COPY – MOVE IMAGE FORGERY DETECTION IN A PARALLEL ENVIRONMENT

M.Sridevi, C.Mala and S.Sandeep

Department of Computer Science & Engineering,

National Institute of Technology, Tiruchirappalli, India
msridevi@nitt.edu
mala@nitt.edu
king.sandeep@gmail.com

***ABSTRACT*:**

Image forgery is the manipulation of digital images to conceal meaningful information or objects in the image. Among different image forgery techniques, copy – move forgery is one of the frequently used passive image forgery approach. The existing methods such as Principle Component Analysis (PCA), Discrete Wavelet Transform (DWT) & Singular Value Decomposition (SVD) are time consuming. Hence it is not suited for digital forensic science, surveillance system applications which uses image, video or multimedia security. This paper proposes a parallel algorithm for the copy – move image forgery detection to decrease execution time of the algorithm. The method uses overlapping blocks and lexicographical sorting in a parallel manner. The simulation results show that the proposed parallel version detects the forged region faster, so that it is best suited for real time applications.

KEYWORDS:

Copy-move detection, Image forgery, Block matching, Parallel environment, Complexity.

1. INTRODUCTION

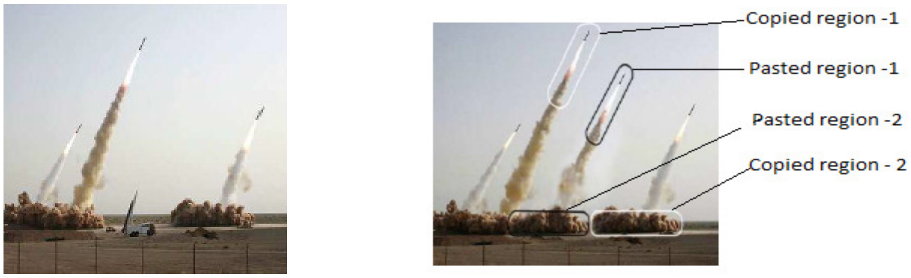
An image is generally accepted as a proof of occurrence of a depicted event. With computer's becoming more prevalent in every field, accepting digital image as an official document has become a common practice. The availability of low-cost hardware and software tools makes it easy to create, alter, and manipulate digital images. It has become difficult or impossible to trace these operations. As a result, the integrity and authenticity of digital images are lost. Currently there are few methodologies available to verify the authenticity and integrity of digital images in an automatic manner. Image forgery detection system is needed in many fields for protecting copyright and preventing forgery or alteration of images with malicious intentions. It is applied in areas such as journalism, scientific publications, digital forensic science, multimedia security, surveillance systems etc. The digital image forgery detection techniques are classified into active [2][3][7] and passive approaches [1][4][5][6]. In active approach, the digital image requires pre-processing of image such as watermark embedding or signature generation, which would limit their application in practice. Unlike the watermark and signature-based methods, the passive technology does not need any digital signature to be generated or to embed any watermark in advance. Block matching techniques are employed in passive technology for detecting the forged

Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 19–29, 2012.
DOI : 10.5121/csit.2012.2303

regions. There are three techniques widely used to manipulate digital images. They are

1) Copy-Move: A part of the image is copied and pasted into another part of the same image as shown in Figure 1.
2) Tampering: Tampering means manipulation of an image to achieve a specific result.

3) Splicing: A form of photographic manipulation in which there is digital splicing of two or more images into a single composite image.



a) Original image

b) Forged image

Figure 1. Copy-move forgery example

In Copy-Move image forgery, a part of the image is copied and pasted into another part of the same image. Copy-paste is a typical synthesis method to insert or hide some meaningful objects or information in an image. The authenticity of the image is doubted when duplication of objects or regions occurs in the image. Since the copied segments come from the same image, its properties will be compatible with the rest of the image, thus it is very difficult for a human eye to detect such type of forgery.

The copy-move image forgery detection techniques are used to detect the duplicated image regions. These regions may not be the forged regions and may not be an exact duplicate. The identical regions may be malfunctions by using retouching tools or additive noise or compression to get the resulting forged image. These techniques find profound use in forgery detection and security related aspects.

2. Related Works

Digital image forgeries can be detected using Chromatic Aberration[11], by exploiting Sensor Pattern Noise. Digital image forgeries can be exposed by analyzing Color Filter Array Interpolation. A variety of principles of Optical Physics like lighting inconsistencies are applied to establish the state of an image. Most recently, Fourier Mellin Transform (FMT) [5] and 1-D projection of log-polar values were applied to make the image forgery detection a more robust scheme.

One approach to detect copy-move forgery detection, proposed by Fridrich et al. [5], essentially performs an exhaustive search by comparing the image to every cyclic-shifted versions of it. Since this approach requires $(MN)^2$ steps for an image sized M ×N, it is difficult to use in practical.

Popescu et al [4] proposed a copy-move image forgery detection algorithm using block matching approach and Principal Component Analysis (PCA). Each block is represented as 16x16 and coefficients in each block are vectorized and the corresponding covariance matrix is constructed. A new linear basis is obtained by finding the eigenvectors of the covariance matrix. The duplicated regions are detected by performing lexicographically sorting to the image blocks. The algorithm proposed by Popescu et al [4] has $O(32k \log k)$ complexity where k is number of overlapping blocks.

Li et al [8] reduced the dimension of the image by considering only the low frequency subband of Discrete Wavelet Transform (DWT) output and further the length of the feature vectors are reduced using Singular Value Decomposition, (SVD) [8][9][10]. The complexity of the algorithm proposed by Li et al takes $O(8k \log k)$.

Another approach for detecting copy-move forgeries is the block-matching [1][4] procedure, which first divides the entire image into overlapping blocks. The overlapping blocks [1][6] are lexicographically sorted and similar blocks are identified as forged regions. The detection probability varies based upon the feature extraction method.

This paper proposes a parallel version of copy - move image forgery detection using block matching technique. It is seen from the performance analysis that the proposed parallel version of block matching algorithm decreases the execution time of it. As the parallel version performs the task faster, it is very well suited for real time applications. The sequential and parallel block matching techniques were experimented and their results are compared. The proposed method provides best results and better reduction in execution time.

The rest of the paper is organized as follows: Section 3 explains the proposed parallel copy – move image forgery detection using block matching technique. Section 4 presents the simulation results of the algorithm and performance analysis of the sequential and parallel algorithms. Section 5 concludes the paper.

3. PROPOSED PARALLEL COPY-MOVE IMAGE FORGERY DETECTION (PCMIFD) METHOD

This section proposes parallel algorithm for the following two operations required for detection of digital image forgery.

1) Overlapping blocks

2) Sorting

3.1 Parallelizing Overlapping Blocks:

Copy move image forgery approach detects duplicated connected image blocks. The distance between every duplicated block pair will be the same because each block is moved by the same amount. The input image is divided into many overlapping blocks by moving the block throughout the entire image starting from top left corner of the image. Consider an image, I of dimensions r x c. It is divided into (r-b+1) x (c-b+1) overlapping blocks of size b x b, where b is the specified block size (even numbered squared matrix). Vectors of dimension (b^2+2) are used to represent the blocks, where the last 2 vector locations are used to store block location (x,y) of the image [top left corner of the block]. A matrix is constructed using vectors of blocks as shown in Figure 2. The parallel algorithm for creation of block vectors and the array H is given in algorithm 1. It can use maximum of (r – b +1) processors where i_{th} processor constructs i_{th} vector of array H as shown in Figure 2.

Notation used:

b : Block Size

r x c : Image Size

H: Array of block vectors

I: Image intensity array

P : Processor;

y : Number of processors where y < (r-b+1)

Algorithm 1: To Create block vectors and array.

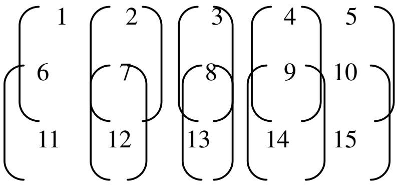
```
1. Assume $P_i$ processors where i=[0…y-1]

2. Initialize all the variables (k, l, q) to 0 and j=[ i* (r-b+1)/ y ] for all processors

3. for j =  [i * (r-b+1) / y]  to  [(i+1) * (r-b+1/ y)-1]  doing in parallel

    a) while ( k != (c - b+1) ) do
    b)  for l=0 to (b-1) do
    c)   for q=0 to (b-1) do
              H[ j x ( c – b+1) +k][ l x b + q]= I[j + l][k + q] // Construction of array H

    // Adding location for the vectors
        H[ j x ( c - b+1) + k][b x b]= j
        H[ j x (c – b+1) + k][b x b +1]=k
```



i) Sliding of block 2x2 into an original image 3x5

Vectors (b^2 =4)				Block location (x,y)	Processor No.
1	6	2	7	(1,1)	P_1
2	7	3	8	(1,2)	P_1
3	8	4	9	(1,3)	P_1
4	9	5	10	(1,4)	P_1
6	11	7	12	(2,1)	P_2
7	12	8	13	(2,2)	P_2
8	13	9	14	(2,3)	P_2
9	14	10	15	(2,4)	P_2

(ii) Matrix (8 x 4) Representation

Figure 2. Representation of overlapping blocks

3.2 Parallel sorting algorithm:

The matrix constructed in Section 3.1 is then lexicographically sorted using radix sort in a parallel manner. The similar information will be close to each other in the sorted list and hence the identical regions can be easily identified. The complexity of radix sort for each pass over *k* numbers (b^2) is O(256+*k*). The parallel sort algorithm is given in algorithm 2.

Algorithm 2: Lexicographical Sorting

```
    for $i \leftarrow 1$ to $d$ do (in parallel)  ;  where d is digit
        use a Radix sort to sort array H on digit $i$
    (Since each digit (pixel value) is in the range 0 to 255, counting sort is chosen as the stable
sort)
    end
```

The steps involved in proposed Parallel Copy- Move Image Forgery Detection (PCMIFD) system is shown in Figure 3.

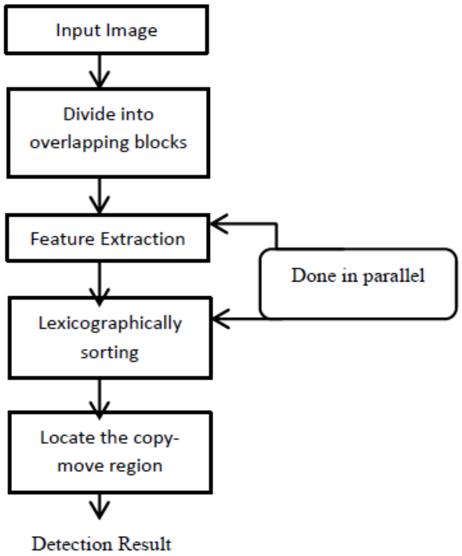

Input Image
Divide into overlapping blocks
Feature Extraction
Done in parallel
Lexicographically sorting
Locate the copy-move region
Detection Result

Figure 3: Flow diagram of Parallel Copy-Move image forgery detection system

The proposed PCMIFD parallelize both matrix construction and sorting operations. The intensity features are extracted from the input image by making use of overlapping blocks. And a matrix is created based on the extracted features. The lexicographical sorting is applied to the matrix to find the duplicated region. The duplicated regions are adjacent in the sorted list. Because, they shares common properties such as colour, intensity, texture, etc. The duplicated region may or may not be the forged region. The forged region can be distinguished from the original region by calculating correlation between them. The algorithm for the PCMIFD method is given in Algorithm 3.

Algorithm 3: PCMIFD method

Consider an *rX c* forged grayscale / color image (I) where r is the width of the image, c is the height of the image

1. Slide a *b X b* window over image I, where b is the block size.
2. Extract the pixel intensity information of each block, which gives k1=(r-b+1)*(c-b+1) blocks
3. Every block is represented as a vector. A matrix (H) is made with k1 x b^2 size, which contains all the block vectors and their top left corner position. The construction of vectors and array is given in algorithm 1.
4. The rows of the list, *H* are lexicographically sorted, (refer Algorithm 2) in parallel manner. Hence, similar rows, probably as a result of duplicated blocks, become adjacent to each other.
5. Blocks which are identical are mapped on the image using the location.

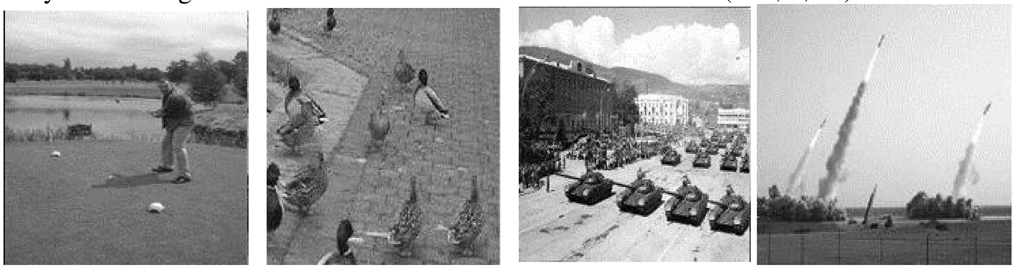
4. EXPERIMENTAL RESULTS & PERFORMANCE ANALYSIS

The proposed method was implemented in a parallel environment using Java threads. Almost 100 benchmark forged images were considered for testing the algorithms and few of the tested images are shown in Figure 4 a) – d). The tested results are shown in Figure 5.2. Two sets of simulation were conducted and the results were analysed with respect to its execution times. The two sets of simulation are given below:

1) In the first set of simulation, Copy – move image forgeries were detected using sequential lexicographical sorting.

2) The proposed method was used to test the copy – move image forgeries in parallel environment using java threads.

The simulation results and performance of the both schemes (sequential and parallel) were analyzed. The algorithms were also tested for various block sizes (b=2, 8, 16).



a) Image1 (Img1) b) Image2 (Img2) c) Image3 (Img3) d) Image4 (Img4)

Figure 4. Sample Images for testing

Figure 5.1 Tested image samples

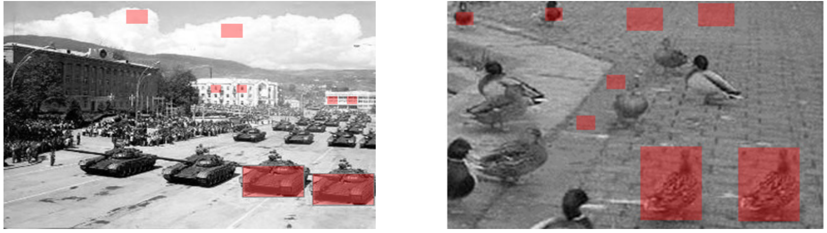


Figure 5.2 Output images (Copy – pasted regions are shown in red color)

Case 1: (Sequential copy – move image forgery detection)

The sequential implementation was executed for various images shown in Figure 4 a) – d) by selecting different block sizes. The execution time of the algorithm for block size (2 x 2 , 4 x 4, 8 x 8) were noted for the four different images and a graph was plotted as shown in Figure 6.1 and 6.2.

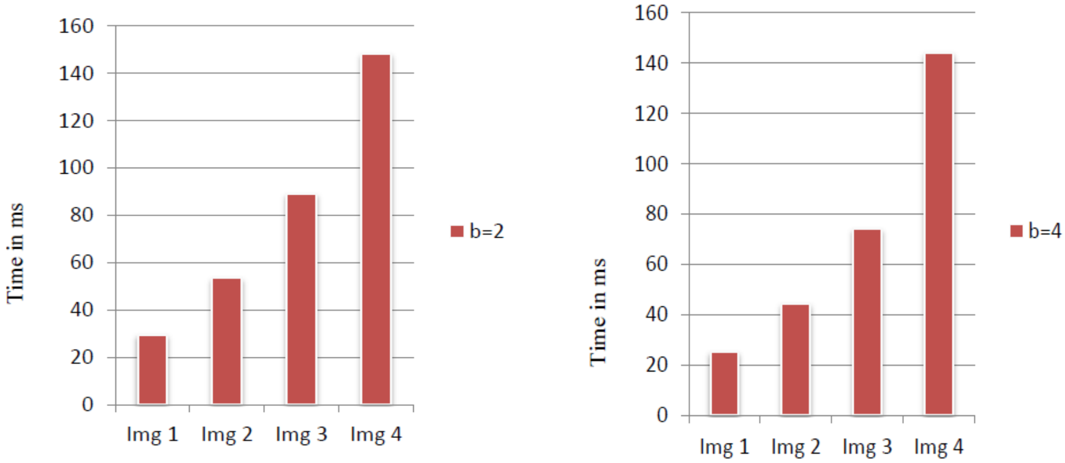


Figure 6.1 Sequential execution of the algorithm for block sizes, b=2 and b=4

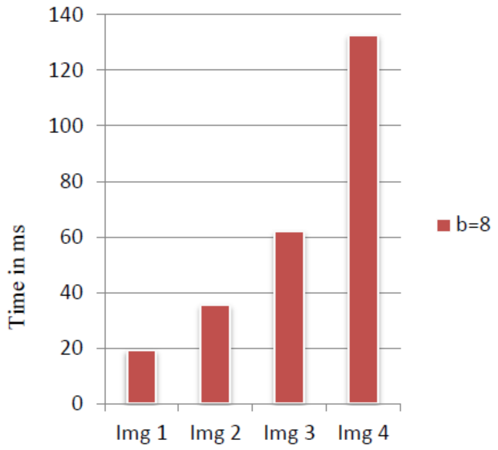


Figure 6.2 Sequential execution of the algorithm for block size, b=8

Case 2: (Parallel copy – move image forgery detection)

The different set of images as given in Figure 4 a) – d) was processed with the parallel algorithm given in algorithm 3. The execution time of the method was plotted for the various block size as shown in Figure 7.1 and 7.2.

It is inferred from the graph (Figure 6 and 7), that the computational time decreased with increase in block sizes. In the algorithm, it is stated that as block size increases, fewer vectors need to be sorted, resulting in performance improvement.

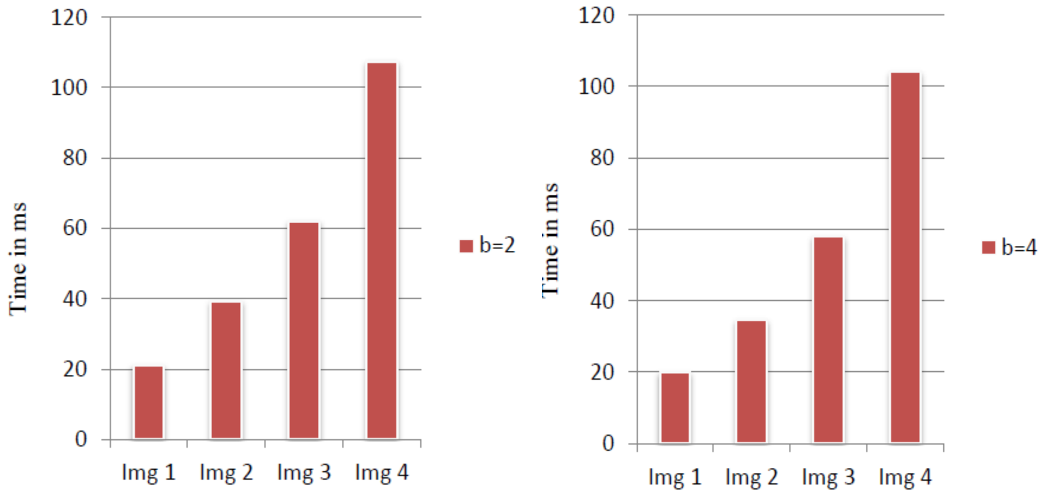


Figure 7.1 Parallel execution of the algorithm for block sizes, b=2 and b=4

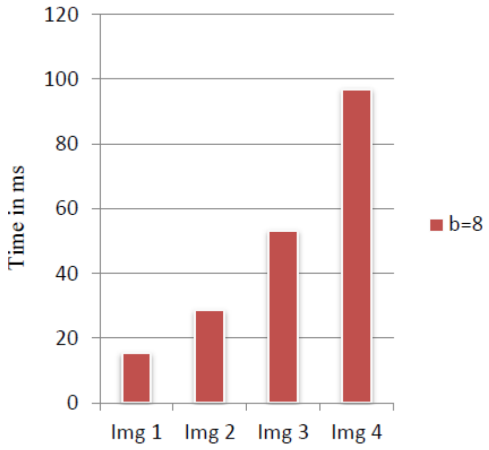


Figure 7.2 Parallel execution of the algorithm for block size, b=8

The sequential and parallel implementations of the algorithm were executed by keeping constant block size (2 x 2) for all four images in Figure 4. The execution times of both sequential and parallel methods for various images are plotted in Figure 8. It is inferred from the graph that the parallel implementation of the algorithm took lesser time than the sequential algorithm. There is a considerable performance gain achieved by the proposed method.

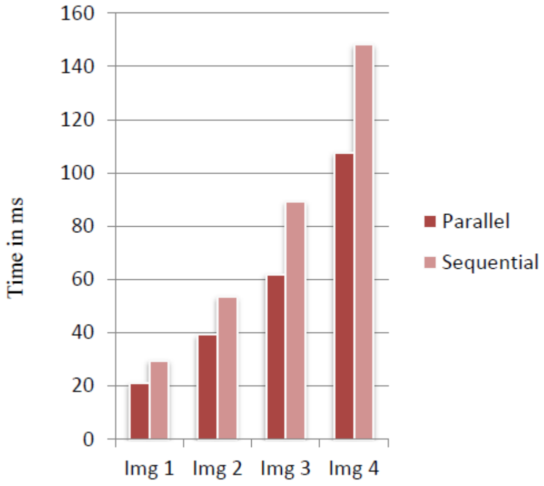


Figure 8. Performance comparison based on mode of execution

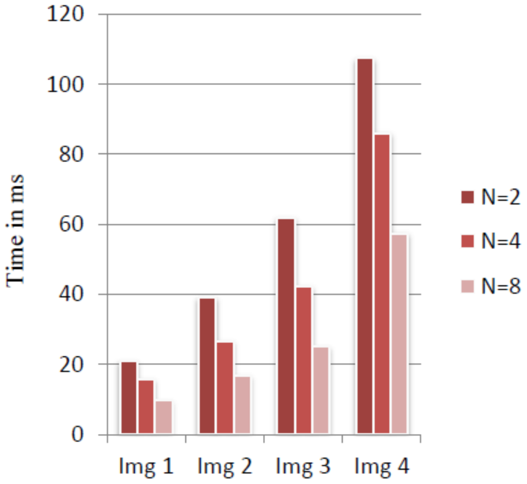


Figure 9. Performance comparison based on number of processors

The algorithm 3 was implemented for different number of processors (say N = 2, 4 and 8) and a graph was plotted for parallel version and it was shown in Figure 9. It is inferred from the graph, a noticeable performance improvement was observed when number of processors are increased.

The accurate detection of digital image forgery depends on the block size and the copied region. The selection of smaller block size yields false detection whereas larger block size leads to failure in detection. So, proper block size has to be chosen for detecting the duplicated regions in the forged image. The false forgery detection rate for various block sizes, b=2, b=4 and b=8 are compared in Table 1 for various images of Figure 4.

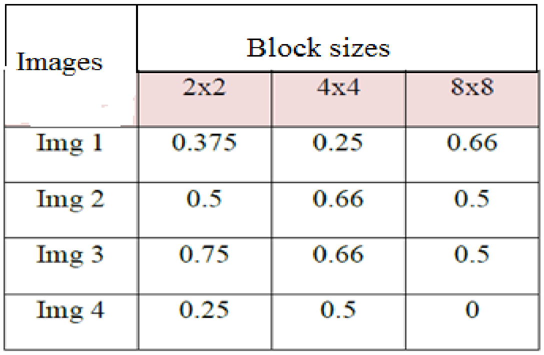
$$FDR = 1 - \frac{NoTD}{TNoD}$$

where FDR – False Detection Rate,

NoTD – Number of true detection

TNoD – Total Number of detection

Table 1. False detection rate



Images	Block sizes		
	2x2	4x4	8x8
Img 1	0.375	0.25	0.66
Img 2	0.5	0.66	0.5
Img 3	0.75	0.66	0.5
Img 4	0.25	0.5	0

5. CONCLUSION

This paper analysed copy-move image forgery techniques and proposed a parallel block matching algorithm to detect the forged regions, if copy and paste are done in the digital image. The simulation results show that the proposed parallel algorithms reduce the execution time. The false detection rate enables us to decide correct block size for accurate detection. Usage of the parallel environment has drastically reduced time complexity of the algorithm.

REFERENCES

[1] Michael Zimba and Sun Xingming,"DWT-PCA (EVD) Based Copy-move Image Forgery Detection" in *International Journal of Digital Content Technology and its Applications*, Vol. 5, N0.1, January 2011,pp.251-258.

[2] P. Meerwald and A. Uhl, "A Survey of Wavelet-Domain Watermarking Algorithms," in *Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents*, Vol. 4314, 2001, pp. 505-516.

[3] F. Hartung and M. Kutter, "Multimedia Watermarking Techniques," in *Proceedings of the IEEE*, Vol. 87, No. 7, July 1999, pp. 1079-1107.

[4] A. C. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Duplicated Image Regions," Technical Report, TR2004-515, Department of Computer Science, Dartmouth College, 2004.

[5] J. Fridrich, D. Soukal, and J. Lukas, "Detection of Copy-Move Forgery in Digital Images," in *Proceedings of Digital Forensic Research Workshop*, August 2003.

[6] S. Bayram, T. Sencar, N Memon" An Efficient and Robust Method for Detecting copy move Forgery", ICASSP 2009, pp. 1053-1056.

[7] C. T. Hsieh and Y. K. Wu, "Geometric Invariant Semi-fragile Image Watermarking Using Real Symmetric Matrix," *WSEAS Transaction on Signal Processing*, Vol. 2, Issue 5, May 2006, pp. 612-618.

[8] G. Li, Q. Wu, D. Tu, and S. Sun, "A Sorted Neighborhood Approach for Detecting Duplicated Regions in Image Forgeries based on DWT and SVD," in *Proceedings of IEEE International Conference on Multimedia and Expo*, Beijing China, July 2-5, 2007, pp. 1750-1753.

[9] Yingkun Hou, Chunxia Zhao, Yong Cheng, Zhengli Zhu, "Image Watermarking Resynchronization to Geometric Attacks in DWT Domain", JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 4, No. 4, pp. 88 ~ 98, 2010

[10] Chi-Man Pun, Moon-Chuen Lee, Cong Lin, "Geometric Invariant Shape Representation Based on Radon and Adaptive Stationary Wavelet Transforms",*JCIT:Journal of Convergence Information Technology*, Vol. 5, No. 6, pp. 54 ~ 65, 2010

[11] A. C. Popescu and H. Farid, "Exposing Digital Forgeries in Color Filter Array Interpolated Images," *IEEE Transactions on Signal Processing*, Vol. 53, 2005, pp. 3948–3959.