H. Treharne, V. Pankajakshan, C. Culnane and W. Jiang, A Novel Semi-Fragile Image Watermarking, Authentication and Self-restoration Technique Using the Slant Transform, *University of Surrey Guildford*-GU2 7XH, UK

[14] Xinpeng Zhang and Shuozhong Wang, Fragile Watermarking Scheme with Extensive Content Restoration Capability, *Springer-Verlag Berlin Heidelberg*, IWDW 2009, LNCS 5703, pp. 268–278, 2009.

[15] Yusuk Lim, Changsheng Xu, David Dagan Feng. Webbased Image Authentication Using Invisible Fragile Watermark, *Australian Computer Society*, Inc. 2002.

[16] Yong-Zhong He, Zhen Han, A Fragile WatermarkingScheme with Pixel-wise Alteration Localisation *IEEE*, 978-1-4244-2179-4/08, 2008.