A Keypoint-Based Region Duplication Forgery Detection Algorithm

Mahmoud EMAM†,††, Nonmember, Qi HAN†(a), Member, Liyang YU†, and Hongli ZHANG†, Nonmembers

SUMMARY  The copy-move or region duplication forgery technique is a very common type of image manipulation, where a region of the image is copied and then pasted in the same image in order to hide some details. In this paper, a keypoint-based method for copy-move forgery detection is proposed. Firstly, the feature points are detected from the image by using the Förstner Operator. Secondly, the algorithm extracts the features by using MROGH feature descriptor, and then matching the features. Finally, the affine transformation parameters can be estimated using the RANSAC algorithm. Experimental results are presented to confirm that the proposed method is effective to locate the altered region with geometric transformation (rotation and scaling).

key words: copy-move, forgery detection, Förstner Operator, MROGH descriptor

1. Introduction

Digital images are more frequently used to make important decisions. This is more obvious in the area of forensics, where digital images can be used to describe the scene of a crime and therefore provide critical evidence for innocence or accusation. Many highly sophisticated image editing softwares exist, which allow anyone to easily modify images without leaving any subtle traces of forgeries. The most commonly used way to tamper with images, known as “Copy-move” forgery technique, it can be done easily by copying a part of the image itself and pasting it into another part in the same image. Figure 1 shows an example of Copy-move forgery, where the child’s shirt pocket is copied from his left hand side and then moved to the other side. Counterfeits may perform some post-processing operations on the forged images in order to gild some details, which makes the task of detecting forgery significantly harder. The post-processing operations includes for example; noise addition, blurring, JPEG compression, geometrical transformations.

A number of copy-move forgery detection (CMFD) methods have been proposed to solve this problem [1]. Generally, these methods are classified into two main categories: block-based methods and keypoint-based methods (or visual feature-based methods) [1]. Keypoint-based methods extract feature points and use local features to identify duplicated regions instead of using blocks. The algorithms based on block-matching are computationally complex. Meanwhile, keypoint-based methods can effectively overcome the main issues such as computational complexity and robustness. So, here in our paper we are only concerned with the keypoint-based methods. In order to detect the forgery with geometrical transformations, many methods used Scale Invariant Feature Transform (SIFT) algorithm [2] to detect local invariant features of the image, then search the matched feature points by SIFT feature matching to detect the forgery. In the first robust CMFD method based on SIFT features [3], there was no estimation for the geometric transformation parameters rather than the weak performances. Another preliminary method [4] has been proposed, where geometric distortions are modeled as affine transforms of pixel coordinates. These affine transformation parameters are estimated by means of the Random sample consensus (RANSAC) algorithm [5]. RANSAC algorithm can estimate the parameters with a high degree of accuracy even if a significant number of mismatched pairs are present [5]. In general, SIFT and SURF detectors are used to detect the keypoints [1], and the related local image descriptors are used to find matches between these points. To eliminate the false matches and estimate the geometric transformation applied, RANSAC algorithm [5] can be used.

In this paper, Förstner operator (FOP) [6] is used for detecting local features from the forged image. FOP can be used to detect the distinct points, corners and centers of the circular features. After detecting the local features from the image, these keypoints are described and then matched. The matched points are then clustered according to the distance between them. Afterwards, RANSAC algorithm can be used to estimate the affine transformations that the forged regions have undergone and then to remove the false alarms.

This paper proceeds like this: in Sect. 2, our proposed method is detailed; in Sect. 3, experimental results are presented and discussed; finally, in Sect. 4, the paper is con-
2. Proposed Method

Figure 2 illustrates a flow diagram of the proposed method, and generally it can be divided into three main parts. Firstly, the Förstner Operator (FOP) [6] is used to initially detect out the keypoints from the input image. Afterward, the keypoints are then described by using the MROGH descriptor [7]. Finally, the feature matching determines whether points are then described by using the MROGH descriptor [7].

2.1 Local Visual Features Detection

A more detailed description for many visual features detecting methods has been presented [8]. In our proposed method, Förstner operator (FOP) is used for detecting keypoints from the forged image. FOP is a local feature detector proposed by Förstner et al. [6]. It investigates a local window in the image of a given size, having locally maximal localization accuracy, and determines the average change of intensity which results from shifting the window by a few pixels in various directions. This is achieved by evaluating the second moment matrix Eq. (1):

$$M_{\sigma} = \nabla_{x,y} \nabla_{x,y}^T = G_{\sigma} \ast \begin{bmatrix} g^2_{x,\tau} & g_{x,\tau} g_{y,\tau} & g^2_{y,\tau} \end{bmatrix}$$

where the gradient $\nabla_{x,y} = [g_{x,\tau}, g_{y,\tau}]^T$ is determined first by using the differentiation scale $\tau$ and rotationally symmetric Gaussian function $G_{\sigma}$ with standard deviation $\sigma$. Only the locations of the selected optimal windows within the image are taken into consideration.

2.2 Feature Point Description

Several different keypoint descriptors have been proposed [9]. In our proposed method we used Multi-support Region Order-based Gradient Histogram (MROGH) [7]. MROGH descriptor is a 2D histogram by aggregating gradient distributions into intensity orders, which not only contains gradient information but also information about relative relationship of intensities as well as spatial information [7]. It also rotation invariant without estimating the dominant orientation which is a major error source of most descriptors [7].

Given a set of detected keypoints $P = p_1, p_2, \ldots, p_n$, we need to generate a descriptive vectors $F_i, i = 1, 2, \ldots, n$ for each keypoints. Firstly, the gradient is computed in a rotation invariant way from the support regions. Secondly, the quantified gradient for each point in a given region is computed based on a number $\omega_1$ of quantifiable levels, which can achieve a rotation invariant representation. Thirdly, all points in the support region are sorted according to their intensities and then partitioned to a number $\omega_2$ segments according to their orders. Finally, gradient information are pooled together in each segment. Then, the histogram of gradient can be used for each segment. Afterwards, a 2D MROGH histogram with length $\omega_1 \times \omega_2 \times \omega_3$ can be obtained where $\omega_3$ is the number of support regions. We empirically choose $\omega_1 = 8$, and $\omega_2 = 6$ because they can achieve a good performance as presented by experiments [7].

2.3 Feature Matching

For each feature $f_j \in F; j = 1, 2, \ldots, \omega_1 \times \omega_2 \times \omega_3$, we first used $kd-tree$ to obtain the $k$ nearest neighbors $N_l, l = 1, 2, \ldots, k$ with corresponding distances denoted as $D_l, l = 1, 2, \ldots, k$ that represents the sorted Euclidean distance. As proposed in [2], the keypoint is matched if the ratio between $D_1$ and $D_2$ (first and second closest neighbor) is less than a threshold $(D_1/D_2 <\text{threshold})$. However, this matching method is unable to deal with multiple keypoint matching. So, we used another matching procedure which presented in [10] and denoted as $g2NN$ method. This method iterating the nearest neighbor test between $D_{l}/D_{l+1}$ until this ratio is greater than $g2NN\text{threshold}$, i.e.:

$$D_l/D_{l+1} < g2NN\text{threshold}$$

Hence, we obtained the set of all matched points. These matched points are then saved for further processing and the other isolated keypoints are removed.

2.4 Post-Processing

In this stage, the matched points are clustered according to the distance between them based on a distance threshold $Dthr$. Then, all the clusters with members less than a minimum member number $\zeta$ in each cluster are removed. For the remaining clusters, we used RANSAC algorithm to estimate the affine transformations that the forged regions have undergone and then to remove false alarms. For each estimated homography matrix, we find all inliers $D$ less than $\alpha$ that fit with this transformation according to:

$$D = \left\| H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} - \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \right\|$$

where $(x, y, 1)^T$ and $(x', y', 1)^T$ are the homogeneous coordinates of a pair of matched points and $H$ is the estimated affine homography matrix that can be defined as follows:

$$H = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$
Here, we can get some false alarms. To remove it, we applied distance-based clustering again for each homography whose corresponding inlier pairs are less than $\gamma$. Then, all the clusters with members less than $\zeta$ in each cluster are removed. Finally, we applied some morphological operations to get the final detection of the duplicated regions.

3. Experiments and Discussion

3.1 Dataset and Parameters Setup

In this section, we evaluate the performance of the proposed method by conducting a series of experiments. The image dataset [11] which is available at http://www.dicgim.unipa.it/cvip/ is used. This dataset is subdivided into several subsets (D0, D1 − 2, D3). The dataset has 970 images in total, which are realistic copy-move forgeries. We compared the performance of our proposed method with two different existing CMFD methods; ZERNIKE method [12] which is more common and robust to geometric transforms [1] and SIFT vertex method [11] which is the best algorithm introduced by Ardizzone et al. [11]. To evaluate the performance of our method, precision-recall (PR) curves and $F_1$ score are employed, Eq. (3):

$$P = \frac{T_P}{T_P + F_P}, R = \frac{T_P}{T_P + F_N}, F_1 score = 2 \times \frac{P \times R}{P + R}$$

where; $T_P$ is the number of tampered pixels, which are classified as tampered. $F_P$ is the number of authentic pixels, which are classified as tampered. $F_N$ is the number of tampered pixels, which are classified as authentic.

We set up the parameters of the proposed algorithm as follows: the number of support regions $\omega$ in MROGH is set to 1, the number $\omega_1$ of quantifiable levels is set to 8 orientation bins, number of order segments $\omega_2$ is set to 6, 12 nearest neighbors are taken for $kd$ tree method, $g2NNthreshold$ in Eq. (2) is set to 0.8, the distance threshold $Dth$ is set to 100, $\zeta$ is set to 6, $\alpha$ is set to 4, and $\gamma$ is set to 8.

3.2 Detection Performance

3.2.1 Plain Copy-Move Forgery

The proposed method is applied first to D0 to evaluate the performance of our method against plain Copy-move forgery. Some detected examples can be shown in Fig. 3, in which all forged images are correctly detected by the proposed method. The precision, recall, and $F_1$ score of the proposed, SIFT vertex, and ZERNIKE are: 99.51%, 46.27%, 63.17%; 99.75%, 21.5%, 35.38%; and 93.8%, 74.8%, 83.23%; respectively. This means that the proposed method and SIFT vertex are very accurate in finding the correct matches (very slight false positives) compared with ZERNIKE. Additionally, they are not able to cover all the areas of the Copy-move region (more false negatives exist) but, our method still better than SIFT vertex (more than twice recall rate). Furthermore, in our experiment we found that we can observe the forgery in the images and can be easily identifiable, even when the forged regions are not detected correctly, see Fig. 4 (d) for more details.

3.2.2 Robustness against Rotation Manipulations

In the $D1 − 2$ subset, the cloned regions are rotated before being pasted by different angles according to the following: rotation in the range of $[-5 : 5]$ with step 1, rotation in the range of $[-25 : 25]$ with step 5, and rotation in the range of $[0 : 330]$ with step 30. A visual example for the detection results of the proposed method can be shown in Fig. 5. Table 1 compares the performance evaluation results of the proposed method, SIFT vertex and ZERNIKE for all Copy-Rotate-move forgery images. We can observe that the proposed method achieves a good performance against rotation manipulations. We also noticed that, when rotation is made by a large angle, the matched points decrease due to the impact of rotation, but there are still enough matched points to be detected (see for example Fig. 5 (d)).
3.2.3 Robustness against Scaling Manipulations

In this experiment, the forged regions are resized by a different scaling factors $s$. We select $s = 0.95, 1, 1.05,$ and $2$ to explore the performance of our method against scaling manipulation. As shown in Fig. 6, the matched points decrease as $s$ increases. Table 2 shows the performance evaluation results of the proposed method, SIFT vertex, and ZERNIKE against some selected scaling factors from the subset $D_1 – D_2$. We can conclude that our method is of robustness to small scaling factors only. This is due to the FOP is not able to detect sufficient keypoints from the forged image with a large scale (see for example Fig. 6 (d)).

4. Conclusion and Future Work

In this paper, we proposed a keypoint-based method to detect copy-move forgery. We used Förstner Operator as a keypoint detector and MROGH as a local image descriptor. The experimental results show the robustness of our method, especially in the textured regions. But it still needs more improvements in the poorly textured images. As most of the keypoint-based methods, our method cannot be used if no keypoint is detected, e.g. if non-textured regions are used to conceal the objects in the images. In the future, we will try to solve this problem by using an affine-covariant feature detector. Also, we will try to investigate some other post-processing techniques, to recover some missing matches and hence increase the recall rate of our method.

Acknowledgments

The work is supported by National Natural Science Foundation of China (Grant No:61471141, 61301099, 61361166006), Fundamental Research Funds for the Central Universities (Grant No: HIT.KISTP.201416, HIT.KISTP.201414)

References