

# Detection of Copy-Scale-Move Forgery in Digital Images Using SFOP and MROGH

Mahmoud Emam<sup>1,2(✉)</sup>, Qi Han<sup>1(✉)</sup>, and Hongli Zhang<sup>1</sup>

<sup>1</sup> School of Computer Science and Technology,  
Harbin Institute of Technology, Harbin 150080, China  
ma7moud\_emam@yahoo.com, {qi.han,zhanghongli}@hit.edu.cn

<sup>2</sup> Department of Mathematics, Faculty of Science,  
Menoufia University, Shebin El-koom 32511, Egypt

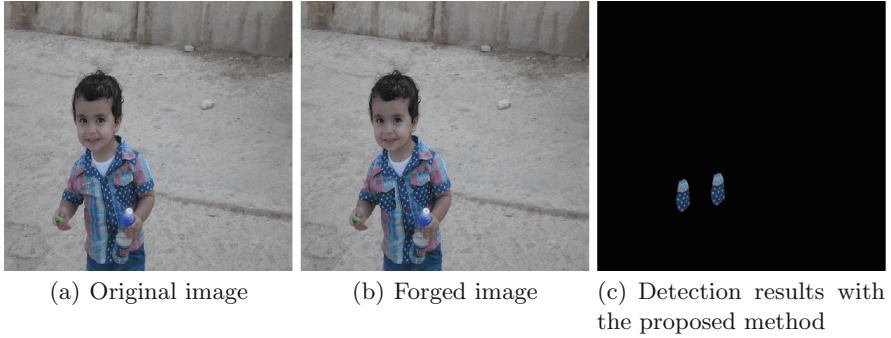
**Abstract.** Social network platforms such as Twitter, Instagram and Facebook are one of the fastest and most convenient means for sharing digital images. Digital images are generally accepted as credible news but, it may undergo some manipulations before being shared without leaving any obvious traces of tampering; due to existence of the powerful image editing softwares. Copy-move forgery technique is a very simple and common type of image forgery, where a part of the image is copied and then pasted in the same image to replicate or hide some parts from the image. In this paper, we proposed a copy-scale-move forgery detection method based on Scale Invariant Feature Operator (SFOP) detector. The keypoints are then described using MROGH descriptor. Experimental results show that the proposed method is able to locate and detect the forgery even if under some geometric transformations such as scaling.

**Keywords:** Image forensics · Copy-move · Forgery detection · Scale invariant feature · RANSAC · MROGH descriptor

## 1 Introduction

Due to the existence of highly sophisticated software for editing the digital images, it became easily modify images without leaving any subtle traces. Copy-move forgery technique is the most commonly used technique where, a part of the image is copied and then pasting it into another part in the same image. Therefore, Copy-move forgery detection (CMFD) algorithms aims at detecting the same or similar regions in the forged images. Figure 1 shows an example of Copy-move forgery, where the pocket of the child's shirt is copied from his left hand side and then pasted into the other side of the shirt. Some post-processing operations can be performed on the forged images after Copy-move operation, which makes the task of forgery detection more harder. Typically, post-processing operations are applied to cover up the forgery such as geometric transformation (e.g. scaling).

Several researchers have introduced algorithms for detecting image copy-move forgery which can be found in these surveys [1, 6]. Generally, these methods can be classified into two main categories: block-based methods [14] and



**Fig. 1.** Copy-move forgery example

keypoint-based methods [2]. Due to the limitations of block-based methods especially in the robustness against scaling manipulations and time complexity, keypoint-based methods attract many researcher's attention. Keypoint-based methods detect keypoints and then use the local features to identify duplicated regions instead of using overlapping blocks [15]. Typically, SIFT [13] and SURF [3] are used as a keypoint detectors, and their corresponding descriptors are used to find matches between these keypoints. To estimate the geometric transformations applied to the forged regions, Random sample consensus (RANSAC) algorithm [9] can be used.

Huang et al. [12] introduced a method to detect copy-move forgery based on local statistical features, known as scale invariant feature transform (SIFT). But in that method there is no estimation for the geometric transformation parameters rather than the weak performances. Another method has been proposed in [15], but that method can't manage affine transformation. Bo et al. [5] presented another CMFD method based on Speeded Up Robust Feature (SURF) descriptor to overcome geometric operations such as scaling, their experimental results were introduced visually. The algorithm appears to be promising, but it still need more improvements to automatically localize the duplicated regions. The existing CMFD methods mentioned in [1, 6] can achieve an acceptable performance, but still there exist some challenges especially in the affine transformation manipulations (e.g. scaling).

In this paper, Scale Invariant Feature Operator (SFOP) detector [11] is used as a local feature detector to extract the keypoints from the forged image. After detecting the keypoints from the image, these keypoints are described using MROGH descriptor and then matched. The matched points are then clustered according to the distance between them. Then, RANSAC algorithm can be used to estimate the affine transformation parameters and remove the false matches. The rest of this paper proceeds like this: in Sect. 2, each step of the proposed method is explained; in Sect. 3, experimental results are presented and discussed; finally, Sect. 4 summarizes the paper and next research target.

## 2 Proposed Method

### 2.1 Local Features Detector and Descriptor

In our proposed method, SFOP detector is used for detecting keypoints from the forged image. The SFOP, which is a scale-space extension of the detector proposed by Förstner [10], is a local feature detector proposed by Förstner et al. in [11]. SFOP uses the general spiral feature model of [4] to unify different types of features within the same framework, and it achieves a better coverage under various geometric transformations than the other local feature detectors [7].

Given a forged image  $I_{forged}$ , Firstly we applied SFOP to detect the keypoints  $P = p_1, p_2, \dots, p_n$ . Secondly, we used Multi-support Region Order-based Gradient Histogram (MROGH) [8] to generate a descriptive vectors  $F_i, i = 1, 2, \dots, n$  for each keypoint  $p_i \in P, i = 1, 2, \dots, n$ . A two-dimensional MROGH histogram with length  $\lambda_1 \times \lambda_2 \times \lambda$  can be obtained where;  $\lambda$  is the number of support regions,  $\lambda_1$  is the number of quantifiable levels, and  $\lambda_2$  is the number of order segments. We empirically choose  $\lambda_1 = 8$ , and  $\lambda_2 = 6$  because they can achieve a good performance as presented by experiments in [8].

### 2.2 Feature Matching

For each feature  $f_j \in F; j = 1, 2, \dots, 48$ , we used *kd-tree* to obtain the  $k$  nearest neighbors  $N_l, l = 1, 2, \dots, k$  with corresponding distances denoted as  $d_z, z = 1, 2, \dots, k$  that represents the sorted Euclidean distance. The keypoints are then matched if the ratio between  $D_1$  and  $D_2$  is less than a threshold ( $D_1/D_2 < thr$ ). But, this matching strategy can't deal with multiple keypoint matching. So, we used another matching procedure *g2NN* as presented in [2]. This method iterates the nearest neighbors test between  $D_r, D_{r+1}$  while:

$$D_r/D_{r+1} < g2NN_{thr} \quad (1)$$

Now, we obtain the set of all matched points. These matched points are then kept for further post-processing and the other mismatched keypoints are then removed.

### 2.3 Post-processing

In this step, the matched keypoints are clustered according to the distance between them based on a threshold  $D_{thr}$ . After that, all the clusters with members less than a minimum member number  $\zeta$  in each cluster are discarded, for the others, we used RANSAC algorithm [9] to estimate the affine transformation parameters and remove the false alarms. For each estimated homography matrix, we find all inliers  $D$  less than  $\alpha$  that fit with this transformation according to:

$$D = \left\| H \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} - \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \right\|_2$$

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

$$H = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Hence, we can get some false alarms. To remove it, we again used distance-based clustering 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 detected duplicated regions.

### 3 Experimental Results and Discussion

#### 3.1 Dataset

We evaluate the performance of the proposed method by conducting a series of experiments. In the following experiments, we have used benchmark evaluation database appeared in [6]. The dataset contains 48 high resolution color images of different sizes, varying from  $533 \times 800$  (*giraffe image*) to  $3900 \times 2613$  (*sailing image*). The tampered images have been generated by cutting and pasting image region(s). The image region(s) selected for duplication can be geometrically transformed before being pasted. The duplicated region can vary in size (e.g., small, medium, or large). A sample of original images, forged images and its ground truth indicating the forged regions are shown in Fig. 2.

To evaluate the performance of the proposed method, precision-recall (PR) curves [16] and  $F_1$  score are employed. Equations 2, 3, and 4 show how the precision, recall, and  $F_1$  rates are calculated;

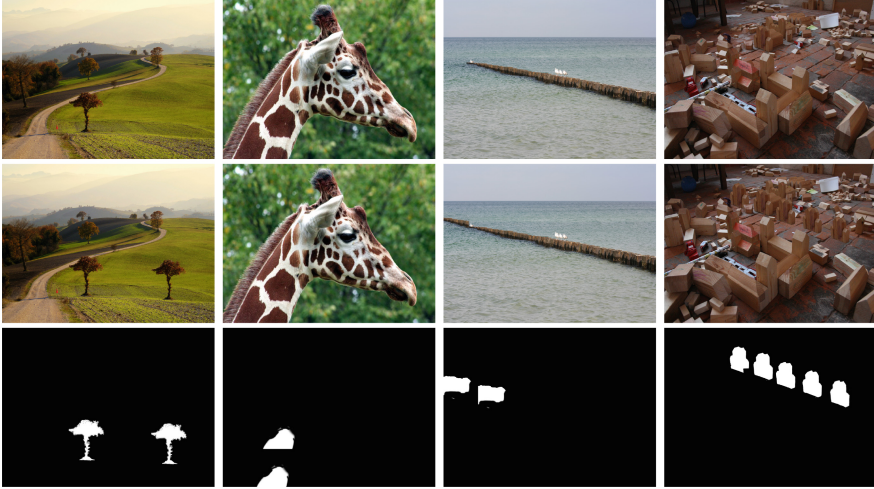
$$Precision = \frac{T_P}{T_P + F_P} \tag{2}$$

$$Recall = \frac{T_P}{T_P + F_N} \tag{3}$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{4}$$

where;

- $T_P$  (True Positive) represents the number of tampered pixels, which are classified as tampered.
- $F_P$  (False Positive) represents the number of authentic pixels, which are classified as tampered.
- $F_N$  (False Negative) represents the number of tampered pixels, which are classified as authentic.



**Fig. 2.** Examples for copy-move forgery: the up row is the original images, middle row is the corresponding forged images, and bottom row is the ground truth map

### 3.2 Parameters Setup

We set up the parameters of the proposed method as in Table 1.

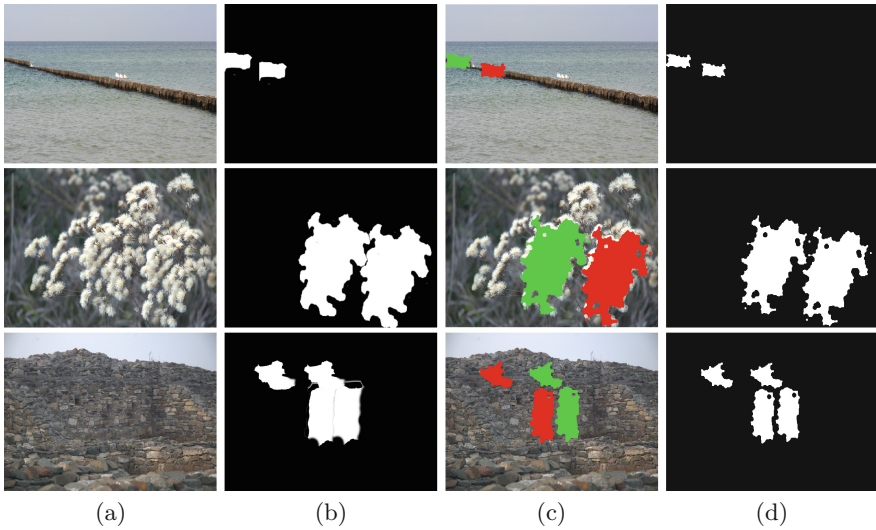
**Table 1.** Parameters setup for the proposed method

Parameters	Description	Value
$g2NN_{thr}$	ratio threshold for collecting the matched features	0.8
$D_{thr}$	the distance threshold	120
$k$	the number of nearest neighbors in $kd-tree$	8
$\zeta$	minimum member number threshold for each cluster	3
$\alpha$	distance threshold for collecting the inliers	4
$\lambda$	the number of support regions in MROGH	1
$\lambda_1$	the number of quantifiable levels	8
$\lambda_2$	the number of order segments	6

### 3.3 Detection Performance of the Proposed Method

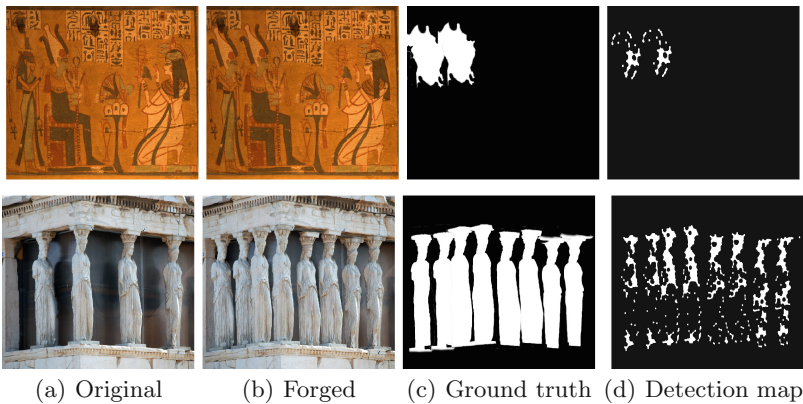
**Plain Copy-move Forgery.** We evaluate how the proposed method can perform under plain Copy-move forgery without any post-processing operations. Figure 3 shows some visual detected examples, in which the forged regions are correctly localized by the proposed method. All images in the dataset are detected and the values of precision, recall, and  $F_1$  score are computed.

The proposed method has a very high precision rate (exactly, 95.66%), an acceptable recall rate value (exactly, 49.12%), and the comprehensive assessment



**Fig. 3.** Some visual results: (a)the tampered images, (b)the corresponding binary mask, (c)the correctly detected matches, (d)the detection map produced by our proposed method

$F_1$  score equal to 64.91%. Therefore, the proposed method is very accurate in finding the correct matches (very slight false positives). But, it is not able to cover all the areas of the Copy-move region (more false negatives exist). 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 as shown in Fig. 4.(d).

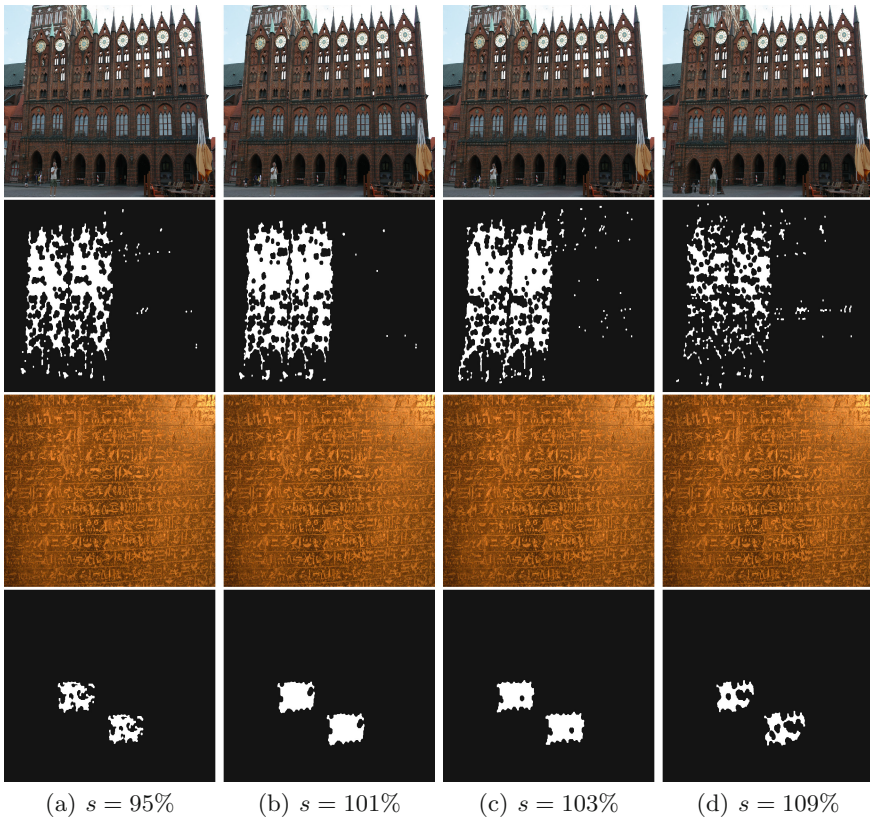


**Fig. 4.** An example of Copy-move forgery detection (proposed method)

**Robustness against Scaling Manipulations.** In the benchmark dataset, the cloned regions are scaled before being pasted by different scaling factors  $s$ . To evaluate the robustness of our proposed method against scaling manipulations, We test the performance of our method under 10 different scaling factors of the original size of the forged region ( $s = 91\%, 93\%, 95\%, 97\%, 99\%, 101\%, 103\%, 105\%, 107\%$ , and  $109\%$ ). The detection results of some forged images with different scaling factors  $s$  are shown in Fig. 5.

Table 2 shows the performance evaluation results of the proposed method against scaling manipulations for all Copy-Scale-move forgery images in the benchmark dataset.

We noticed that the proposed method achieves a good performance against scaling manipulations. We also noticed that, when scaling factor increase, the matched points decrease due to the impact of scaling, but there are still enough matched points to be detected (see for example Fig. 5(d) and Table 2 when  $s$  is more than 101%).



**Fig. 5.** The detection results with different scaling factors  $s$

**Table 2.** Detection Performance of the proposed method against scaling manipulations

(%)	$s = 91$	$s = 93$	$s = 95$	$s = 97$	$s = 99$	$s = 101$	$s = 103$	$s = 105$	$s = 107$	$s = 109$
Precision	93.70	96.55	96.86	96.05	96.32	96.05	95.66	96.45	96.95	96.43
Recall	32.62	36.81	40.44	43.96	46.57	47.04	45.99	44.23	39.50	35.08
$F_1$ score	48.39	53.30	57.06	60.31	62.79	63.15	62.12	60.65	56.13	51.44

Hence, we can conclude that our method is of robustness to scaling manipulations, since the scale invariant feature operator (SFOP) is able to detect sufficient keypoints from the forged image with a different scaling factors. But, it still needs some improvements in the flat regions.

## 4 Conclusion

In this paper, a forensic method has been proposed to detect and localize copy-move regions under scaling manipulations. We used scale invariant feature operator (SFOP) as a feature points detector due to its scaling invariance. The proposed method is able to find the correct matches but, it is not able to cover all the regions of the Copy-move region due to the existence of more false negatives. Furthermore, our method can observe the forgery in the images and can be easily identifiable. The experimental results show the robustness of our method against scaling manipulations, especially in the non-flat regions. But, it still needs some improvements in the flat regions. In the future, we will try to solve this problem by using a dense interest point to find more matched keypoints. Also, we will try to use some other post-processing techniques, to recover some missing matches and hence increase the recall rate of the method.

**Acknowledgment.** The authors would like to thank all anonymous reviewers for their insightful comments. Additionally, This work is supported by the National Natural Science Foundation of China (Grant Number: 61471141, 61301099, 61361166006), the Fundamental Research Funds for the Central Universities (Grant Number: HIT. KISTP. 201416, HIT. KISTP. 201414).

## References

1. Al-Qershi, O.M., Khoo, B.E.: Passive detection of copy-move forgery in digital images: state-of-the-art. *Forensic Sci. Int.* **231**(1), 284–295 (2013)
2. 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. Inf. Forensics Secur.* **6**(3), 1099–1110 (2011)
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.* **110**(3), 346–359 (2008)
4. Bigu, J., et al.: A structure feature for some image processing applications based on spiral functions. *Computer Vis. Graph. Image Process.* **51**(2), 166–194 (1990)



5. Bo, X., Junwen, W., Guangjie, L., Yuewei, D.: Image copy-move forgery detection based on surf. In: 2010 International Conference on Multimedia Information Networking and Security (MINES), pp. 889–892. IEEE (2010)
6. Christlein, V., Riess, C., Jordan, J., Riess, C., Angelopoulou, E.: An evaluation of popular copy-move forgery detection approaches. *IEEE Trans. Inf. Forensics Secur.* **7**(6), 1841–1854 (2012)
7. Ehsan, S., Kanwal, N., Clark, A.F., McDonald-Maier, K.D.: Measuring the coverage of interest point detectors. In: Kamel, M., Campilho, A. (eds.) *ICIAR 2011, Part I*. LNCS, vol. 6753, pp. 253–261. Springer, Heidelberg (2011)
8. Fan, B., Wu, F., Hu, Z.: Aggregating gradient distributions into intensity orders: a novel local image descriptor. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2377–2384. IEEE (2011)
9. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **24**(6), 381–395 (1981)
10. Förstner, W.: A framework for low level feature extraction. In: Eklundh, J.-O. (ed.) *Computer Vision—ECCV 1994*. LNCS, vol. 801, pp. 383–394. Springer, Heidelberg (1994)
11. Förstner, W., Dickscheid, T., Schindler, F.: Detecting interpretable and accurate scale-invariant keypoints. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 2256–2263. IEEE (2009)
12. Huang, H., Guo, W., Zhang, Y.: Detection of copy-move forgery in digital images using sift algorithm. In: 2008 Pacific-Asia Workshop on Computational Intelligence and Industrial Application, PACIIA 2008, vol. 2, pp. 272–276. IEEE (2008)
13. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
14. Nathalie Diane, W.N., Xingming, S., Moise, F.K.: A survey of partition-based techniques for copy-move forgery detection. *Sci. World J.* **2014**, 13 (2014)
15. Pan, X., Lyu, S.: Detecting image region duplication using sift features. In: 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 1706–1709. IEEE (2010)
16. Powers, D.M.: Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation (2011)