Fast Copy-Move Forgery Detection

HWEI-JEN LIN[1], CHUN-WEI WANG[1] and YANG-TA KAO[2]
[1]Department of Information Engineering and Computer Science
Tamkang University
[2]Department of Network Information Technology
Chihlee Institute of Technology
[1]151 Ying-Chuan Road, Tamsui
Taipei, Taiwan, ROC
[2]313 Wun-Hua Road, Sec. 1, Banciao,
Taipei, Taiwan, ROC
[1]086204@mail.tku.edu.tw, http://pria.cs.tku.edu.tw
[2]ydkao@mail.chihlee.edu.tw, http://int.chihlee.edu.tw

Abstract: - This paper proposes a method for detecting copy-move forgery over images tampered by copy-move. To detect such forgeries, the given image is divided into overlapping blocks of equal size, feature for each block is then extracted and represented as a vector, all the extracted feature vectors are then sorted using the radix sort. The difference (shift vector) of the positions of every pair of adjacent feature vectors in the sorting list is computed. The accumulated number of each of the shift vectors is evaluated. A large accumulated number is considered as possible presence of a duplicated region, and thus all the feature vectors corresponding to the shift vectors with large accumulated numbers are detected, whose corresponding blocks are then marked to form a tentative detected result. Finally, the medium filtering and connected component analysis are performed on the tentative detected result to obtain the final result. Compared with other methods, employing the radix sort makes the detection much more efficient without degradation of detection quality.

Key-Words: - Forgery Detection, Copy-move Forgery, Singular Value Decomposition (SVD), Principal Component Analysis (PCA), Lexicographical Sort, Scale Invariant Feature Transform (SIFT) Descriptors, Log-polar Coordinates, Radix Sort, Connected Component Analysis

1 Introduction

Art forgery dates back more than two-thousand years. With the widespread use of Internet and availability of powerful image processing and editing software, digital images are easy to acquire through internet and to manipulate and edit. For example, the image "*Fonda Speak To Veitnam Veterans At Anti-War Rally*" shown in Fig. 1(a) was synthesized using the two images shown in Fig.s 1(b) and 1(c). For protecting copyright and preventing forgery or alteration of document with malicious intentions, the tasks of forgery detection become more and more urgent.

Recently, various techniques for temper or forgery detection or even recovery have been proposed in the literature. Some techniques have been proposed for image tamper detection and recovery. Various watermark techniques [1][2][3][4][5][6] have been proposed in recent years, which can be used not only for authentication, but also for being an evidence for the tamper detection. Wang et al. [7] and Lin et al. [8] both embedded watermarks consisting of the authentication data and the recovery data into image blocks for image tamper detection and recovery in the future. An example of image tamper detection and recovery given by Lin et al. [8] is shown in Fig. 2. Li et al. [9] transformed the image from its spatial domain to the frequency domain based on the DWT, extracted some information from the frequency domain as the eigenvalue of this image, and then hid this eigenvalue in the middle frequency band of the frequency domain as an evidence for the tamper detection. The eigenvalue hidden in the image could be used to recover this image

The drawback of watermark techniques is that one must embed a watermark into the image first. Many other techniques that work in the absence of any digital watermark or signature have been proposed. Popescu [10] detected resampling (e.g., scaling or rotating) based on statistical correlations. E. S. Gopi et al. [11] exploited the property of correlation by using Auto Regressive coefficients as the feature vector for identifying the location of digital forgery in a sample image.

The major weakness of this approach is that it is only applicable to uncompressed TIFF images, and JPEG and GIF images with minimal compression. Some researchers [12][13][14][15] estimated light source direction [16] and used lighting inconsistencies for revealing traces of digital tampering. Defects of cameras such as chromatic aberration [17][18] and sensor pattern noise [19][20][21], and the color filter arrays the cameras use for interpolating colors can be also used to detect forgeries [22].

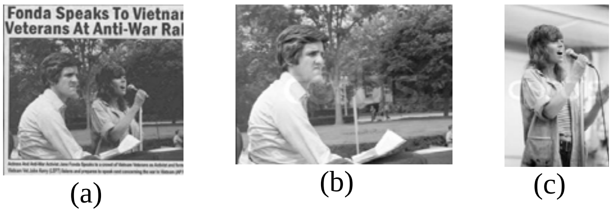


Fig. 1. (a). A synthesized image "*Fonda Speak To Veitnam Veterans At Anti-War Rally*"; (b)&(c). original images.

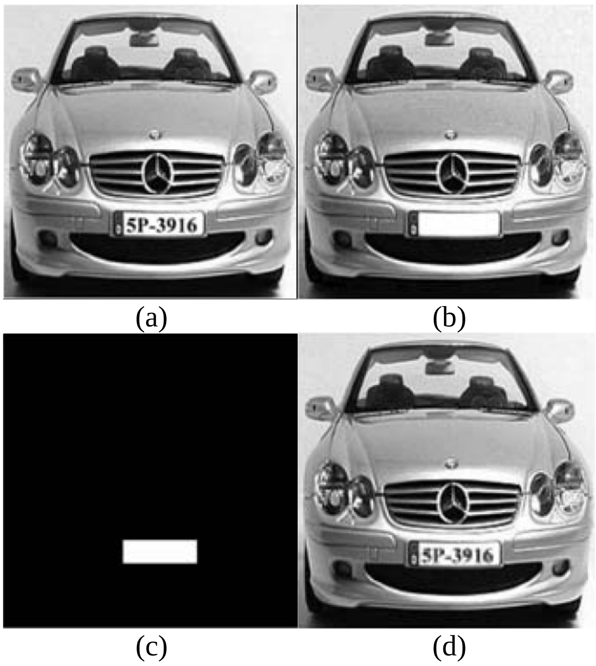


Fig. 2. Image tamper detection and recovery (a). original image; (b). tampered image; (c). result of temper detection; (d). recovered image.

Copy-move forgery is a specific type of image tampering, where a part of the image is copied and pasted. Many methods have been proposed to detect copy-move forgeries [23][24][25][26][27][28]. G. Li et al. [25] applied DWT to the given image, and used SVD on fixed-size blocks of low-frequency component in wavelet sub-band to yield a reduced dimension representation, then lexicographically sorted the SV vectors to detect duplicated image blocks. A. C. Popescu et al. [24] applied a principal component analysis (PCA) on small fixed-size image blocks to yield a reduced dimension representation. Duplicated regions are then detected by lexicographically sorting all of the image blocks. W. Luo et al. [26] also first divided an image into small overlapped blocks and extracted block characteristics vector, and then compared the similarity of these blocks to identify possible duplicated regions. A. N. Myna et al. [27] presented an approach based on the application of wavelet transform that detects and performed exhaustive search to identify the similar blocks in the image by mapping them to log-polar coordinates and using phase correlation as the similarity criterion. H. Huang et al. [28] first extracted SIFT descriptors of an image, which are invariant to changes in illumination, rotation, scaling etc. Owing to the similarity between pasted region and copied region, descriptors are then matched between each other to seek for any possible forgery in images. Fig. 3 shows some examples of copy-move forgery detection results yielded by the method proposed by A. N. Myna et al. [27] and by the method proposed by H. Huang [28].

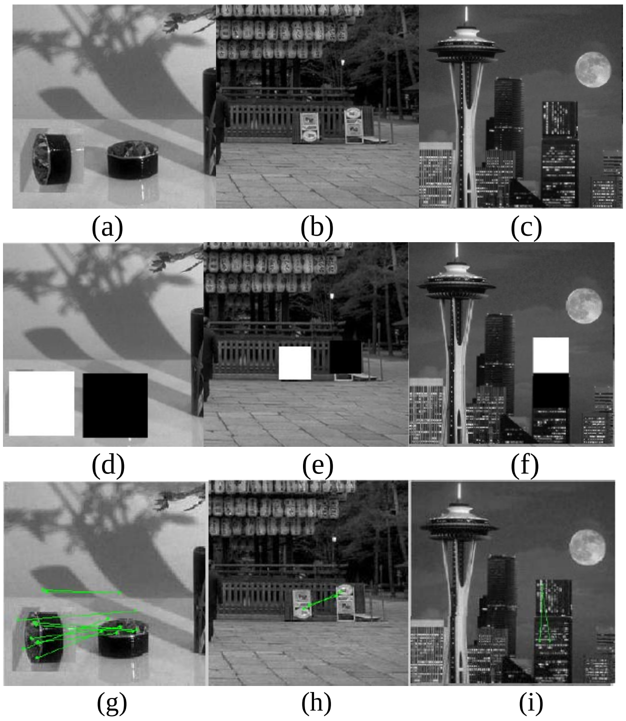


Fig. 3. (a), (b) & (c). original images; (d), (e), & (f). detected results by A. N. Myna et al.; (g), (h), & (i). detected results by H. Huang et al..

A good copy-move forgery detector should be robust to some types of manipulations including lossy compression, Gaussian noise, and rotation or scaling. Most of the existing methods do not deal with all those manipulations and are time-consuming.

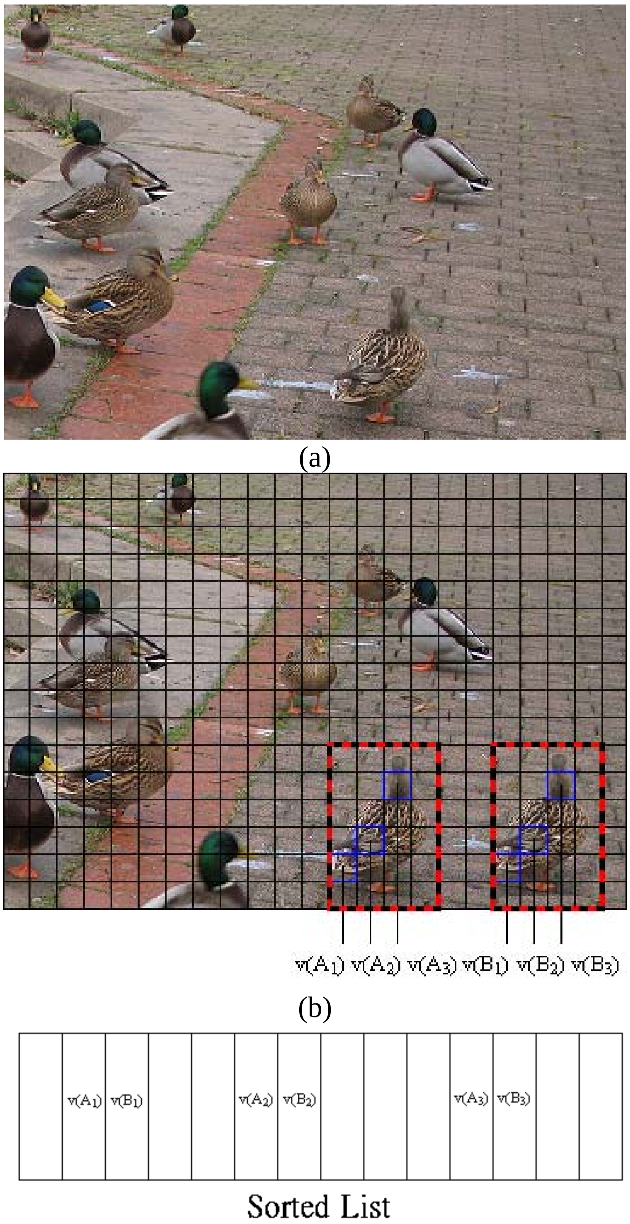
In this paper, we focus on detection of the copy-move forgery, and propose an efficient method for detecting copy-move forgeries in digital images, which is robust to some types of manipulations including lossy compression, Gaussian noise, and rotation. To improve the computational complexity in detecting the regions of forgeries, we propose to use the radix sort for sorting the feature vectors of the divided sub-blocks, as an alternative to lexicographic sorting, which is commonly used by the existing copy-move forgery detection schemes. Our experimental results show that the proposed method can detect copy-move forgeries in the images very accurately, even when the copied region was undergone severe image manipulations, such as additive Gaussian noise, lossy JPEG compression, and rotation etc, or even compound processing. In addition, it is observed that use of radix sort considerably improves the time efficiency at the expense of a slight reduction in the robustness.

The rest of this paper is organized as follows. Related work is discussed in Section 2. In Section 3, the proposed method is described in details. In Sections 4 and 5, we show some experimental results and make a conclusion for this paper.

2 Related Work

In most methods of copy-move forgery detection, the detected image is divided into overlapping blocks of equal size, which are represented in the form of (feature) vectors, and then lexicographically sorted for later detection. Suppose a detected image of size $N \times N$ is divided into $k = (N\text{-}b+1)^2$ overlapping blocks of size $b \times b$, represented as vectors of length b^2. In the sorted list, vectors corresponding to blocks of similar content would be close to each other in the list, and thus identical regions could be easily detected by evaluating shift vectors formed by pairs of adjacent feature vectors, and detecting large accumulated numbers of shift vectors.

The image shown in Fig. 4(b) is a tampered result, where the regions enclosed by dotted rectangles are duplicated, of the original image given in Fig. 4(a) by incurring copy-move forgery. Dividing the tempered image into overlapping blocks for forgery detection, we can observe that, for example, blocks B_1, B_2, and B_3 are copies of blocks A_1, A_2, and A_3, respectively. If we let v(x) denote the feature vector of a block X, then $v(A_1) = v(B_1)$, $v(A_2) = v(B_2)$, and $v(A_3) = v(B_3)$. When the feature vectors are sorted, identical vectors would be grouped together in the sorted list, as shown in Fig. 4(c), from which the duplicated regions could be easily detected. The time required by sorting the feature vectors is dependent on both the total number of divided blocks and the size of feature vectors.



(a)

(b)

(c)

Fig. 4. (a). An original image; (b). Three pairs of identical blocks are enclosed by blue squares; (c). Sorted list of feature vectors, in which identical vectors.

A. C. Popescu et al. [24] used the principle component analysis (PCA) and represented each block of size 16× 16 as a feature vector of length 32, and lexicographically sorted the vectors in $O(32\times k\lg k)$ time. The time complexity for sorting was reduced by G. Li et al. [25] to $O(8k\lg k)$ by the use of SVD. W. Luo et al. [26] defined a feature vector of 7-dimension to represent blocks so as the time complexity for sorting was further reduced to $O(7k\lg k)$. In this paper, we shall propose a further efficient method for sorting the feature vectors, whose time complexity is reduced to $O(9k)$.

3 The Proposed Method

For resisting against various modifications and improving the efficiency for sorting feature vectors, we represent each block B of size $b\times b$ $(=16\times 16)$ by a 9-dimensional feature vector $v_B = (x_1, x_2, \ldots, x_9)$, which is defined as follows. Firstly, the block B is divided into four equal-sized sub-blocks, S$_1$, S$_2$, S$_3$, and S$_4$, as shown in Fig. 5 and let *Ave*(.) denote the average intensity function. Then as described in (1), f_1 denotes the average intensity of the block B, the entries f_2, f_3, f_4, and f_5 denote the ratios of the average intensities of the blocks S$_1$, S$_2$, S$_3$, and S$_4$ to f_1, respectively, and f_6, f_7, f_8, and f_9 stand for the differences of the average intensities of the blocks S$_1$, S$_2$, S$_3$, and S$_4$ from f_1, respectively. Finally, entries f_i's are normalized to integers x_i's ranging from 0 to 255, as described in (2), where $\lfloor\cdot\rfloor$ denotes a *floor* operator. Although these 9 entities contain duplicated information, they together possess higher capability of resistance against some modifications, such as JPEG compression and Gaussian noise.

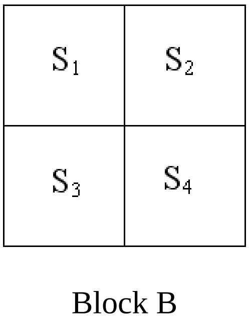


Fig. 5. A block B is divided into four equal-sized sub-blocks S$_1$, S$_2$, S$_3$, and S$_4$.

$$f_i = \begin{cases} f_i = Ave(B) & \text{if } i = 1, \\ Ave(S_{i-1})/(4Ave(B)+\varepsilon_1) & \text{if } 2 \le i \le 5, \\ f_i = Ave(S_{i-5}) - Ave(B) & \text{if } 6 \le i \le 9. \end{cases} \quad (1)$$

$$x_i = \begin{cases} \lfloor f_i \rfloor & \text{if } i = 1, \\ \lfloor 255 \times f_i \rfloor & \text{if } 2 \le i \le 5, \\ \left\lfloor 255 \times \dfrac{f_i - m_2}{m_1 - m_2 + \varepsilon_2} \right\rfloor & \text{if } 6 \le i \le 9, \end{cases} \quad (2)$$

where $m_1 = \max_{6\le i\le 9}\{f_i\}$ and $m_2 = \min_{6\le i\le 9}\{f_i\}$.

Unlike the matrix constructed by A. C. Popescu et al. [24], which stores floating numbers, the feature vectors we extract store integers. As a result, we may use the efficient radix sort algorithm to perform lexicographical sorting over those vectors. If the given image of size $N\times N$ is divided into overlapping blocks of size $b\times b$, then there are totally k blocks, where $k = (N-b+1)^2$. Let v_1, $v_2,\ldots,v_k$ be the feature vectors corresponding to these k blocks. To perform radix sort on these vectors of size 9, we regard each of them as a 9-digit number with each digit ranging from 0 to 255. The sorting algorithm is given in the following, where the input array A stores these vectors; that is, $A[i] = v_i, 1\le i\le k$, and d = 9.

```
RADIX-SORT(A,d)
    for $j \leftarrow 1$ to $d$
        do use a stable sort to sort array $A$ on digit $j$
```

Since each digit in the vectors, ranging from 0 to 255, is not large, counting sort is chosen as the stable sort used in the radix sort. Each pass over k numbers then takes time O(256+k). There are 9 passes, so the total time for sorting the feature vectors is O(9(256+k)) = O(9k) since 256 << k.

From what follow, we let v_1, $v_2,\ldots,v_k$ denote sorted list of the feature vectors of blocks B$_1$, B$_2$,…, B$_k$, respectively. The position of the top-left corner point of each block B$_i$ is recorded in P(B$_i$) and a shift vector is defined as the difference of two adjacent feature vectors in the sorted list as shown in (3).Two duplicated regions caused by copy-move forgery form a number of pairs of identical feature vectors, each pair then make the same shift vector, thus the accumulative number of a shift vector can be used to detect the duplicated regions.

$$u(i) = P(B_{i+1}) - P(B_i) \qquad (3)$$

As the example illustrated in Fig. 6, several pairs of corresponding feature vector make same vector u, whose accumulative number is 7 in this example. With those accumulative numbers of shift vectors, we detect the duplicated regions as follows. For the accumulative number of a shift vector greater than a given threshold T_1, the four corner points of all the corresponding blocks are marked. For example, if the accumulated number of a shift vector u_0 is greater than T_1, then for each i, the top-left points of the respective blocks B_i and B_{i+1} corresponding to v_i and v_{i+1} are marked if $u(i) = u_0$. Fig. 7(a) shows the result of marked points for Fig. 4(b). Finally, the medium filtering is performed to remove noises and the connected component analysis is applied to obtain the final detected result as given in Fig. 7(b).

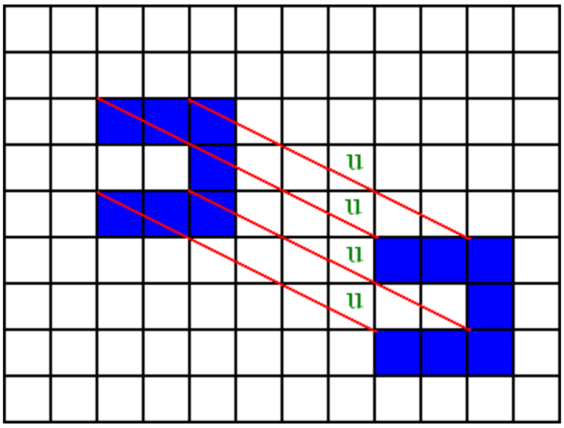
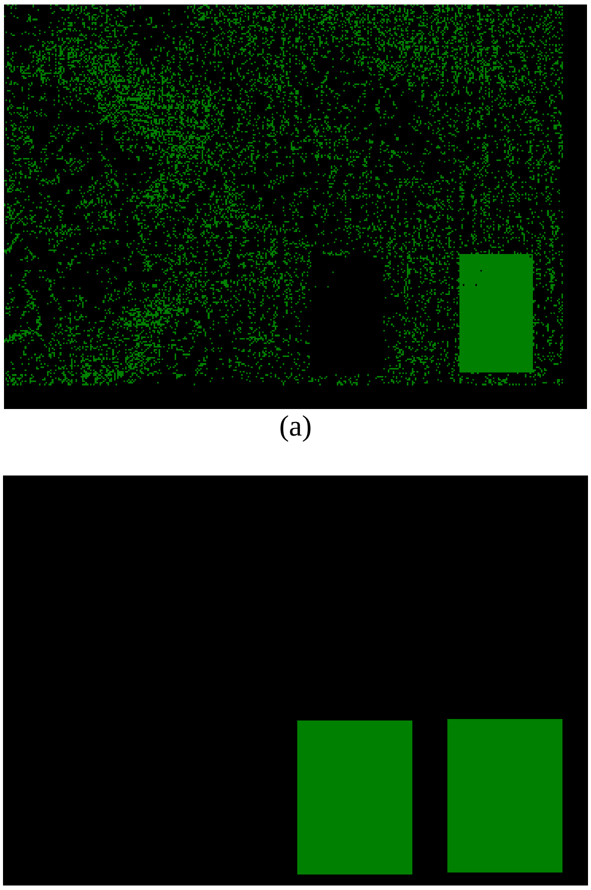


Fig. 6. Duplicated regions form several identical shift vector u.

To deal with rotation, we simply detect on the given image associated with its rotated versions. In our experiments, we considered rotations through angles of 90, 180, and 270 degrees. This way we may detect rotated copy-move forgeries with any angle of rotation. As shown in Fig. 8, the region is copied, rotated by angle 90 degrees, and pasted to another region in the image. In this case, the accumulated number of shift vectors cannot reflect the duplication. To detect rotated copied images, we combine three rotated versions of the image with the original one, and perform forgery detection on this combined image.



(a)

(b)

Fig. 7. (a). Corner points of detected blocks are marked according to the accumulated numbers of shift vectors for the tampered image given in Fig. 4(b); (b). final detected result.

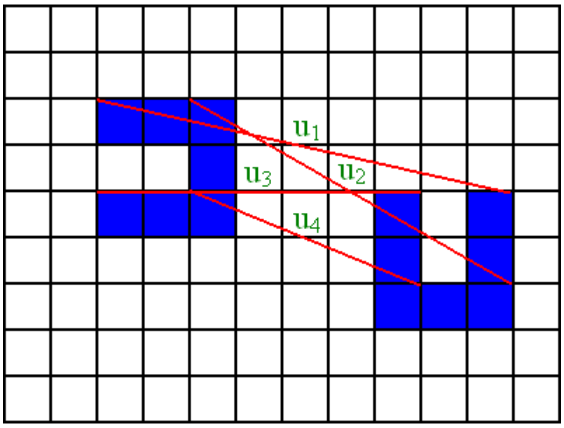
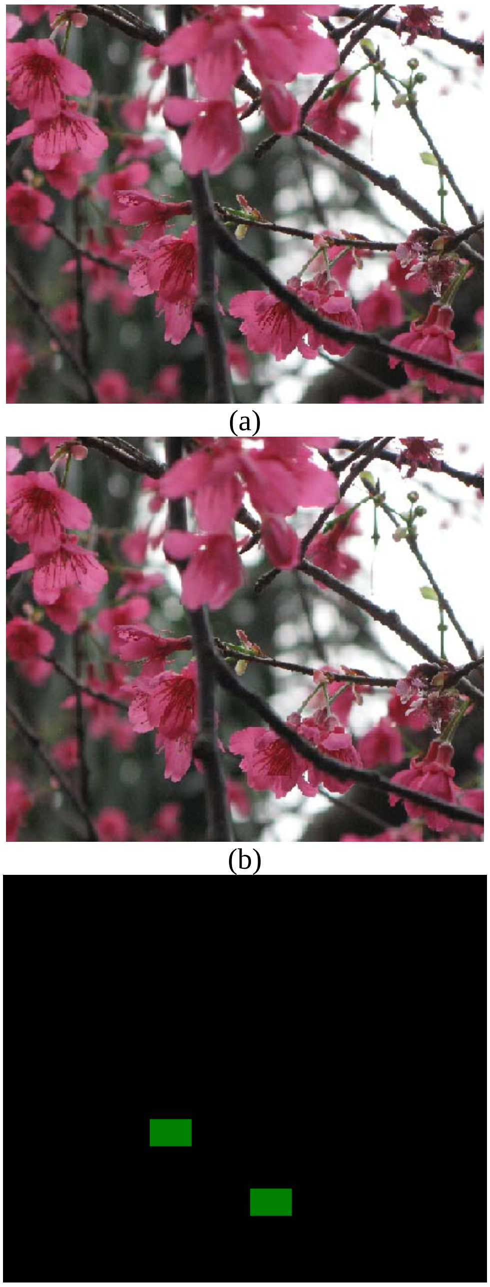


Fig. 8. A region is copied, rotated through 90 degrees, and pasted to another region.

4 The Experimental Results

The proposed method was implemented on a computer of CPU 3.0GHz with memory 1GB. The test images were cropped from 50 natural images. We tested over 50 tampered images with no modification, 150 tampered images with JPEG compression, and 150 tampered images with Gaussian noise, 50 tampered images with rotation images, 150 tampered images with rotation and Gaussian noise, and 150 tampered images with rotation and JPEG compression. For detecting on color images, only the green channel is used since the human eyes are most sensitive to the green color. For parameter setting, we set b = 16, T_1 = 100, and T_2 = 10.

More detected results over tampered images are shown in Fig.s 9~11. Fig. 12 shows the detected results over compressed tampered images with various quality factors. Fig. 13 shows the detected results over some images with Gaussian noise at various SNRs (signal to noise ratios). Fig. 14 shows the detected results over some images with rotation. Table 1 shows detection rates for some datasets of copy-move images with some modification. Table 2 detection rates for some datasets of copy-move images with rotation and some other modification.

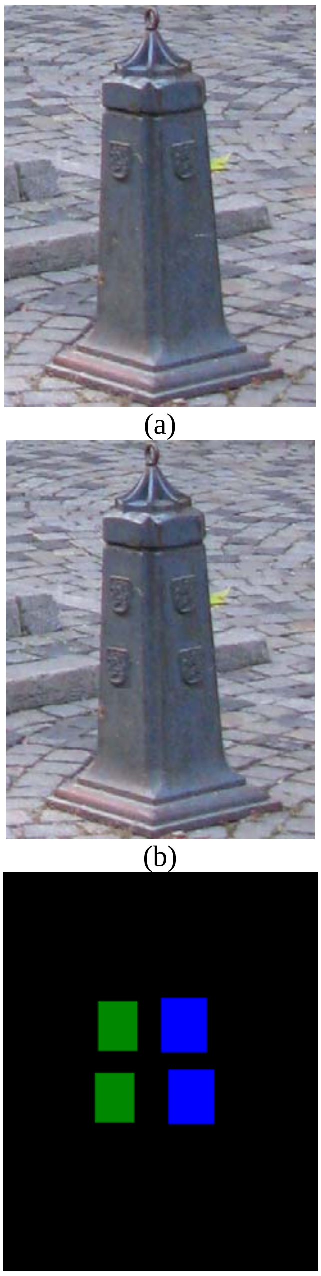


(a)

(b)

(c)

Fig. 9. (a). The original images; (b). the tampered images; (c). the detecting results.



(a)

(b)

(c)

Fig. 10. (a). The original images; (b). the tampered images; (c). the detecting results.

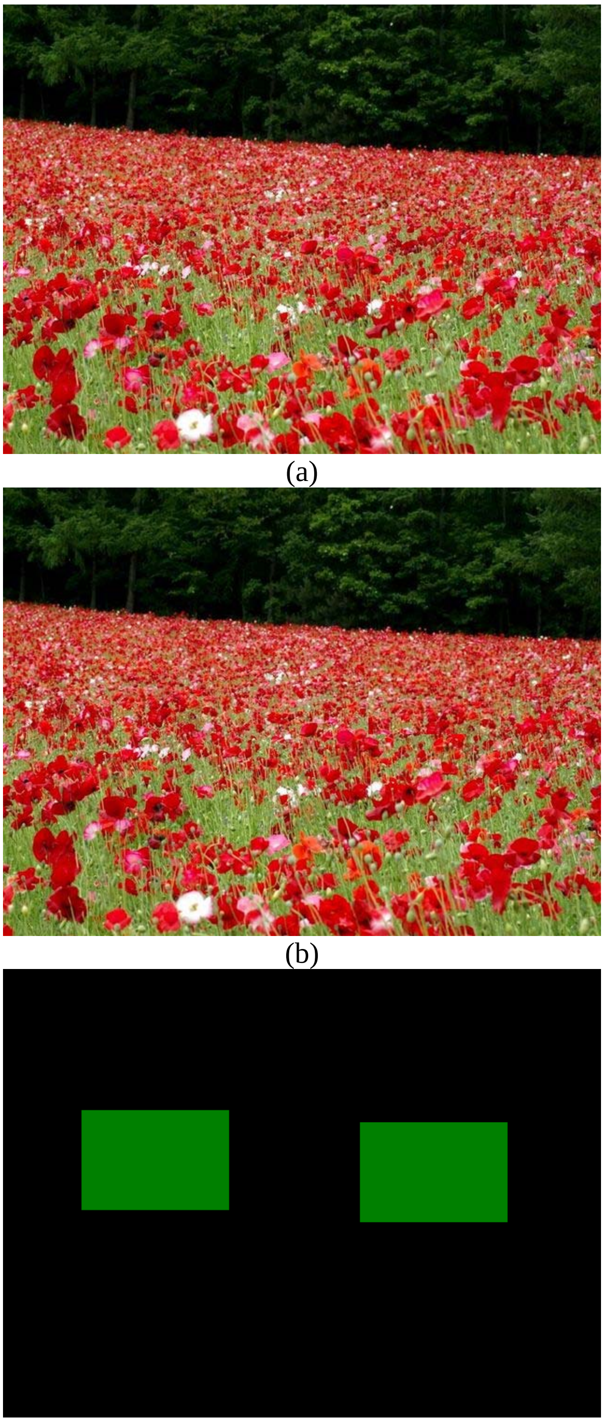

(a)
(b)
(c)

Fig. 11. (a). The original images; (b). the tampered images; (c). the detecting results.

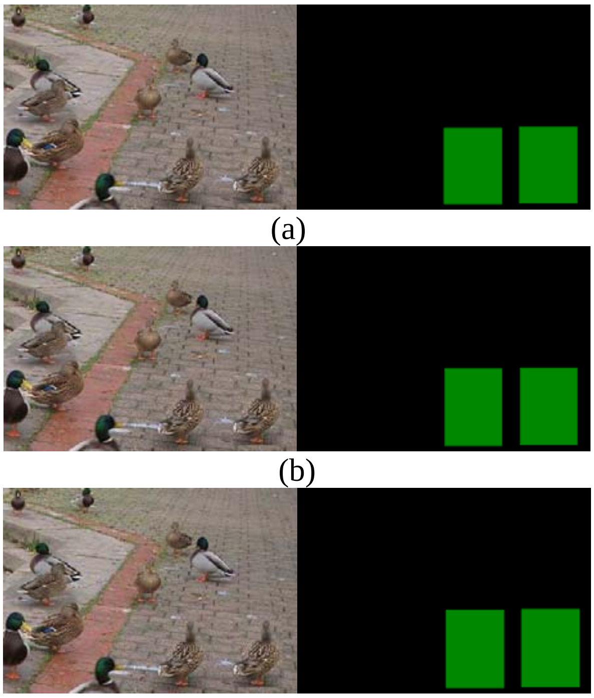

(a)
(b)
(c)

Fig. 12. Detected results over compressed versions of the image given in Fig. (4a), with various quality factors (QFs): (a). QF = 90; (b). QF = 70; (c). QF = 50.

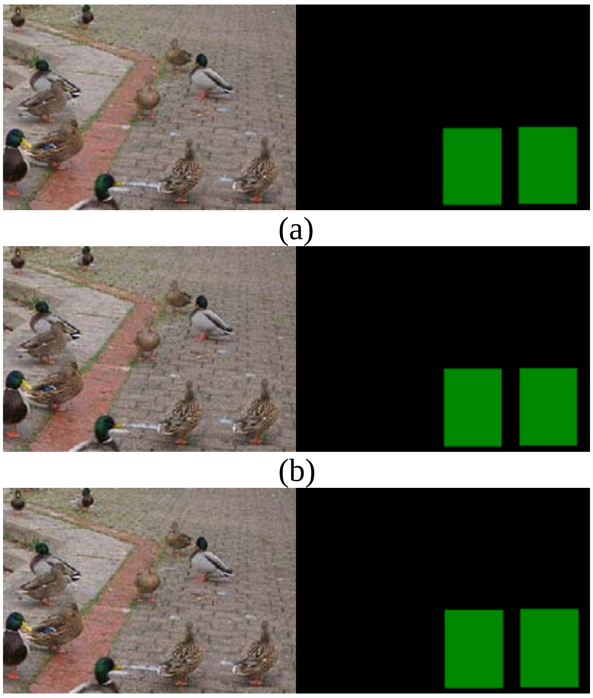

(a)
(b)
(c)

Fig. 13. Detected results for the image given in Fig. (4a) with Gaussian noise at various SNRs: (a). SNR = 10db; (b). SNR = 20db; (c). SNR = 35db.

Table 1. Detection rates for datasets of copy-move with/without modification.

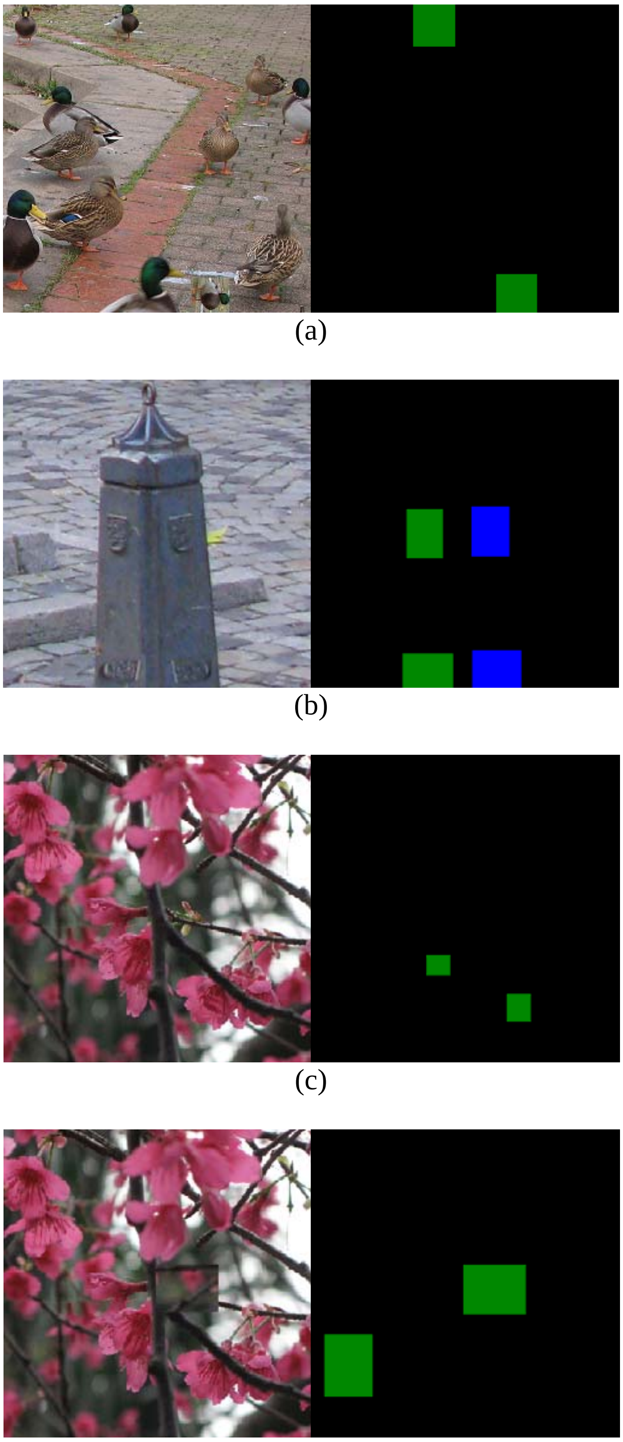
Data sets of Copy-move images	No. of images	Detection rate (%)
without midification	50	98
JPEGcompression QF = 100	50	98
JPEG compression QF = 90	50	98
JPEG compression QF = 80	50	96
Gaussian noise SNR = 10	50	98
Gaussian noise SNR = 20	50	98
Gaussian noise SNR = 35	50	94

Table 2. Detection rates for datasets of copy-move images with rotation and some other modification.

Data sets of Copy-move images with rotation	No. of images	Detection rate (%)
without midification	50	98
JPEGcompression QF = 100	50	94
JPEG compression QF = 90	50	94
JPEG compression QF = 80	50	88
Gaussian noise SNR = 10	50	98
Gaussian noise SNR = 20	50	78
Gaussian noise SNR = 35	50	54

4 Conclusion and Future Work

In this paper, we propose an efficient method for copy-move forgery detection. Using of radix sort dramatically improves the time complexity and the adopted features enhance the capability of resisting of various attacks such as JPEG compression and Gaussian noise. Both efficiency and high detection rates have been demonstrated in our experimental results. However, a few small copied regions were not successfully detected. Although duplicated regions with rotation through some fixed angles can be detected, our method does not deal with rotation arbitrary angles. In the future, we would like to search for some feature invariant to rotation to deal with this problem. In addition to rotation problem, we would try to extend our work to video images.



(a)

(b)

(c)

(d)

Fig. 14. Detecting results for the rotated duplicated regions.

Acknowledgements

The work presented in this paper was supported by the National Science Council, grant NSC 97-2221-E-263 -002.

References:

[1] 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.

[2] C. T. Hsieh, Y. K. Wu, and K. M. Hung, "An Adaptive Image Watermarking System Using Complementary Quantization," *WSEAS Transaction on Information Science and Applications*, Vol. 3, Issue 12, 2006, pp. 2392-2397.

[3] K. M. Hung, C. T. Hsieh and Y. K. Wu, "Multi-Purpose Watermarking Schemes for Color Halftone Image Based on Wavelet and Zernike Transform," *WSEAS Transaction on Computer*, Vol. 6, Issue 1, 2007, pp. 9-14.

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

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

[6] W. Lu, F. L. Chung, and H. Lu, "Blind Fake Image Detection Scheme Using SVD," *IEICE Transaction on Communications*, Vol. E89-B, No. 5, May 2006, pp. 1726-1728.

[7] M. S. Wang and W. C. Chen, "A Majority-Voting based Watermarking Scheme for Color Image Tamper Detection and Recovery", *Computer Standards & Interfaces*, Vol. 29, Issue 5, 2007, pp. 561-570.

[8] P. L. Lin, C. K. Hsieh, and P. W. Huang, "A Hierarchical Digital Watermarking Method for Image Tamper Detection and Recovery, *Pattern Recognition*, Vol. 38, Issue 12, 2005, pp. 2519-2529.

[9] K. F. Li, T. S. Chen, and S. C. Wu, "Image Tamper Detection and Recovery System Based on Discrete Wavelet Transformation," *International Conference Communications, Computers and Signal Processing*, Vol. 1, 2001, pp. 26-28.

[10] A. C. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Traces of Resampling," *IEEE Transactions on Signal Processing*, Vol. 53, 2005, pp. 758-767.

[11] E. S. Gopi, N. Lakshmanan, T. Gokul, S. KumaraGanesh, and P. R. Shah, "Digital Image Forgery Detection using Artificial Neural Network and Auto Regressive Coefficients," *Electrical and Computer Engineering*, 2006, pp. 194-197.

[12] M. K. Johnson and H. Farid, "Exposing Digital Forgeries by Detecting Inconsistencies in Lighting," *in Proceedings of ACM Multimedia and Security Workshop*, New York, 2005, pp. 1-9.

[13] R. Brunelli, "Estimation of Pose and Illuminant Direction for Face Processing," *Image and Vision Computing*, Vol. 15, No. 10, October 1997, pp. 741-748.

[14] A. P. Pentland, "Finding the illuminant direction," *Journal of the Optical Society of America*, Vol. 72, Issue 4, 1982, pp. 448-455.

[15] W. Zhou and C. Kambhamettu, "Estimation of Illuminant Direction and Intensity of Multiple Light Sources," *in Proceedings of the 7th European Conference on Computer Vision-Part IV*, 2002, pp. 206-220.

[16] P. Nillius and J. O. Eklundh, "Automatic Estimation of the Projected Light Source Direction," *in Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, 2001, pp. 1076-1083.

[17] T. E. Boult and G. Wolberg, "Correcting Chromatic Aberrations Using Image Warping," *in Proceedings of Computer Vision and Pattern Recognition*, 1992, pp. 684-687.

[18] M. K. Johnson and H. Farid, "Exposing Digital Forgeries Through Chromatic Aberration," *in Proceedings of the 8th workshop on Multimedia and security*, 2006, pp. 48-55.

[19] J. Lukas, J. Fridich, and M. Goljan, "Detecting Digital Image Forgeries Using Sensor Pattern Noise," *in Proceedings of the SPIE Conference on Security, Steganography, and Watermarking of Multimedia Contents*, Vol. 6072, January 2006, pp. 362-372.

[20] N. Khanna, A. K. Mikkilineni, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Scanner Identification Using Sensor Pattern Noise," *in Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, Vol. 6505, No. 1, 2007, pp. 65051K.

[21] J. Lukas, J. Fridrich, and M. Goljan, "Determining Digital Image Origin Using Sensor Imperfections," *in Proceedings of SPIE Electronic Imaging, Image and Video*

Communication and Processing, January 16-20, 2005, pp. 249-260.

[22] 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.

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

[24] 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.

[25] 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.

[26] W. Luo, J. Huang, and G. Qiu, "Robust Detection of Region Duplication Forgery in Digital Image," *in Proceedings of the 18th International Conference on Pattern Recognition*, Vol. 4, 2006, pp. 746-749.

[27] A. N. Myna, M. G. Venkateshmurthy, and C. G. Patil, "Detection of Region Duplication Forgery in Digital Images Using Wavelets and Log-Polar Mapping," *in Proceedings of the International Conference on Computational Intelligence and Multimedia Applications* (ICCIMA 2007), Vol. 3, 2007, pp. 371-377.

[28] H. Huang, W. Guo, and Y. Zhang, "Detection of Copy-Move Forgery in Digital Images Using SIFT Algorithm," *in Proceedings of IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, Vol. 2, 2008, pp. 272-276.