Copy Move Image Forgery Detection Based on Polar Fourier Representation

Yitian Wang and Sei-ichiro Kamata

***Abstract*—With the rapid development of multimedia technology, it's easy for someone to obtain an image and edit it according to their own preferences or some ulterior purpose. Copy–Move is a common type of digital image forgery where a part of the original image is copied and pasted at another position in the same image. In this paper, we propose an efficient methodology for enhancing block matching based on Copy–Move forgery detection. The main contribution of this work is the utilization of polar representation to get the representative features for each block. The main feature used in this paper is the frequency of each block based on Fourier transform. The experimental results show the efficiency of the proposed method for detecting copy-move regions, even when the copied region has undergone severe image manipulations such as rotation, scaling, Gaussian blurring, brightness modification, JPEG compression and noise addition.**

***Index Terms*—Copy move, forgery detection, Fourier transform, polar coordinate system.**

I. Introduction

A copy-move forgery is created by copying and pasting content within the same image as shown in Fig. 1. It can be a crucial task in image forensics to prevent people from being cheated by unauthentic images, which may influence some important judgement such as in a court of law. It is necessary to verify the authenticity of the images. The techniques of image forgery detection can be categorized into two methods: block matching and key point matching.

Copy move forgery detection (CMFD) is used to judge one image if it is forged by copy move operation and find copy move regions. Sometimes the copy-moved regions are not easy to find by human eyes, but we can find them automatically. The rest of our paper consists of the following contents:

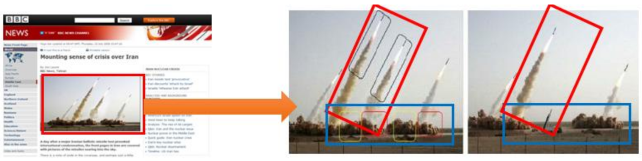

Fig. 1. An example of forged image.

First we introduce some previous work that are related to copy move detection methods including block-based methods which we focus on, key point based methods where we have done some experiments to compare with and have provided common processing pipeline which combines the two kinds of methods together. Then we propose a polar Fourier representation using radix sort. The preprocessing work will be discussed in the first place before going to explain the polar system conversion. The remaining parts include the description of Fourier transformation, feature extraction and matching procedure. Finally we provide the experimental results of proposed algorithm. After introducing the experimental environment, the results of CM detection are discussed in comparison with the existing methods using the benchmark datasets. At last we provide the conclusions and future work to be solved.

II. Related Work

Related work include block matching (BM) method, key point based methods and other combined methods. The main objective of CMFD is to determine whether an image contains copied regions. However, from the recent research we know that it is computationally impossible to examine every possible pair of regions with different shapes and sizes. What we can do is to try to improve the accuracy of the forgery detection.

The BM method is the most commonly used methodology for matching purpose, where an image is divided into fixed-sized overlapping blocks and the blocks are processed instead of individual pixels. For the BM forgery revealing method, region based representative features (RF) have to be extracted for each block before the matching or the classification step. Some existing methods are based on lexicographic sorting of quantized Discrete Cosine Transform (DCT) coefficients of image blocks [1]. However this system cannot detect duplicated regions with a rotation of more than 5 °. Bayram *et al.* use Fourier-Mellin transform (FMT) in this field. They propose frequency domain features from image blocks using the FMT [2]. Wu *et al.* also propose a log-polar Fourier (LPF) transform-based algorithm [3]. Our proposed method is inspired by the polar system which can effectively find copied regions with rotation modification. The Fourier transform can reduce much redundant information from the image.

Lynch *et al.* [4] utilize the average intensity of each block as the RF. The authors propose an Expanding Blocking (EB) scheme to speed up the matching process, as well as to evaluate their system accuracy under JPEG compression and blurring effects. Popescu *et al.* [5] use a principal component analysis (PCA) on image blocks, but the detection accuracy is not sufficient. However the above methods fail to detect forgery under any type of geometric transformations such as rotation, scaling, etc. Bayram *et al.* [6] discuss the rotation

Manuscript received January 13, 2018; revised April 7, 2018. This work was partially supported by JSPS KAKENHI Grant Number 15K00248.

The authors are with Graduate School of Information, Production and Systems of Waseda university, Kitakyushu, Japan (e-mail: yzwang@fuji.waseda.jp, kam@waseda.jp).

doi: 10.18178/ijmlc.2018.8.2.680

modification and propose a copy-rotate-move forgery detection based on spatial domain (CRMS). The blocks are divided into nested frames, and the features are extracted from each block as the averages of the frames intensity. This method can detect the duplicated regions efficiently even when an image was modified by severe rotation transformation.

Some algorithms identify high-entropy regions (key points) in the image and extract feature vectors only at the key points. Therefore, this kind of methods reduce the computational complexity, but may result in missing the duplicated regions completely. Among them, scale invariant feature transform (SIFT) [7] and speed up robust feature (SURF) [8] are applied to CMFD applications. Amerini *et al.* propose a scheme that builds clusters from the locations of detected features and uses random sample consensus (RANSAC) to estimate the geometric transformation between the original area and its replicated areas [9]. Alternatively, the same affine transformation selection (SATS) scheme [10] finds an initial transformation using matched pairs that are spatially close to each other.

The common processing pipeline is discussed by Christlein *et al.* [11] as follows:

Step 1: A color image is used as input image and converted to 8-bit grayscale image. For block based method, the grayscale image is tiled into overlapping blocks. In contrast, for key point based method, the grayscale image is scanned for extracting key points. Next, feature vector descriptor f_i is computed for each block or key point.

Step 2: Every f_i is matched with the similar nearby feature by searching on its nearest neighbor. There exists many matching methods. In the proposed method, correlation is employed as the matching method.

Step3: All the matched feature vectors are clustered in order to find possible copy-move regions. The common used of clustering method are K-Nearest Neighbor, patch-matching, KD-tree and hierarchical clustering.

Step4: All the mismatching pairs are removed by geometric transformation estimation.

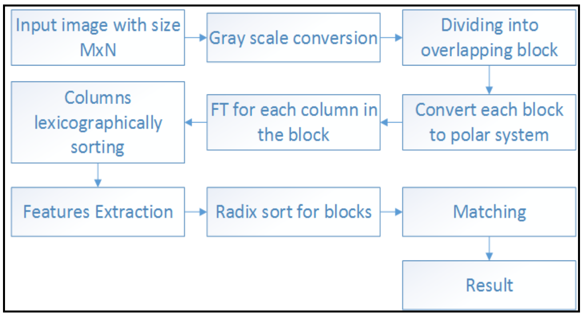
Input image with size MxN
Gray scale conversion
Dividing into overlapping block
Convert each block to polar system
FT for each column in the block
Columns lexicographically sorting
Features Extraction
Radix sort for blocks
Matching
Result

Fig. 2. Block diagram of our proposed algorithm.

In this paper, we propose a hybrid system that utilizes the polar Fourier representation of the blocks and the frequency information of each block as shown in Fig.2. The proposed system is based on using the polar coordinate system of each block to get the frequency correlation features. The Fourier coefficients are calculated for polar block columns. After sorting the Fourier polar block columns, the correlation arrangement is performed between most similar columns.

III. Polar Fourier Representative Feature Extraction

A. Polar System Convertion

The proposed algorithm is applied to intensity band channels of an image. Thus, if the input image is a true color image (24 bit/pixel), it is converted into the corresponding grayscale version by

$$I = 0.299R + 0.587G + 0.114B\,, \tag{1}$$

where *R* is the red channel, *G* is the green channel and *B* is the blue channel of image information.

For an image of the size $M \times N$, the image is divided into small fixed-size overlapping blocks with $b \times b$ pixels by sliding the block one pixel at a time. This results in *K* blocks where $K = (M-b+1)\times(N-b+1)$.

Polar CMFD is based on polar coordinate representation instead of Cartesian coordinate system [12]. The polar coordinate system is a two-dimensional coordinate system in which each point *P* in the plane is determined by a distance r from a fixed point *O* (that is called the pole or origin) and an angle θ from a fixed direction. When converting the point *P* from Cartesian coordinate system (x, y) to polar coordinate system, *P* is represented by the ordered pair (r,θ).

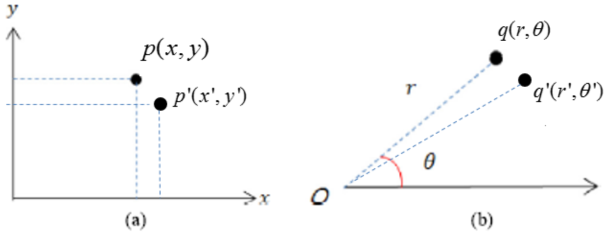

Fig. 3. The relation between polar and Cartesian coordinates.

The relation between polar and Cartesian coordinates can be described by the following formulas:

$$\theta = \tan^{-1}\frac{y}{x} \quad , \tag{2}$$

$$r = \sqrt{x^2 + y^2} \quad . \tag{3}$$

Fig. 3 shows that we set $p = [x, y]^T$ and $p' = [x', y']^T$ as the feature vectors of two pixels in Cartesian coordinate system, $q = [r, \theta]^T$ and $q' = [r', \theta']^T$ as the feature vectors of two pixels in polar coordinate system, matrix $A^{(m,\varphi)} = \begin{bmatrix} m\cos\varphi & m\sin\varphi \\ -m\sin\varphi & m\cos\varphi \end{bmatrix}$ and feature vector $a = [m, \varphi]^T$ to represent the rotation and scaling for pixels.

Hence, we can find that the relationship between the two pixels is given by

$$f(p') = f(A^{(m,\varphi)} \bullet p)\,, \tag{4}$$

which can also be represented by

$$f(q') = f(q + a)\,. \tag{5}$$

Equation (5) is simpler in calculation and the quality can be utilized in our work. So we can find that polar representation will be effective to deal with rotation changes.

For the original image, the horizontal axis is *X*, and the vertical axis is Y. After changing into the polar coordinate system, the horizontal axis is r, and the vertical axis is θ. We call the blocks in polar coordinate system by polar blocks.

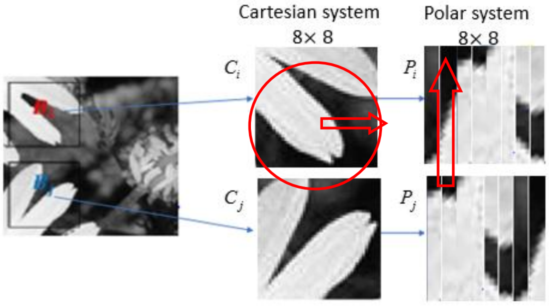

Fig. 4. Cartesian vs. polar representation for two duplicated regions with a rotation alteration. Compare to image P_i, P_j suffers shift in horizontal direction.

Geometric image transformations like (rotation, scaling or reflection) in Cartesian coordinate system result in translation (shifting) in polar coordinate system. Fig. 4 depicts an example, where C_i and C_j are similar blocks with a little rotation. Here we set $b=8$ and make a circle with center point of the block. The polar representation of C_i and C_j is shown in this figure. The 8 columns in the polar block is get from 8 directions with 4 pixels of the circle in the original block. The 8 directions mean 0, 45, 90, …, 315 degrees of the circle. Then we use interpolation to get 8 columns with 8 pixels. The red arrows show that how we put 0 degree pixels of original block into the first column of polar block. It is noticed that both blocks have the same polar image with a little shift in columns. Similarly, the scaling transformation results in row shifting in the polar coordinates. Here we use polar coordinate system, but not log-polar system, for it is not necessary to do the log operation. Log operation is used to strengthen the middle point, weaken the edge point to achieve the effect of human eyes. Actually, polar coordinate system itself can work for small blocks. As the images in our first testing dataset are small, we can calculate transformation at every pixel in the block.

B. Fourier Transform

As mentioned before, Fourier transform (FT) is one of the most effective well-known methods that describe the image frequency and thus use for extraction blocks representative feature. Usually, 2D-FT is computed for each block, and the first coefficient is used as representative feature for each block. In this paper, 1D-FT coefficients are extracted from the polar columns to suit the nature of polar blocks and to be rotation invariant. We also use 1D-FT coefficients which are extracted from the polar rows for the scale modification.

Discrete Fourier Transform (DFT) is shown as follows,

$$X[k]=\sum_{n=0}^{N-1} x[n]e^{-j2\pi\frac{k}{N}n}. \qquad (6)$$

Here $x[n]$ represents the intensity of pixels for one column and $X[k]$ of the coefficients. Usually, researchers use Fast Fourier Transform (FFT) to calculate the coefficients in digital images. Here we use the DFT, but not DCT, because the DFT can retain more information.

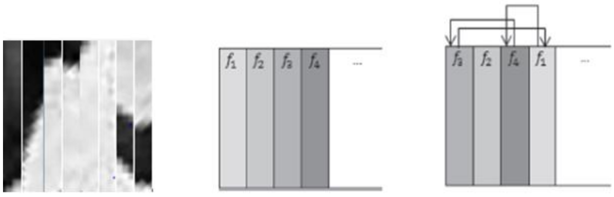

Fig. 5. Example of FT: (a) polar block, (b) FT is computed for each column (c) FT columns are sorted.

Fig. 5 illustrates the procedure of how to make features. For each block, FFT is performed to each column (e.g. f_1, f_2, f_3, f_4). The FFT coefficients of all columns are sorted by radix sort in lexicographical order [13], which is non-comparison-based sorting. Radix sort is a sorting algorithm that sorts data with integer keys. This way can match rotated blocks because both original and rotated blocks have the same columns but in different order. Fig5c presents the sorting procedure (directions are declared as arrows). For example, f_3 moved to the 1st column, f_2 moved to the 2nd column, f_4 moved to the 3rd column and f_1 became the 4th column. Here we describe radix sort in lexicographical order in detail. Lexicographically, there are two arrays of 5 numbers, (1,2,3,5,4) and (1,2,3,4,5), (1,2,3,4,5) is in front of (1,2,3,5,4). According to such a rule, the first of all the arrays consist of the 5 numbers is (1,2,3,4,5), and the last one is (5,4,3,2,1).

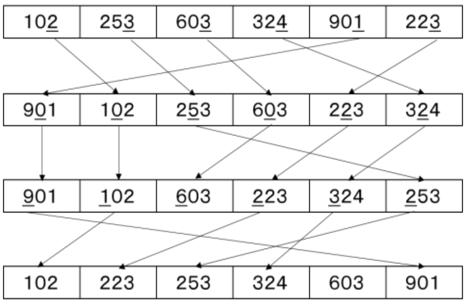

Fig. 6. Example of radix sort.

We calculate the amplitude of FFT coefficients for each column, and then make the columns in lexicographical order. Fig.6 is a simple example for radix sort, and we treat every coefficient as the key. We will compare the coefficients of the column from the final one to the first one.

C. Representative Feature Extraction

The frequency coefficients for each block are reshaped to a row vector in an improved raster scan order as shown in Fig 7.

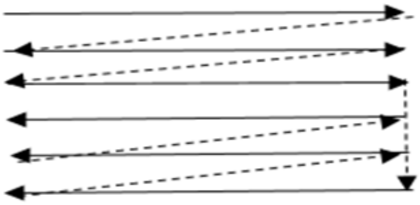

Fig. 7. Improved raster scan.

As we know that high frequencies for the DFT coefficients are located in the middle, so we want to arrange them from small to large in order to get feature vector with suitable length. Therefore we design this improved scan method to achieve our target.

The objective of this step is to get the low frequencies of all the columns after each other at the beginning of the feature vector while keeping high frequencies at the last. For reducing the length of the feature vector and thus reducing the processing time, the number of coefficients greater than n have been truncated. Then we add the indices for block position which require 2 elements. The final feature vector contains *l* elements, where $l = n + 2$.

D. Matching Procedure

Assume saving all the blocks feature vectors in a feature matrix A_s with the size $K \times l$, where each row is considered a block feature and the number of rows refers to the number of blocks, *K*. Here *s* represents the index of the testing images in the dataset, each image has one feature matrix. Radix sort algorithm is also used for sorting the feature matrix to near similar blocks and reduce the exhaustive matching time.

Correlation between each pair of rows in the sorted feature matrix is computed. Assume saving the sorted matrix in A_s, then each row A_{s_j} is compared to the next row $A_{s_{j+1}}$. Correlation between each adjacent row in the sorted blocks is calculated by formula (7),

$$corr = \frac{\sum_{i=1}^{l}(K_{x_i} - \overline{K_x})(K_{y_i} - \overline{K_y})}{\sqrt{\sum_{i=1}^{l}(K_{x_i} - \overline{K_x})^2 \sum_{i=1}^{l}(K_{y_i} - \overline{K_y})^2}} \tag{7}$$

where K_x and K_y represent row A_{s_j} and $A_{s_{j+1}}$, $\overline{K_x}, \overline{K_y}$ are the mean of K_x and K_x respectively, and l is the number of elements in the row. If the correlation is greater than a threshold T_1, the two blocks represented by the two rows are supposed to be similar.

For more smooth areas, the nearby overlapped blocks are usually similar, which may be detected as cloned. So, a spatial distance dis is measured by (8) and only the blocks where $dis > T_2$ are considered for eliminating false positives.

$$dis = \sqrt{(A_{s_j^x} - A_{s_{j+1}^x})^2 + (A_{s_j^y} - A_{s_{j+1}^y})^2} \; . \tag{8}$$

Here $(A_{S_j^x}, A_{S_j^y})$ is the position of block i and $(A_{S_{j+1}^x}, A_{S_{j+1}^y})$ is the position of block $i+1$. Finally, the complete procedure of the proposed method is summarized.

IV. Experiments and Results

A. Datasets and Evaluation Criteria

The experiments are carried out on the Matlab R2012a, RAM 4 GB and processor 2.30 GHz. A publicly available image database for forensics is provided by DVMM, Columbia University [14] and the image manipulation dataset provided by Christlein *et al.* [11].

DVMM Dataset has 933 authentic image blocks of size 128 × 128 pixels. The authentic category refers to those images which are original without any modifications. We use this dataset to produce 1000 CM forged images. The tampered images in the dataset were manually created. In addition, we add modifications, such as JPEG artifacts, noise, additional scaling or rotation by software.

Image manipulation dataset not only provides the tampered images but also a ground truth dataset for comparison. This dataset has 96 images in different types from original to be manipulated.

All parameters of the experiment are set as: T_1 = 0.99:0.9995, T_2 = 16, the block size *b* = 8, overlapping size equals to 1 and $l = 26$. We perform six types of experiments to examine the robustness of our system.

The images shown in Fig. 8 present the results of CRM forensic using T_1 only (Fig. 8a) where false positives are circled, and using T_2 (Fig. 8b).

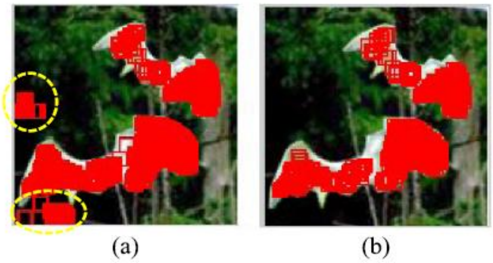

Fig. 8. Results of copy–rotate–move forensic.

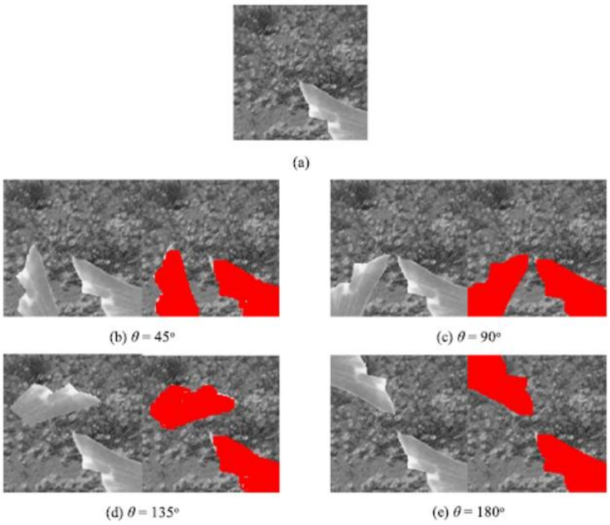

Fig. 9. Results of Copy–Rotate–Move forensic under different angles.

B. Visual Result

Experiment 1: CM forgery under rotation: in this experiment, a region is copied and rotated by some angles before being pasted in another place. Fig. 9 presents an example of Copy–Rotate–Move (CRM) detection with different angles 45 °, 90 °, 135 °, and 180 ° marked on the tampered images. Our proposed method called PCM successfully detects duplicated region under rotation with different angles.

Experiment 2: CM forgery under reflection: the images shown in Fig. 10 presents an example of CM with reflection, with/without rotation by different angels (10 °, 20 °, and 40 °).

Experiment 3: CM forgery under scaling: in this experiment, a random region is copied and scaled before being pasted into another place in the image. Fig. 11 presents the detection result for Copy–Scale–Move (CSM) forgery. (a) original Image, (b) CSM tempered image, (c) detection result.

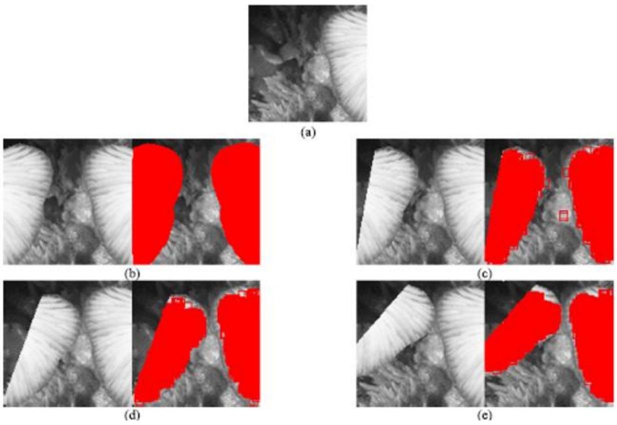

Fig. 10. Results of CM detection with reflection and rotation effects.

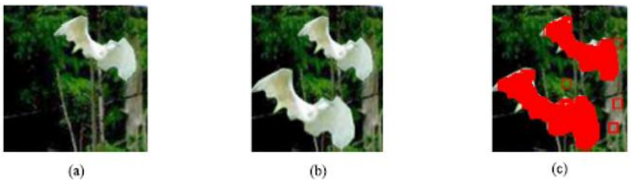

Fig. 11. Result of CSM detection regions

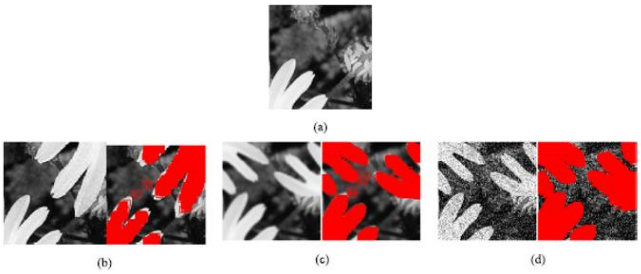

Fig. 12. Results of CM detection regions under complex modifications.

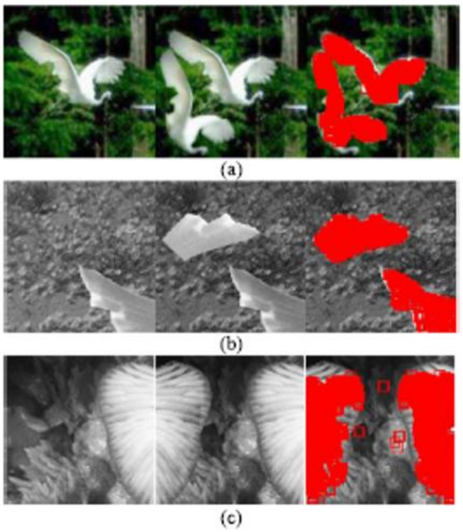

Fig. 13. Results of CM forgery detection after applying intensity modifications to CM region only.

Experiment 4: CM forgery in the wake of complex modifications: the proposed method has been evaluated with a complex modification. The results of this experiment are shown in Fig. 12(a) The original image(b) The copied region image has been modified by two transformations; rotating by 180°, scaling by 2 and compressed by JPEG with quality Q = 30. (c) The copied region was rotated twice by (180°, 90°) and blurred by Gaussian blurring. (d) The copied region was rotated twice by (180°, 90°) and Gaussian noise was added to the tempered image.

Experiment 5: CM detection in wake of modifying only the CM region. We have increased the detection complexity by applying intensity modification to the copied region only. Fig. 13 presents the detection results where the CM region was modified a little before pasting. (a) The region was blurred with Gaussian blurring, (b) after increasing brightness, (c) after decreasing brightness, all images are presented in the order (left) the original, (middle) CM forged image and (right) the detection result.

C. Comparison with Other Methods

The system also has been compared to some related work systems. Table I presents the performance of PCM algorithm compared to four other algorithms; DCT [1], EB [4], PCA [5] and CRMS [6] algorithms.

Measuring PCM performance: In this experiment, 1000 tampered images under different modifications have been processed for CM forgery detection by PCM proposed system. The detection accuracy in image level is represented by

$$P_1 = \frac{\textit{the number of correctly detected images}}{\textit{the number of all detected images}} .$$

And also the detection accuracy in pixel level is represented by

$$P_2 = \frac{\textit{the number of correctly detected pixels}}{\textit{the number of all detected pixels}} .$$

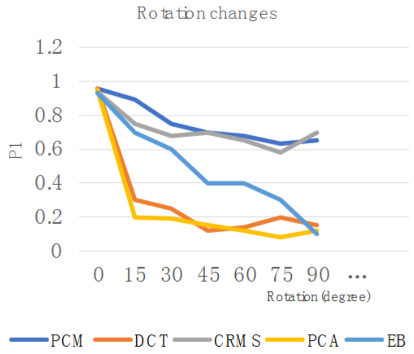

Fig. 14. Comparision with Rotation changes.

Fig. 14 shows the P_1 when rotation changes by different methods. From it we can see that our method can effectively resist the rotation changing while DCT and PCA method cannot work well when rotation is large. Our method is comparable with other state-of-art methods.

Table I shows the average of the precision rate, which measured on the average number of test images when the images contain a duplicated region without modification (WM), under a rotation modification (RM), scale modification (SM), Gaussian blur (GB), noise addition (NA) and JPEG compression with quality of 90.

We compare our method with recent work ULPF [15] and SIFT based method [16] with the 96 tampered images. Precision rate is calculated by P_2 and results are shown in Table II. The result is the average of different degree for the same modification. We can see that our method is comparable with ULPF and better than SIFT.

TABLE I: THE PERFORMANCE OF PCM ALGORITHM IS COMPARED WITH DCT [1], EB [4], PCA [5] AND CRMS [6]

P_1	WM	RM	SM	GB	NA	BM	JPEG	T(s)
PCM	95.2	62.5	65.5	89.1	61.2	86.1	46.3	9.50
DCT	96.3	18.5	12.1	86.2	43.0	87.2	40.0	4.70
EB	95.3	27.3	36.2	91.3	21.3	89.0	32.0	7.68
PCA	93.4	17.2	19.2	72.1	63.02	79.3	47.3	3.56
CRMS	98.9	60.1	61.5	90.0	54.2	80.6	53.0	1.52

TABLE II: RESULTS OF CM FORGERY DETECTION AFTER APPLYING MODIFICATIONS

P_2	WM	RM	NA	JPEG
PCM	70	65	56	32
ULPF	72	60	59	35
SIFT	47	42	37	26

The execution time of the proposed system has been compared to others. In this experiment, the average execution time for 1000 forged images used in this research. Table I presents the time performance of PCM algorithm compared to DCT [1], EB [4], PCA [5] and CRMS [6] algorithms. It is noticed that PCM takes the highest execution time while CRMS has the lowest. This is due to the existence of domain conversion and sorting steps compared to CRMS algorithm. Although it seems more complex and longer algorithm, forensic systems usually work offline and the execution time is not the most important issue.

The proposed method achieves higher detection accuracy than the comparison works as shown in Table I. However, the accuracy of our method for the CM forgery detect is almost equivalent to the one of ULPF as shown in Table II. There has a room for improvement of our method in this direction. And our proposed method requires more execution time than the other comparison methods.

In the experiments, the performance of our method compares with some conventional methods using rather simple benchmark datasets, DVMM etc. A more complicated dataset for several evaluations should also be utilized in the future work.

V. CONCLUSION

In this paper, we have proposed a method for block based copy–move forgery detection. The proposed method has been evaluated using DVMM splicing dataset and an image manipulation dataset. The experimental results show that the proposed method can detect CM forgery in an image with a relative robustness to JPEG compression, Gaussian blurring, rotating, scaling and reflection.

Although the proposed method achieves higher detection accuracy in comparison with several conventional methods, there still exists some limitations that are waited to be solved. The major drawback of my proposed method is the execution time. The proposed method allocates more time for compilation than the other comparison methods. A more complicated dataset for evaluation should also need to be used in the future work.

REFERENCES

[1] Y. Huang, W. Lu, W. Sun and D. Long, "Improved DCT-based detection of copy–move forgery in images," *Forensic Science International*, vol. 206, no. 1-3, pp. 178-184, Jan. 2011.

[2] J. Fridrich, "Digital image forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 26-37, Feb. 2009.

[3] Q. Wu, S. Wang, and X. Zhang, "Detection of image region-duplication with rotation and scaling tolerance," *Computational Collective Intelligence, Technologies and Applications*, pp. 100-108, 2010.

[4] G. Lynch, F. Y. Shih and H. Y. M. Liao, "An efficient expanding block algorithm for image copy–move forgery detection," *Information Sciences*, vol. 239, pp. 253-265, 2013.

[5] A.C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions. Technical report TR2004-515," *Department of Computer Science,* 2004.

[6] S. M. Fadl, N. Semary, and M. M. Hadhoud, "Copy–rotate–move forgery detection based on spatial domain," in *Proc. 2014 9th IEEE International Conference on Computer Engineering & Systems*, 2014, pp. 136-141.

[7] A. Myna, M. Venkateshmurthy, and C. Patil, "Detection of region duplication forgery in digital images using wavelets and log-polar mapping," in *Proc. the International Conference on Computational Intelligence and Multimedia Applications*, 2007, pp. 371-377.

[8] S. Nikolopoulos, S. Zafeiriou, N. Nikolaidis, and I. Pitas, "Image replica detection system utilizing R-trees and linear discriminant analysis," *Pattern Recognition*, vol. 43, no. 3, pp. 636–649, Mar. 2010.

[9] J. H. Hsiao, C. S. Chen, L. F. Chien, and M. S. Chen, "A new approach to image copy detection based on extended feature sets," *IEEE Transactions on Image Processing*, vol. 16, no. 6, pp. 2069-2079, June 2007.

[10] M. K. Bashar, K. Noda, N. Ohnishi, and K. Mori, "Exploring duplicated regions in natural images," *IEEE Transactions on Image Processing,* no. 99, pp. 1-1, Mar. 2010.

[11] V. Christlein, C. Riess, and J. Jordan, "An evaluation of popular copy-move forgery detection approaches," *IEEE Transactions on Information Forensics and Security,* vol. 7, no. 6, pp. 1841-1854, Dec. 2012.

[12] I. Reyzin, "Cartesian to polar coordinate transformation," U.S. Patent, No. 6,064,388, May 16, 2000.

[13] M. Zagha and G. E. Blelloch, "Radix sort for vector multiprocessors," in *Proc. the ACM/IEEE Conference on Supercomputing*, 1991, pp. 712–721.

[14] T. T. Ng, S. Chang, and Q. Sun, "A data set of authentic and spliced image blocks," ADVENT Technical Report, Columbia University, pp. 203-2004, 2004.

[15] C. S. Park, C. Kim, J. Lee, and G. R. Kwon, "Rotation and scale invariant upsampled log-polar Fourier descriptor for copy-move forgery detection," *Multimedia Tools and Applications*, vol. 75, no. 23, pp. 16577-16595, Dec. 2016.

[16] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics and Security,* vol. 10, no. 3, pp. 507-518, Mar. 2015.

Yitian Wang was born in 1993 in Jiangsu, China. In 2015, she earned the bachelor degree from South China University of Technology in Guangzhou, China and earned the master degree in 2017 from Graduate School of Information, Production and Systems of Waseda university in Kitakyushu, Japan. During school, she participated in the facial AU detection research, and the main research direction is image processing.

Sei-Ichiro Kamata received the M.S. degree in computer science from Kyushu University, Fukuoka, Japan, in 1985, and the Ph.D. degree in computer science from the Kyushu Institute of Technology, Kitakyushu, Japan, in 1995. From 1985 to 1988, he was with NEC, Ltd., Kawasaki, Japan. In 1988, he joined the Kyushu Institute of Technology, as a faculty Member. From 1996 to 2001, he was an associate professor with the Department of Intelligent System, Graduate School of Information Science and Electrical Engineering, Kyushu University. Since 2003, he has been a professor with the Graduate School of Information, Production and Systems, Waseda University. In 1990 and 1994, he was a visiting researcher with the University of Maine, Orono, USA. His research interests include image processing, pattern recognition, image compression, and space-filling curve applications.