Detection of Copy-Move Forgery Using Krawtchouk Moment

Mustafa Bilgehan İmamoğlu[1], Güzin Ulutaş[1], Mustafa Ulutaş[1]

[1]Department of Computer Engineering, Karadeniz Technical University, Trabzon, Turkey
bilgehan@ktu.edu.tr, guzin@ieee.org
ulutas@ieee.org

Abstract

Copy move forgery is a popular image tampering technique that copies part of the image onto another region of the same image. Detection is possible by exploring similar regions in an image and based on regional features' similarity. Many feature extraction algorithms are used to extract information from non overlapping blocks of the image. Similar features from separate regions are an indication of a copy move forged image. Image moments are used to represent the image blocks during forgery detection recently. Krawtchouk moments are used to extract regional features and detect copy move forgery in this paper. They are not used to extract features and detect forged images before. Experimental results indicate that the proposed method can detect copy move forgery for both regular and irregular shaped regions. Besides, the method is resilient to additive white Gaussian noise, blurring attacks. Experimental results also show that the method has higher accuracy ratio compared to similar works for post processed (Gaussian blurred) forged images.

1. Introduction

The demand for high resolution and low cost digital imaging hardware resulted in widespread use of digital imaging devices. Digital images are widely used in many areas. Sometimes they are used as evidence during an investigation. It is easy to create tampered images by using both proprietary and open source image editing tools. For example, in July 2010, Malaysian politician claimed to have been knighted by the queen Elizabeth II. The image was a splicing between an original ceremony photo and politician's face as can be seen in Fig. 1. In 2004, a photograph showing one of the candidate and a famous actress shared a demonstration podium in the 1970s was a fake in fact. These examples show that developing techniques to verify the integrity and authenticity of the images is an important issue to make the digital images more trustworthy.

Digital image authentication methods can be roughly divided into two categories: Active and passive (blind) methods. Active methods use digital signatures and/or watermark techniques to ensure the integrity of the images. Watermarking techniques requires insertion of some data (called watermark) into image when it was taken. Since present imaging device do not specially equipped with such a module, using active approaches to authenticate an image is not practical.

Passive methods do not need the presence of any priori information about the image. They use some statistical features of the image to determine forgery. There are two common form of digital image tampering: copy-move forgery and image splicing. Image splicing creates a forged image by cutting a region of the image and pasting it into another image. The other form of forgery is usually used to conceal a special object in the original image. Some region of the image is copied and pasted into another region of the image to create a forged image. Researchers have developed methods to detect this form of forgery.

The first method for detecting copy move forgery was proposed by Fridrich et al. in 2003 [1]. Their work divides an image into non overlapping blocks of 8×8 size. Discrete Cosine Transform (DCT) is used to extract features from the blocks. Then feature vectors are lexicographically sorted and similar vectors are analyzed to determine the forgery. After this work, in 2004, Popescu et al. used PCA to extract features from the blocks [2]. Their work aimed to decrease the dimension of the feature vector. In 2009, Bayram et al. used Fourier Mellin Transform to make the forgery method robust against copy-rotate-move forgery attacks [3]. Their method also utilized counting bloom filters during the similarity search. Some local visual features such as SIFT is also used to detect the forgery [4, 5].

(a) (b)

Fig. 1. (a) Original image (b) Forged image

In recent years image moments are also used to extract features from the non overlapping blocks of the image. In 2007, Mahdian et al. used blur moments to detect copy move forgery [8]. Their work represented each block by their blur invariants which are function of central moments. Their work also utilized k-d tree representation during the similarity search. In 2010, Ryu et al. extracted information about the blocks using Zernike moments [9]. Magnitude of Zernike moments were compared to judge the forgery. Image moment is a weighted average of the image pixels' intensities. Moments have the ability to represent global features of the image. As a consequence, they find extensive applications in image processing. In 2003, Yap et al. proposed a new set of orthogonal image moment called Krawtchouk moment [10]. It is based on the discrete classical Krawtchouk polynomials.

In this paper, Krawtchouk moments are used to extract features of non overlapping blocks which forms the image. For each block, the Krawtchouk moments of order $(n+m)$ are calculated to form the feature vector. Then, blocks' similarities are tested by inspecting the lexicographically sorted array of

features. Experimental results show that the method successfully detects copy move forgery. The method is also resilient to additive white Gaussian noise and blurring attacks as shown in the results. The method has higher accuracy ratios when Gaussian blurring is used as post processing operation during forgery.

The rest of the paper is organized as follows. Section 2 gives the detail of the Krawtchouk moments. The details of the proposed method are given in section 3. Experimental results are shown in section 4. At last, conclusion is drawn in section 5.

2. Krawtchouk moment

In this section, a brief explanation of the Krawtchouk polynomials introduced by Mikhail Krawtchouk is given [10]. The definitions of the Krawtchouk moments are also introduced in this section.

2.1. Krawtchouk polynomials

n-th order classical Krawtchouk polynomial is given in (1).

$$K_n(x;p,N) = \sum_{k=0}^{N} a_{k,n,p} x^k = {}_2F_1\left(-n,-x;-N;\frac{1}{p}\right) \quad (1)$$

where $x, n = 0, 1, 2, \cdots, N$ and $N > 0, p \in (0,1)$. ${}_2F_1$ given in (2) is the hypergeometric function

$${}_2F_1(a,b;c;z) = \sum_{k=0}^{n} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!} \quad (2)$$

The definition of the Pochammer symbol denoted by $(a)_k$ is given in (3).

$$(a)_k = a(a+1)\cdots(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)} \quad (3)$$

The set of (N+1) Krawtchouk polynomials form a complete set of discrete basis functions with weight function given in (4).

$$\omega(x;p,N) = \binom{N}{x} p^x (1-p)^{N-x} \quad (4)$$

Orthogonality condition is satisfied as the following

$$\sum_{x=0}^{N} \omega(x;p,N) K_n(x;p,N) K_m(x;p,N) = \rho(n;p,N)\delta_{nm} \quad (5)$$

where $n, m = 1, 2, \cdots, N$. ρ is defined as in (6).

$$\rho(n;p,N) = (-1)^n \left(\frac{1-p}{p}\right)^n \frac{n!}{(-N)_n} \quad (6)$$

Krawtchouk polynomials are normalized according to norm as given in (7).

$$\widetilde{K_n}(x;p,N) = \frac{K_n(x;p,N)}{\sqrt{\rho(n;p,N)}} \quad (7)$$

The square root of the weight given in (4) is used as a scaling factor to create weighted Krawtchouk polynomials as in (8).

$$\overline{K_n}(x;p,N) = K_n(x;p,N)\sqrt{\frac{\omega(x;p,N)}{\rho(n;p,N)}} \quad (8)$$

2.1. Krawtchouk polynomials

Local features of an image can be obtained by Krawtchouk moments. The moment is calculated as given in (9) using weighted polynomial for an image $f(x,y)$ of size N×M. Q_{nm} denotes the (n+m)th order Krawtchouk moment.

$$Q_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \overline{K_n}(x;p_1,N-1)\, \overline{K_m}(y;p_2,M-1) \times f(x,y) \quad (9)$$

The image can be constructed from the moments using (10).

$$f(x,y) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \overline{K_n}(x;p_1,N-1)\, \overline{K_m}(y;p_2,M-1) \times Q_{nm} \quad (10)$$

3. Forgery detection using Krawtchouk moments

The details of the proposed copy move forgery detection algorithm is given in this section. The method uses Krawtchouk moments to extract some features from the blocks. The algorithm consists of two phases. The image is partitioned into non overlapping blocks at first and corresponding feature vectors are stored in a matrix. Then, this matrix is lexicographically sorted to move similar vectors closer. In the second phase, the matrix is searched for similar blocks. Suspicious block pairs are denoted by shift vectors. Shift vector between two blocks is designated by a pair of integers corresponding to distances between upper left x and y-axes coordinates of these blocks. Blocks are marked as forged if the number of suspicious block pairs that have the same shift vectors exceed a certain threshold.

Input image *f* of *N*×*M* pixels is partitioned into overlapping *B*×*B* pixel blocks by sliding the window one pixel at a time from top left to bottom right in row major to create $(N-B+1)(M-B+1)$ blocks. Krawtchouk moments for all blocks are then calculated to form a feature matrix of $N_{blocks} \times m_{ord}^2$. Features in *i*th row, F_i are constructed by using (11) where m_{ord} is maximum moment order determined by user at run time.

$$F_i = \left[Q_{00} Q_{01} \cdots Q_{0m_{ord}} \cdots Q_{m_{ord}m_{ord}}\right] \quad (11)$$

Q_{nm} denotes Krawtchouk moments of (*n*+*m*)th order. Feature extraction transforms *B*×*B* blocks to $1 \times m_{ord} m_{ord}$ size vectors. Number of elements in feature vectors is a parameter set by user. Rows of feature matrix *A* of size $N_{blocks} \times m_{ord}^2$ are then lexicographically sorted. Sorting feature matrix rows relocates feature vectors closer corresponding to similar blocks. Similarity of blocks is then tested by comparing rows within a user defined window of size *r*. Feature vector in *i*th row of *A*, A^i, is compared with candidate feature vectors denoted by $A^{i+1}, \cdots, A^{i+r}$. Square root of the sum of squared differences of corresponding elements in both A^i and one of the candidate vectors A^{i+r} is used as the similarity measure given in (12).

$$\sqrt{\sum_{j=1}^{m_{ord}^2} \left(A_j^i - A_j^{i+r}\right)^2} < T_{sim} \quad (12)$$

T_{sim} is a user defined threshold value to judge similarity among blocks. The magnitude of difference between the corresponding elements is a clue about the similarity of these two vectors. If two vectors are similar, the coordinates of blocks

represented by these feature vectors are used to measure the distance between the blocks. Assume that upper left corner coordinates of these blocks denoted by (x^i, y^i) and (x^{i+r}, y^{i+r}) respectively. The distance between two blocks d is calculated using (13).

$$d = \sqrt{(x^i - x^{i+r})^2 + (y^i - y^{i+r})^2} \quad (13)$$

The frequency of the corresponding shift vector is incremented if the distance between blocks is smaller than a predetermined threshold value T_{dist},. The shift vector s between two corresponding block is given in (14).

$$s = (s_x, s_y) = \left((x^i - x^{i+r}), (y^i - y^{i+r})\right) \quad (14)$$

All rows of matrix A is processed in a similar manner. Each feature vector is tested for similarity with candidate feature vectors as in (12). The counter indicating frequency of the shift vector (s_x, s_y) is incremented if blocks are similar and the distance between them is smaller than T_{dist}. Shift vectors' frequency is then compared with a threshold value T corresponding to minimum area of copy moved blocks in forged image.

Blocks that contribute the frequency of the shift vector are marked as copy pasted blocks if the frequency of the current shift vector (s_x, s_y) is greater than T. Opening operation is applied on the marked image to remove isolated regions and to improve the performance.

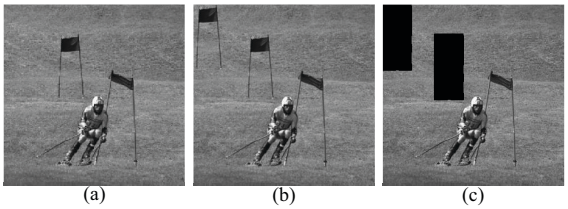
4. Experimental results

In this section, experiments are done on forged images to show the effectiveness of the proposed method. 512×512 8-bit gray level images are copy-move forged with GIMP, an open source image editing software. Algorithms explained in the previous section are coded in Matlab and run on a computer. Maximum order m_{ord} is assumed to be 5 for all experiments resulting 1×25 feature vectors. The value of T_{sim} and d are determined to be 4 and 16 respectively. Two metrics denoted by p (accuracy) and f (false negative) are used to test the capability of the method. Performance metrics defined in [6, 7] are used as in (15), where D_1, D_2 denote non overlapping duplicated regions, R_1, R_2 represent detected duplicated regions, $|.|$ is the area of regions and $\cap$, $\cup$ are the intersection and union of regions respectively.

$$p = \frac{|D_1 \cap R_1| + |D_2 \cap R_2|}{|D_1| + |D_2|} \quad f = \frac{|D_1 \cup R_1| + |D_2 \cup R_2|}{|D_1| + |D_2|} - p \quad (15)$$

Accuracy and false negative parameters indicate the precision of the method in detection of the duplicated regions. The method detects duplicated regions more accurately for larger values of p and smaller values of f. The first experiment is done to show the effectiveness of the proposed method for regular shape (rectangular) copy-move forgery. The original image and tampered image of 512×512 pixels are given in Fig. 2(a) and 2(b) respectively. Detected regions are indicated by black regions and the method can detect duplicated regions as can be seen in Fig. 2(c).

Irregular shaped duplicated regions can also be detected by the proposed method. The second experiment is done to detect irregular shaped duplicated regions in tampered images given in Fig 3(a) and 3(b). Duplicated regions detected by the algorithm are given in Fig. 3(c).



(a) (b) (c)

Fig. 2. (a)Original image (b)Tampered image (c)Detection results of the method.

(a) (b) (c)

Fig. 3. (a)Original image (b)Tampered image (c)Detection results of the method.

Result indicates that the proposed method can detect the duplicated region even if the replaced region has an irregular shape. Some post processing operations such as blurring, adding Gaussian noise can be applied on the forged image to hide evidence of the copy pasted regions. Two experiments are realized to show the robustness of the method for both Gaussian blurring and additive Gaussian noise. First, the tampered image is blurred by a Gaussian blur filter with parameters $w = 5, \sigma = 0.5, 1, 1.5, 2$. Detection results of the forged images distorted by Gaussian blurring with specified parameters are given in Table 1.

Table 1. Detection results of the forged image under Gaussian Blurring

Quality Factor	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$
p (accuracy)	0,95	0,92	0,91	0,90
f (false negative)	0,04	0,07	0,092	0,097

Table 1 indicates that proposed method can detect forged images even if they are Gaussian blurred. Accuracy ratios of the proposed method and related work reported in the literature are compared under different post processing operations in the last experiment. Fig 4 shows the comparison of the methods, when the tampered image is contaminated with additive white Gaussian noise (SNR= 5, 10, 15, 20, 25, 30 dB). Accuracy of all methods decreases as SNR decrease. Proposed method has higher accuracy ratios compared to PCA and DCT based methods. The method also has similar results with FMT based methods.

Last experiment is realized to show the comparison of accuracy ratios of the methods under different blurring operations. Gaussian blur filter with parameters $w = 5, \sigma = 0.5, 1, 1.5, 2, 2.5, 3$ is applied after forgery. Fig. 5 shows that the proposed method has higher accuracy ratios with respect to other works in the literature. Results of the experiments indicate

that the proposed method can detect forgery successfully for both regular and irregular shaped regions. The method can also detect forgery under different post processing operations. Accuracy ratios of the method are higher than that of other methods when the post processing operation is Gaussian blurring.

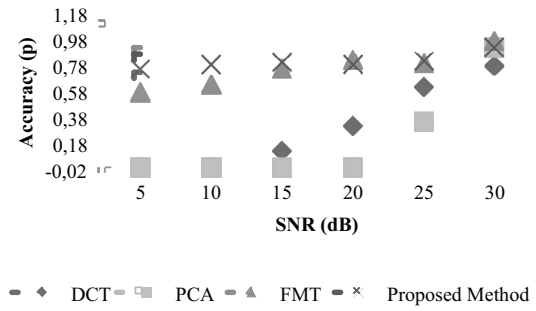


Fig. 4. Comparison of accuracy as a function of image SNR

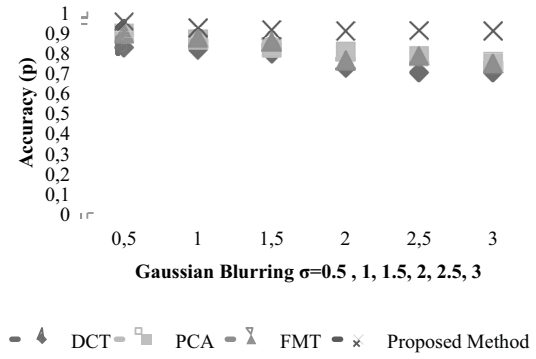


Fig. 5. Comparison of accuracy with Gaussian blurring

5. Conclusions

Fast development of the image editing software makes the tampering of the images easier. Two approaches are used in digital image authentication: Active and Passive. Passive methods utilize some features of the images to determine the forgery. Image moments gained popularity in the forgery detection recently. Krawtchouk moments are used to detect copy-move type of image forgery in this paper. The results indicate that the proposed method can detect forgery of regular or irregular shaped regions. The method is also robust to different post processing operations as shown in the results. When the Gaussian blurring is used as post processing operation, the method yields higher accuracy ratios compared to other works. It is planned to use invariant Krawtchouk moments to detect copy rotate move type forgery as a future work. Thus, the method can even detect copy rotate move type forgery.

6. References

[1] Fridrich, J., "Detection of copy-move forgery in digital images", *Digital Forensic Research Workshop*, Cleveland, OH, 2003, pp. 19–23.

[2] A.C. Popescu, H. Farid, "Exposing digital forgeries by detecting duplicated image regions", *Tech. Rep. TR2004-515*, Dartmouth College, 2004.

[3] Bayram, S, Sencar, H., Memon, N., "An efficient and robust method for detecting copy-move forgery"*, IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 1053- 1056.

[4] Pan, X., Lyu, S., "Region duplication detection using image feature matching", *IEEE Trans. On Information Forensics and Security*, 5 (4), 2010, pp. 857-867.

[5] Amerini, I., Ballan, L., Caldelli, R., Del Bimbo, A., Serra, G., "A sift based forensic method for copy move attack detection and transformation recovery", *IEEE Trans on Information Forensics and Security,* 6 (3), 2011, pp. 1099-1110.

[6] Huang, Y., Lu, W., Sun, W., Long, D., "Improved DCT-based detection of copy-move forgery in images, "*Forensic Science International*, 206(1-3), pp. 178- 184.

[7] Cao, Y., Gao, T., Fan, L., Yang, Q., "A robust detection algorithm for copy move forgery in digital images", *Forensic Science International*, vol. 214, 2012, pp. 33-43.

[8] Mahdian, B., Saic, S., "Detection of copy move forgery using a method based on blur moment invariants", "*Forensic Science International*", vol. 171, 2007, pp. 180-189.

[9] Ryu, S.-J., Lee, M.-J., Lee, H.-K., "Detection of copy rotate move forgery using Zernike moments", International Conference on Information Hiding, 2010, pp. 51-65.

[10] Yap, P.-T., Paramesran, R., Ong, S.-H., "Image Analysis by Krawtchouk Moments", IEEE Transactions on Image Processing, vol. 12, no. 11, 2003, pp. 1367-1377.