Detection of Image Region-Duplication with Rotation and Scaling Tolerance

Qiumin Wu, Shuozhong Wang, and Xinpeng Zhang

School of Communication and Information Engineering, Shanghai University,
Shanghai 200072, China
{rynax_ls,shuowang,xzhang}@shu.edu.cn

Abstract. Copy-move forgery, or region-duplication, is a common type of digital image tampering. This paper presents a novel approach to detect copy-move forgery even if rotation and/or scaling took place before copying. The image under test is divided into fixed-size blocks, and log-polar Fourier transform (LPFT) is performed on the inscribed circles of these blocks. Similarity in the LPFT between different patches provides an indication of the copy-move operation. Experimental results show efficacy of the proposed method.

Keywords: digital forensics, copy-move forgery, log-polar transform, rotation and scaling invariance.

1 Introduction

With the proliferation of sophisticated image-editing software, digital images can be easily manipulated without leaving any perceptible traces. Although digital watermarking can be used in image authentication, application of watermarking is limited as a watermark must be embedded into the image before any tampering occurs. To this end, passive image forensic techniques have been developed for image authentication without the need of any precautious measures, and therefore are more flexible and practical. These techniques work on the assumption that although image forgeries may leave no visual clues of tampering, they alter the underlying statistics of an image.

Copy-move is a common type of image manipulation, which copies one part of the image and pastes into another in the same image. Detection of copy-move forgery is based on the presence of duplicated regions in the tampered image even though the image has undergone post-processing operations such as smoothing and local modifications. Exhaustive search is a straightforward way to detect copy-move, but it has very high computation complexity and is unable to deal with situations where the pasted object has been rescaled and/or rotated.

To reduce computational complexity, research has been done to use block-matching. Some related methods are based on lexicographic sorting of quantized DCT coefficients of image block [1] or PCA results of fix-sized image blocks [2] or using blur moment invariants as eigenvalues of image blocks [3]. In practice, copy-move forgery is often accompanied by scaling and rotation. For example,

J.-S. Pan, S.-M. Chen, and N.T. Nguyen (Eds.): ICCCI 2010, Part I, LNAI 6421, pp. 100–108, 2010.

(a) (b)

Fig. 1. Copy-move with scaling and rotation (a) original image, and (b) forged image

the balloon in Fig. 1(a) is rotated by 15° and scaled by a factor of 0.8 before pasting into the same image to produce the result of Fig. 1(b). Features used in some previous methods such as DCT coefficients and eigenvectors of PCA bases are changed when the duplicated region is scaled/rotated, and therefore the above methods will fail in these cases.

To solve the problem, log-polar transform (LPT) may be performed on image blocks followed by wavelet decomposition [4] or lexicographic sorting [5]. Consideration of LPT comes from its scaling/rotation invariance and its successful use in other fields such as watermarking [6] and image registration [7]. Bayram et al. [8] proposed an approach based on Fourier-Mellin transform (FMT), in which each block is first Fourier transformed, and the resulting magnitude values are then mapped into log-polar coordinates. FMT values are finally projected to one dimension to give the feature vectors. Although these methods can detect copy-move forgery with scaling, they do not work well when large angle rotation of the pasted object is involved. For example, the upper limit of rotation is 10° in the method of [8].

In this paper, we propose an approach to detect copy-move image forgery with rotation/scaling tolerance. The method is based on the log-polar Fourier transform (LPFT). The image to be tested is divided into fixed-size blocks, and the inscribed circles of the blocks are taken to give circular regions of the same size. Unlike the method introduced in [8] where log-polar mapping is done in the Fourier domain of the image, we first perform a log-polar transform (LPT) on every circular region, which is in the image domain, and then take 2D Fourier transform of the LPT result. Similarity between the original and forged regions is revealed by comparing the cross-spectra of the LPFT results. Experimental results show that the proposed method is effective even if scaling and large-angle rotation occurred in the tampered regions.

2 Copy-Move Forgery Detection Using LPFT

As stated in the above, LPFT consists of two steps: LPT and FT. The purpose of LPT is to convert image scaling and rotation into translation of the log-polar representation. We identify tampered regions by comparing cross-spectrum coefficients of the LPFT magnitude spectra.

2.1 Property of LPFT

The log-polar transform is rotation and scaling invariant as rotation and scaling in the image domain corresponds to translational shift in the LPT domain. Denoting the origin in the image domain (x_0, y_0) and the coordinates of any pixel (x, y), we have the log-polar coordinates:

$$\rho = \log \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{1}$$

$$\theta = \arctan\left(\frac{y - y_0}{x - x_0}\right), \text{when } x \neq x_0 \tag{2}$$

where ρ is logarithm of the radial distance from the origin and θ the polar angle. Essentially, by applying a log-polar transform to an image, concentric circles in the image are mapped to parallel lines in the LPT domain. We will use this property to detect copy-move forgery.

Consider a source image **S** and its rotated-scaled replica **R** with a rotation angle θ_0 and scaling factor λ_0. Let f and g represent luminance values in **S** and **R** respectively. Thus,

$$f(x_R, y_R) = g(x_R, y_R) \tag{3}$$

where pixel locations in the two domains have the following relation:

$$\begin{pmatrix} x_R \\ y_R \end{pmatrix} = \lambda_0 \begin{pmatrix} \cos\theta_0 & -\sin\theta_0 \\ \sin\theta_0 & \cos\theta_0 \end{pmatrix} \begin{pmatrix} x_S \\ y_S \end{pmatrix} \tag{4}$$

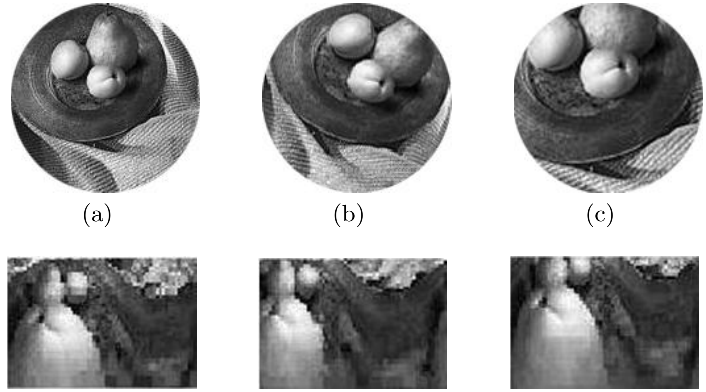
Now we compute LPT of **S** and **R** to produce $\mathbf{S}_{LP}$ and $\mathbf{R}_{LP}$ with pixels denoted, for simplicity, by $S(\rho_S, \theta_S)$ and $R(\rho_R, \theta_R)$ respectively. From the definition of LPT, we can easily obtain:

$$\begin{cases} \rho_R = \rho_S + \log\lambda_0 \\ \theta_R = (\theta_S + \theta_0) \bmod 2\pi \end{cases} \tag{5}$$

Therefore, in the log-polar coordinates, the relation between $S(\rho_S, \theta_S)$ and $R(\rho_R, \theta_R)$ can be established:

$$R(\rho_R, \theta_R) = S(\rho_S + \log\lambda_0, \theta_R + \theta_0) \tag{6}$$

Eq. (6) indicates that **R** is a translated replica of **S** in the log-polar domain. Fig. 2 shows an example in which rotation and scaling in the Cartesian coordinates are converted into translation in the LPT domain. The abscissa and ordinate in the log-polar coordinates correspond to θ and ρ, respectively. The original image in Fig. 2(a) is rotated by 37° , and scaled by a factor of 1.2 and then cropped to keep the same size, see Fig. 2(b). Their LPT versions are shown in Figs. 2(d) and 2(e). The two LPT versions differ by a translational shift. Note that, as the image is enlarged, a part of the image in the log-polar domain is moved out of the display area and another part moved in. Fig. 2(c) is another version of



(a) (b) (c)

(d) (e) (f)

Fig. 2. Image rotation and scaling: (a) original, (b) rotated by 37° and scaled by 1.2, (c) scaled by 1.5, (d) LPT of the original, (e) LPT of the rotated-scaled image, and (f) LPT of the image scaled by 1.5

Fig. 2(a) with a scaling factor 1.5 and the same size cut, and its LPT result is shown in Fig. 2(f). Compared with Fig. 2(d), almost one-third of the content of Fig. 2(f) has changed. Scaling with a larger factor, e.g., greater than 1.5, introduces too much change of the image content in the LPT domain, therefore is not considered in the present work.

Take Fourier transform of both sides of Eq. (6):

$$F_{\mathrm{R}}\left(u,v\right)=F_{\mathrm{S}}\left(u,v\right)\exp\left[2\pi j\left(u\log\lambda_0+v\theta_0\right)\right] \tag{7}$$

and define the normalized cross-spectrum of F_{R} and F_{S} as:

$$G\left(u,v\right)=\frac{F_{\mathrm{R}}\left(u,v\right)F_{\mathrm{S}}^{*}\left(u,v\right)}{\left|F_{\mathrm{R}}\left(u,v\right)F_{\mathrm{S}}^{*}\left(u,v\right)\right|} \tag{8}$$

The asterisk indicates complex conjugate. Since $|F_{\mathrm{R}}|=|F_{\mathrm{S}}|$ as can be seen from (7), $G(u,v)$ is a two dimensional complex sinusoid:

$$G\left(u,v\right)=\exp\left[-2\pi j\left(u\log\lambda_0+v\theta_0\right)\right] \tag{9}$$

We know that Fourier transform of a sinusoidal function is a delta function. Therefore, a peak would be present in the Fourier transform, $F_G(u,v)$, if $\mathbf{R}$ is a rotated-scaled replica of $\mathbf{S}$. For example, Fig. 3 shows $F_G(u,v)$ of Figs. 2(a) and 2(b).

2.2 Copy-Move Forgery Detection

We confine our discussion to gray scale images in this work. For color images, take the luminance component of the image's YCbCr representation. The proposed forgery detection steps are:

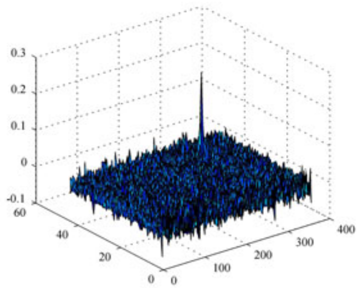


Fig. 3. $F_G(u, v)$ of the two images in Figs. 2(a) and 2(b), x and y axes correspond to u and v respectively, and z axis represents the magnitude value of $F_G(u, v)$

1. Divide the $m \times n$ image $\mathbf{I}$ such as the one shown in Fig. 4(a) into square blocks sized $2r$-by-$2r$, and take the inscribed circle of each block. Denote the circular region as $S(i, j; x, y, z)$ where (x, y) is the center, and r the radius:

$$S\left(i, j; x, y, r\right) = I(x - r + j, y - r + j), \ \sqrt{\left(x - x_0\right)^2 + \left(y - y_0\right)^2} \leq r \qquad (10)$$

The ranges of x and y are

$$\begin{cases} r \leq x \leq m - r \\ r \leq y \leq n - r \end{cases} \qquad (11)$$

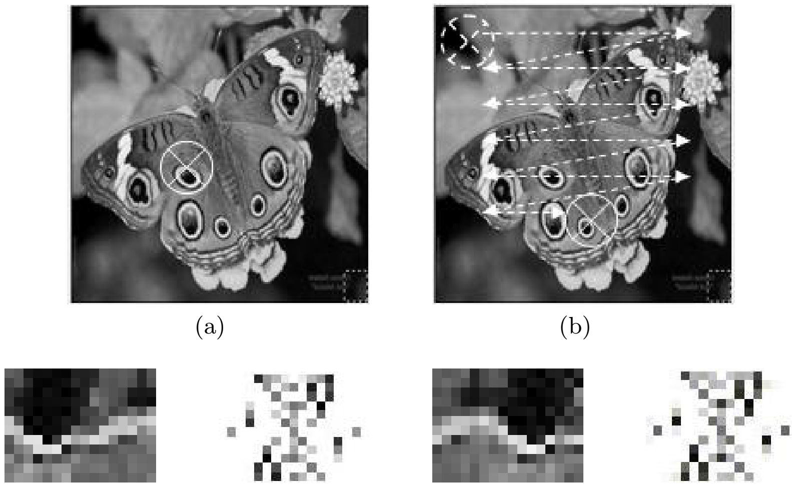
Since x and y have $m - 2r + 1$ and $n - 2r + 1$ different values respectively, there are a total of $K = (m - 2r + 1)(n - 2r + 1)$ circles as sketched in Fig. 4(b).

2. Compute LPT of each circular region, and Fourier-transform the LPT result. The K results of LPFT are denoted as $\{\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_K\}$, see examples in Figs. 4(c)-(f).

3. Compute normalized cross-spectrum $G(u, v)$ between each pair of $\mathbf{V}_k$, and find the maximum, resulting in a total of $L = K \times (K - 1)/2$ maximal values $\{p_1, p_2, \ldots, p_L\}$, each corresponding to a pair of circles in $\mathbf{I}$.

4. Compare the members in $\{p_1, p_2, \ldots, p_L\}$ with a threshold T chosen as the average absolute value of $F_G(u, v)$. If $p_l(l = 1, 2, \ldots, L)$ is greater than T, the two corresponding image blocks in $\mathbf{I}$ are regarded as belonging to copy-move regions.

The best choice of radius r of the circular region is related to the size of tampered region. Without any *a priori* knowledge of the image forgery, we can choose the r value according to the image size. In this work, test images are 512×512 or smaller. We let radius r be 10 pixels that is a compromise between false alarm and correct detection. Detection of any two image regions with overlapped parts is excluded to avoid false alarm caused by repeated detection of the same part in different image regions.



(a) (b)

(c) (d) (e) (f)

Fig. 4. Circular regions in the test image and the LPFT: (a) a circle in the image, (b) scanning over the circular regions, (c) LPT of one circle (source), (d) magnitude of LPFT of the source circle, (e) LPT of the rotated-scaled replica, and (f) magnitude of LPFT of the replica

3 Experimental Results

From the above discussion, there is a peak in $F_G(u,v)$ of two image regions when one is a replica of the other, as shown in Fig. 3. To set the threshold T, we perform a series of experiments for different T values from 0.1 to 0.5 with an increment 0.05. The optimal value turns out to be between 0.3 and 0.35. We normalize $F_G(u,v)$ and let $T = 0.3$ in the following experiments. If the average value of normalized $F_G(u,v)$ is less than T, which means that a peak value exists in the normalized $F_G(u,v)$, the two corresponding image regions are considered in copy-move areas, and marked with red squares.

Figs. 5(a) and 5(b) present a case of copy-move forgery with rotation. The lighter in Fig. 5(a) is rotated by 70° and pasted to another part of the image, as shown in Fig. 5(b), with the detection result in Fig. 5(c). In Fig. 6, the ball is duplicated with a scaling factor 0.67. Fig. 7 shows a rock being scaled by 0.8, copied and pasted to the bottom-left corner. Fig. 8 presents an example in which the pasted skater is taken from the image, rotated by 4° and enlarged by 1.2, and then pasted in the bottom-right corner. All the copied-moved objects and their source regions are correctly identified. In Fig. 9, the detection result of Fig. 1(b) is given, together with horizontal sections of normalized $F_G(u,v)$ with peaks indicating the copy-move operation.

(a) (b) (c)

Fig. 5. Detection of a rotated-copied lighter: (a) original, (b) tampered with the lighter rotated by 70° and pasted to another region, and (c) detection result

(a) (b) (c)

Fig. 6. Detection of a scaled-copied ball: (a) original, (b) tampered with the ball shrunk by 0.67 and pasted, and (c) detection result

(a) (b) (c)

Fig. 7. Detection of a scaled-copied rock: (a) original, (b) tampered with the rock shrunk by 0.8 and pasted, and (c) detection result

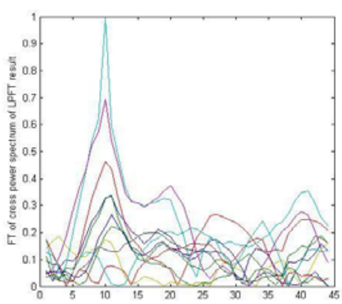

(a)


(b)


(c)

Fig. 8. Detection of a rotated-scaled-copied skater: (a) original, (b) tampered with the skier rotated by 4 degrees, enlarged by 1.2 and pasted, and (c) detection result


(a)



(b)

Fig. 9. Detection of the pasted balloon as in Fig. 1: (a) detection result, and (b) horizontal sections of normalized FT of cross spectrum of LPFT corresponding to tampered areas

4 Conclusions

We have proposed an effective method to detect copy-move image forgery with rotation and scaling tolerance. LPFT is employed to convert scaling and rotation in the image domain into translation in the log-polar domain, and peaks in the cross-spectra of LPFT pairs reveal similarity in the corresponding pairs of image blocks. This way, tampered areas in an image produced by copy-move operations, even accompanied with rotation and scaling, can be identified. Unlike FMT features in [8], the proposed LPFT is robust to large rotation, and experimental results show the efficacy of our method.

The next concern is to cope with more challenging cases containing scaling, rotation, re-sampling, cropping and other complicated post-manipulations.

Acknowledgments

This work was supported by the Natural Science Foundation of China (60773079, 60872116, and 60832010).

References

1. Fridrich, J., Soukal, D., Lukas, J.: Detection of copy-move forgery. In: Proc. of Digital Forensic Research Workshop, pp. 178–183 (2000)
2. Popescu, A.C., Farid, H.: Exposing digital forgeries by detecting duplicated image regions. Dept. Comput. Sci., Dartmouth College, Tech. Rep. TR2004-515 (2004)
3. Mahdian, B., Saic, S.: Detection of copy-move forgery using a method based on blur moment invariants. Forensic Sci. Int. 171, 180–189 (2007)
4. Myrna, A., Venkateshmurthy, M., Patil, C.: Detection of region duplication forgery in digital images using wavelets and log-polar mapping. In: Proc. of International Conference on Computational Intelligence and Multimedia Applications, pp. 371–377 (2007)
5. Bravo, S., Nunez, A.: Passive forensic method for detecting duplicated regions affected by reflection, rotation and scaling. In: Proc. of European Signal Processing Conference, pp. 824–828 (2009)
6. Zokai, S., Wolberg, G.: Image registration using log-polar mappings for recovery of large-scale similarity and projective transformations. IEEE Trans. Image Processing 14(10), 1422–1434 (2005)
7. Lin, C.-Y., Wu, M., Bloom, J., Cox, I., Lui, Y.: Rotation, scale, and translation resilient watermarking for images. IEEE Trans. Image Processing 10(5), 767–782 (2001)
8. Bayram, S., Sencar, H., Memon, N.: An efficient and robust method for detecting copy-move forgery. In: Proc. of International Conference on Acoustics, Speech, and Signal Processing, pp. 1053–1056 (2009)