Improving the Detection and Localization of Duplicated Regions in Copy-Move Image Forgery

Maryam Jaberi, George Bebis
Computer Science and Eng. Dept.
University of Nevada, Reno
Reno, USA
(mjaberi,bebis)@cse.unr.edu

Muhammad Hussain, Ghulam Muhammad
Computer Science Dept., Computer Eng. Dept.
King Saud University
Riyadh, Saudi Arabia
(mhussain, ghulam)@ksu.edu.sa

***Abstract*—Using keypoint-based features, such as SIFT features, for detecting copy-move image forgeries has yielded promising results. In this paper, our emphasis is on improving the detection and localization of duplicated regions using more powerful keypoint-based features. In this context, we have adopted a more powerful set of keypoint-based features, called MIFT, which share the properties of SIFT features but also are invariant to mirror reflection transformations. To improve localization, we propose estimating the parameters of the affine transformation between copied and pasted regions more accurately using an iterative scheme which finds additional keypoint matches incrementally. To reduce the number of false positives and negatives, we propose using "dense" MIFT features, instead of standard pixel correlation, along with hystereresis thresholding and morphological operations. The proposed approach has been evaluated and compared with competitive approaches through a comprehensive set of experiments using a large dataset of real images. Our results indicate that our method can detect duplicated regions in copy-move image forgery with higher accuracy, especially when the size of the duplicated region is small.**

Keywords: blind image forensics, copy-move image forgery, SIFT, MIFT, matching.

I. Introduction

Manipulating digital image contents in order to hide or create misleading images with no observable trace has appeared in many forms [1, 2]. Recently, there have been many research studies on improving image forgery detection [3]. In this study, our focus is on detecting one of these altering techniques named image cloning (copy-move). This tampering method creates a forged image by copying a certain portion of an image and moving it to another location of the same image [4]. The key characteristic of image cloning is that, since the duplicated region is picked from the image itself, the noise components, texture and color patterns are compatible with the rest of the image. Thus, it is not easy to detect the forgery parts [5]. Among the image forgery detection methods proposed in the literature, pixel-based approaches are the most popular; the key idea is exposing image tampering by analyzing pixel level correlations [6]. In general, pixel-based approaches can be classified into two categories: block matching [4, 7, 8, 2, 9, 10, 11, and 12] and feature matching [13, 14, 15, and 16]. The key idea behind feature matching methods is discovering and clustering similar parts in an image. The feature matching approaches presented in [13, 14, 15], employ local statistical features, known as Scale Invariant Feature Transform (SIFT) [16]. In these methods, very similar techniques were used to find corresponding features and potentially interesting areas. An affine transformation between matching regions was estimated using Random Sample Consensus (RANSAC) [17]. The method proposed by Pan and Lyu [14] includes a verification step, which tries to locate the duplicated regions by using the normalized correlation map and thresholding.

As shown in our experimental results, a weakness of Pan's method, as well as of similar methods [13, 15], is that they cannot localize the forged region very accurately. Moreover, these methods were evaluated on a relatively small number of real forged images. In this study, we improve copy-move forgery detection using keypoint-based features by focusing on the issue of accurate detection and localization of duplicated regions. Specifically, we have made several contributions in this work. First, we employ Mirror Reflection Invariant Feature (MIFT) features [18] instead of SIFT features to find similar regions in images. MIFT features share all properties of SIFT features but are also invariant to mirror reflection transformations. Second, since the quality of the affine transformation between copied and pasted regions is critical in localizing the duplicated region accurately, we refine the parameters of the affine transformation iteratively by finding additional keypoint matches incrementally. Third, to extract the duplicated region, we use dense MIFT features and apply hysteresis thresholding [19] instead of standard thresholding, and morphological operators to reduce false positives and negatives. We have evaluated the performance of the proposed methodology by performing a comprehensive set of experiments using a large database of real images (i.e., CASIA v2.0) [20]. Comparisons with competitive approaches show that the proposed method can detect duplicated regions in copy-move image forgery more accurately, especially when the size of the duplicated regions is small.

The rest of this paper is organized as follows: Section 2 describes the proposed approach in detail. Section 3 presents our experimental results and comparisons. Finally, section 4 concludes our work and discusses directions for future research.

II. Method Overview

The key objectives of the proposed approach are: (1) to recognize copy-move manipulated images, (2) to classify images as forged or non-forged, and (3) to accurately locate the duplicated region in the tampered images. Since in copy-move image forgery some part of an image is copied and pasted on another part of the same image, finding similar parts in an image is the key idea explored here as well as in other studies. This is accomplished by extracting and matching local features from different regions of the image in order to find similar regions. Figure 1 illustrates the main steps of our approach.

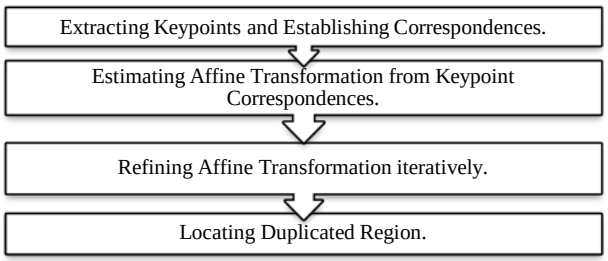


Figure 1. Main steps of proposed methodology.

In the following subsections, we explain the steps of the proposed method in detail.

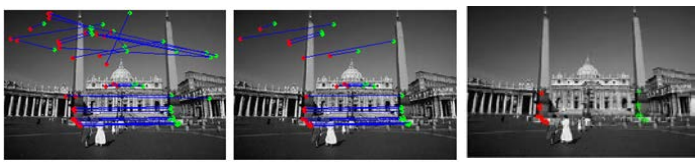
A. *Extracting Keypoints and Establishing Correspondences*

The SIFT algorithm is a powerful feature extraction technique [16], which extracts features invariant to scale, rotation, and brightness. However, SIFT descriptors are not invariant to mirror reflection. To account for this issue, previous approaches proposed extracting SIFT descriptors from horizontally and vertically reflected versions of the original image [14, 15]. In this paper, we have adopted MIFT [18] descriptors that are invariant to mirror reflection transformations. Since we search for duplicated regions in a single image, we divide the image into smaller parts and compare the descriptors among them. The search is performed outside a small window centered at the detected keypoint to avoid finding nearest neighbors of a keypoint from the same region [14]. Once a matching candidate has been found, it is accepted as a distinctive matched point if the ratio of the distances from the first and second nearest neighbors is smaller than the threshold [16]. This threshold can vary from zero to one; a threshold closer to zero yields more accurate but fewer matches. Here, a low threshold is utilized since it reduces false matches.

B. *Estimating Affine Transformation from Keypoint Correspondences*

Using the keypoint correspondences from the previous step, an affine transformation is estimated. To eliminate incorrectly matched keypoints before estimating the affine transformation parameters, a pre-processing step is applied using some simple geometric constraints (see below). To further remove incorrect matches, the affine transformation parameters are estimated using RANSAC [17], which can estimate the model parameters with a high degree of accuracy even when a significant number of wrong matches are present.

The geometric constraints applied in the pre-processing step are the "slope" and "location" constraints. To apply the "slope" constraint, the slope of all lines connecting corresponding keypoints are found and clustered and the group with the largest number of keypoints is selected as the main group. Then, we compare all other groups to the main group and eliminate any group having a different slope (i.e., within a threshold) from the slope of the main group. The "location" constraint is applied on the remaining groups by eliminating groups containing a small number of correspondences as well as removing corresponding keypoints from groups if the keypoint locations are rather far (i.e., within a threshold) from the average keypoint location of the group. To further remove incorrect matches and estimate the affine transformation matrix, we apply the RANSAC algorithm on the remained matched points. Figure 2 shows an example of the above steps.



(a) Initial correspondences (b) correspondences after the pre-processing step (c) Final matches (i.e., RANSAC inliers)

Figure 2. Removing incorrect correspondences using geometric constraints.

C. *Refining Affine Transformation*

Quite often, the correspondences selected as inliers in the previous section do not cover well the region of duplication; as a result, the estimated affine transformation is not precise enough to map the whole duplicated region to the copied region. To deal with this issue, we refine the affine transformation parameters iteratively, by slowly increasing the search window around the corresponding regions. Figure 3 shows the main steps of the refinement process.

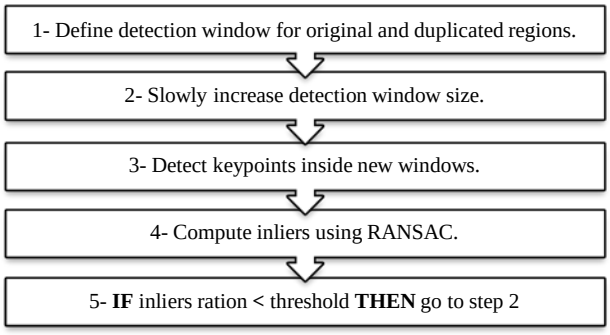


Figure 3. Mains steps of refining the affine transformation.

Given a pair of corresponding regions, first we define a detection window for each region using the inliers found by RANSAC. The detection windows are then slowly resized (i.e., horizontally and vertically). Then, keypoints are detected inside the resized windows and RANSAC is applied to find a new set of inliers. The new inliers are used to re-estimate the affine transformation parameters. By repeating these steps, the affine transformation parameters are refined iteratively until the number of inliers does not increase anymore. Figure 4 shows an example with five iterations. The number of correspondences and inliers at each iteration are shown in Table 1.

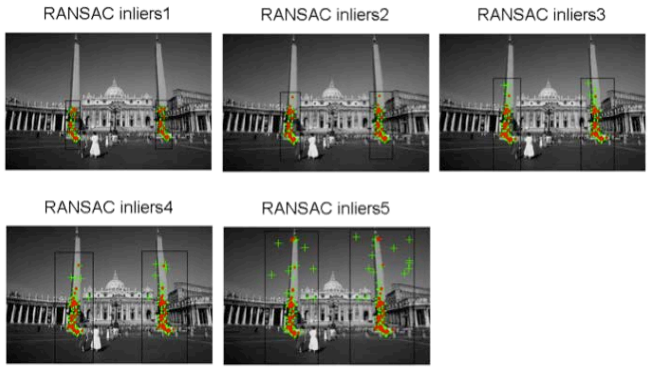


Figure 4. Refining the affine transformation iteratively; the green points show the initial correspondences while the red points show the inliers found by RANSAC.

As it is evident from the example, the iterative process yields more correspondences, covering a larger area inside the

original and duplicated regions; this yields a more accurate affine transformation. It should be mentioned that the threshold used for finding corresponding keypoints during the iterative process is greater than the one used in the initial step. This allows finding more correspondences compared to the initial stage.

TABLE 1. NUMBER OF CORRESPONDENCES AND RANSAC INLIERS AT EACH ITERATION

	Iterations						
	Initial Step	1	2	3	4	5	6
Correspondences		27	28	36	50	71	77
RANSAC Inliers	22	23	28	29	31	33	33

D. Locating Duplicated Region

The last step of our algorithm attempts to accurately locate the duplicated region. Cross-correlation has been used before to locate the duplicated region and verify similarity with the original region [14]. In this study, we detect the duplicated region using dense MIFT features.

1) Dense MIFT Feature Extraction

To detect as many pixels as possible inside the duplicated region, we employ dense MIFT features. The key idea is computing a MIFT descriptor at each pixel location inside the detection window instead of at the keypoint locations only. This is on contrast to traditional methods which employ pixel correlation to find the duplicated region. Since MIFT descriptors can be matched more accurately than pixels, the duplicated region can be detected more precisely. Using the estimated affine transformation, the correspondences between the original and forged regions can be computed at each pixel location. The similarity between corresponding locations is then calculated using dense MIFT descriptors. Thresholding the distance between corresponding MIFT descriptors can then reveal the duplicated region.

2) Hysteresis Thresholding

Using a single threshold to determine the similarity between corresponding MIFT descriptors in the original and duplicated regions might compromise detection results. In this work, we have opted for using hysteresis thresholding [37], a process based on two thresholds, one low and one high, which takes into consideration spatial information. Hysteresis thresholding has been used before in the context of edge detection where the high threshold is used to detect "strong" edges while the low threshold is used to fill in gaps between "strong" edges using "weak" edges [19]. In a similar manner, we use the high threshold to detect "strong" corresponding pixels, that is, corresponding pixels from the original and duplicated region having very similar MIFT descriptors (i.e., very likely to belong to the duplicated region). Additional pixels (i.e., "weak" pixels") are detected if they are adjacent to "strong" pixels and the distance between the corresponding MIFT descriptors in the original and duplicated regions exceeds the low threshold. In our experiments, the low threshold is chosen to be R times lower than the high one, where R is a ratio parameter.

The output of the step above is a group of pixels, which might still contain holes or isolated pixels. To deal with these issues, we apply morphological operations (i.e., dilation and erosion) to remove small holes and eliminate isolated pixels. These operations are applied separately on the images obtained using the high and low thresholds described in the previous section. Then, we simply combine the results to obtain the final duplicated region

III. EXPERIMENTAL RESULTS

In this section, the performance of the proposed approach is analyzed through a set of experiments. For comparison purposes, we have compared our method with the method of Pan and Lyu [14].

A. Dataset

To examine digital forgery detection methods, a dataset containing different types of forgery is required. In this study, we have used the CASIA tampered image detection evaluation database V2.0 (CASIA, 2010) [20]. CASIA v2.0 includes samples of copy-move and copy-paste digital forgeries applied on color images of different sizes, varying from 240 × 160 to 900 × 600. The tampered images have been generated by copying-and-pasting image region(s). The region selected for duplication can be transformed before copying it by applying scaling, rotation, reflection or distortion. The duplicated region can vary in size (e.g., small, medium or large). The resulted image can be post-processed (e.g., by applying blurring) in order to create the final altered image. In this paper, we have only used images corresponding to copy-move forgery. Since the dataset includes both the original and forged images, we have applied pixel subtraction followed by binary thresholding and morphological closing to extract the duplicated region (i.e., ground truth) to evaluate the accuracy of our method. A sample of forged images and the ground truth indicating the forged area is shown in Figure 5.

(a) Original image (b) Forged image (c) Ground truth

Figure 5. A sample of images and the ground truth in the CASIA Dataset

B. Implementation Details

As mentioned earlier, the first step of our approach is to extract a set of keypoint descriptors. In this study, we extract MIFT features; the window centered at keypoints is defined to be 15x15 pixels. Since our aim in this step is to find quite accurate correspondences, we use threshold equal to 0.2 for comparing MIFT descriptors which gives less but more accurate matches. If the number of correspondences is less than 10, we increase the threshold to 0.3 with step of 0.05. When removing incorrect matches using geometric constraints, we group corresponding points based on their slope in 10 groups. Additionally, to refine the affine transformation, the search windows are resized with a rate of 0.2 (i.e., both horizontally and vertically) in each iteration. In this step, we match the MIFT descriptors using a threshold equal to 0.3 in order to allow more matches to be found. In hysteresis thresholding, the high threshold is defined to be 2 times smaller than the low

one[1]. To evaluate the performance of our method, we employ Precision-Recall (PR) curves [21].

C. Detailed Results

To better evaluate the performance of our method, except the first experience, we have classified images into different categories based on the size of the duplicated region and the operations used to create the forgery. 2 shows the different evaluated categories and the number of images within each group. The PR curves shown below for each category correspond to the average PR curves over all the images in that category.

TABLE 2. IMAGE CATEGORIES IN CASIA V2.0 DATASET.

Tampering Region Size	Operations			Total
Medium	Scale	Rotate	-	87
Small	Scale	Rotate	-	17
Medium	Reflection	Scale	Rotate	35
Small	Reflection	Scale	Rotate	48
Small	Scale	Rotate	Blurring	8
Small	Scale	Rotate	Deform	19
Small	Deform			48

1) Effect of Thresholding

First, we compare standard thresholding with hysteresis thresholding. Since the output of thresholding is a group of pixels that might contain holes or isolated pixels, the morphological operations are applied prior to combining the results of the high and low thresholds in the hysteresis thresholding. Figure 6 shows two examples comparing standard thresholding with hysteresis thresholding. The duplicated regions have been produced using scaling in the top image and reflection in the bottom image. Figure 7 shows the corresponding PR curves. Clearly, hysteresis thresholding can locate the duplicated region more accurately.

(a) Detection windows. (b) Standard thresholding (c) Hysteresis thresholding

Figure 6. Comparison between standard and hysteresis thresholding.

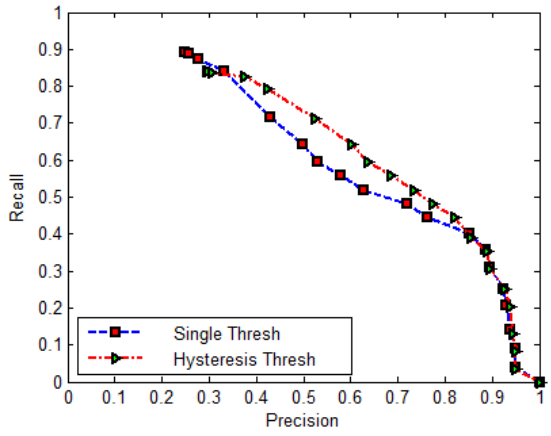
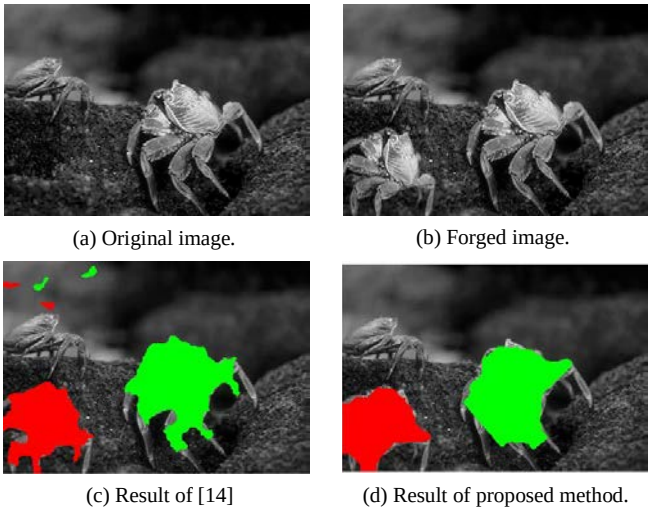


Figure 7. Comparison between standard (single) and hysteresis thresholding.

[1] Since in finding correspondences, a higher threshold yields a lower number of matches, we define the high and low values of hysteresis thresholding in opposite order compared to their definition in the literature.

2) Effect of Scale and Rotation

In this set of experiments, we consider the case where both scale and rotation have been applied to create the image forgery. As shown in Table 1, both medium and small sizes of duplicated regions have been measured. Figure 8 shows an example along with detection results for our method and the method of [14]. Figure 9 shows the corresponding PR curves; as the results indicate, the proposed method performs considerably better than the method of [14], especially when the size of the duplicated region is small.



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

(c) Result of [14] (d) Result of proposed method.

Figure 8. Detection of image forgery assuming both scale and rotation.

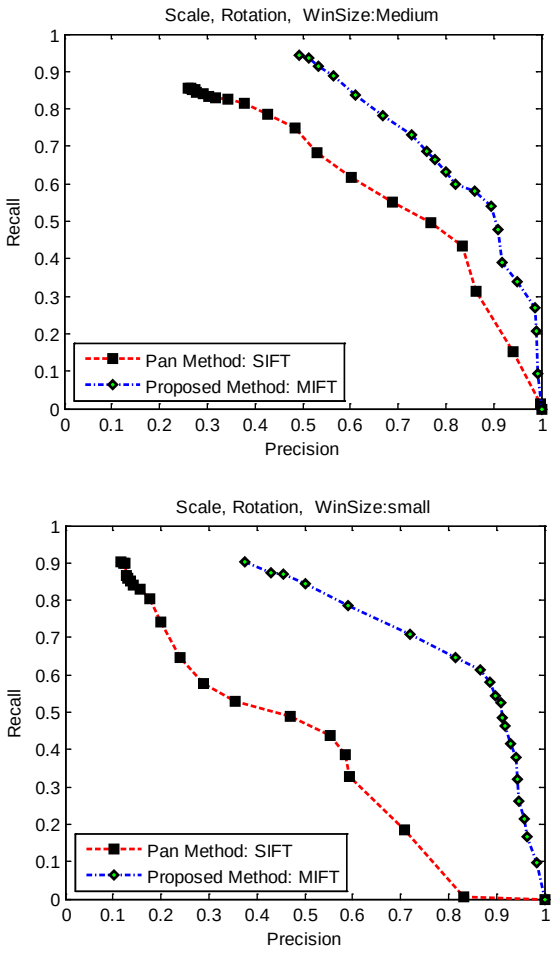
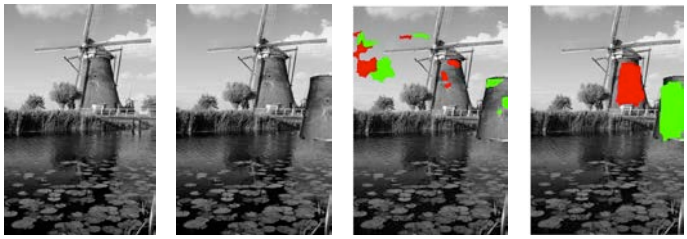


Figure 9. Comparison between the proposed method and the method of [14] assuming scale and rotation.

3) Effect of Reflection

As described earlier, mirror reflection is a common operation used in copy-move image forgery. The method presented in [14] handles reflection by filliping the feature vector of each keypoint horizontally and vertically before finding the similarities among the vectors. The accuracy of the methods is examined in this set of experiment assuming medium and small duplicated region sizes. The combination of mirror reflection with scale and rotation to create the duplicated region is investigated in this part. Figure 10 shows an example along with detection results. The accuracy of proposed method and the method of [14] are compared in Figure 11. The proposed method outperforms the method of [14], especially when the size of the duplicated region is small.



(a) Original image. (b) Forged image. (c) Result of [14] (d) Result of proposed method.

Figure 10. Detection of image forgery assuming mirror reflection, scale, and rotation.

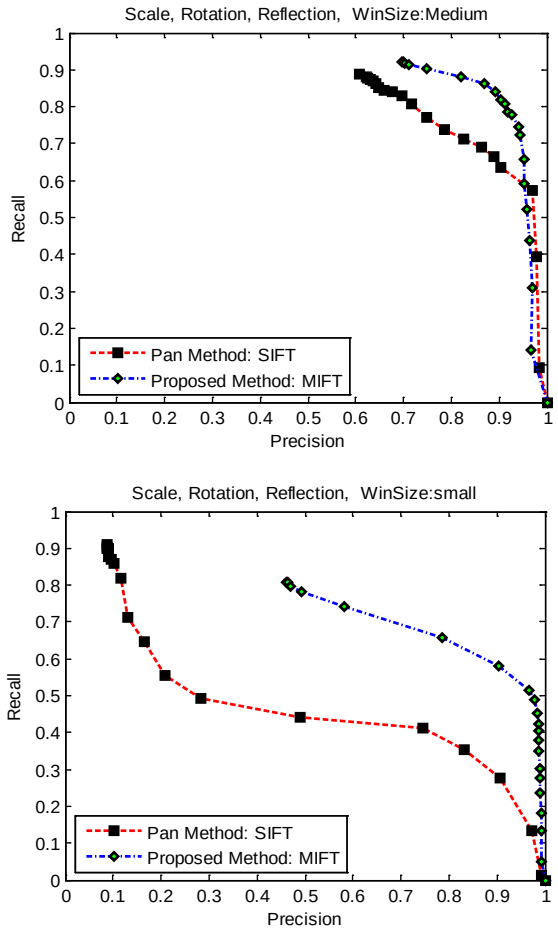
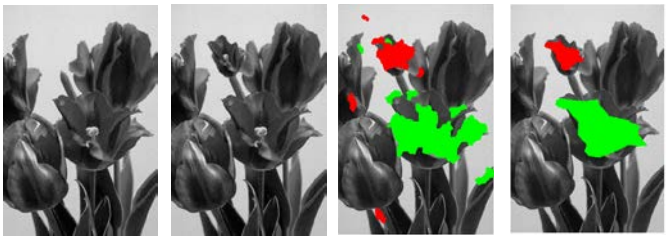


Figure 11. Comparison between the proposed method and the method of [14] assuming mirror reflection, scale, and rotation.

4) Effect of Blurring

In the CASIA dataset, blurring has been applied either on the edges of the duplicated region or on the whole region. This operation is typically combined with other operations such as scale and rotation. In this set of experiments, blurring, scale, and rotation are combined to create the image forgery. Figure 12 shows an example along with detection of the duplicated region. The accuracy of proposed method and the method presented in [14] are compared in Figure 13. This comparison was done using small duplicated region sizes only.



(a) Original image (b) Forged image (c) Result of [14] (d) Result of proposed method.

Figure 12. Detection of image forgery assuming blurring, scale and rotation.

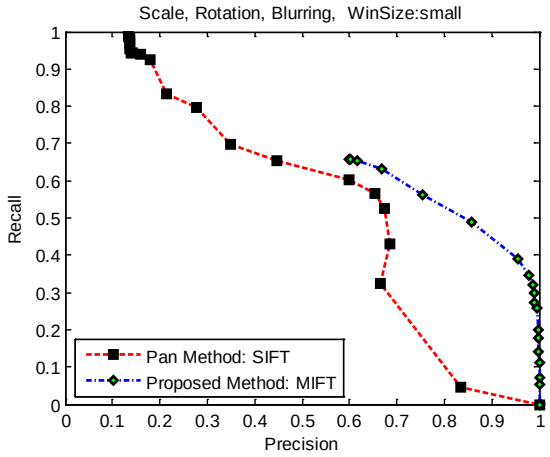
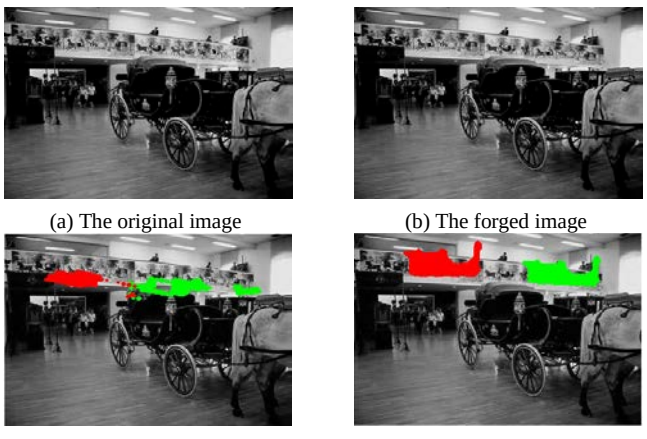


Figure 13. Comparison between the proposed method and the method of [14] assuming blurring, scale, and rotation.

5) Effect of Deformation

Deformation is another operation applied on the images of the CASIA dataset. This operation is typically a non-linear transformation. As shown below, detecting this kind of forgery has lower accuracy than forgery detection in other categories. This is due to the fact that we employ a linear transformation (e.g., affine) to bring similar regions into correspondence. Nevertheless, the proposed method still outperforms the method of [14].



(a) The original image (b) The forged image (c) Result of [14]. (d) Result of proposed method.

Figure 14. Detection of image forgery assuming deformation, scale and rotation.

In this set of experiments, we considered deformation, scale and rotation for image forgery. Figure 14 shows an example along with duplicated region detection results. As Figure 15 shows, our method outperforms the method of [14], however, extracting the duplicated region has a lower accuracy overall when combining all three transformations together. This comparison was done using small duplicated region sizes only.

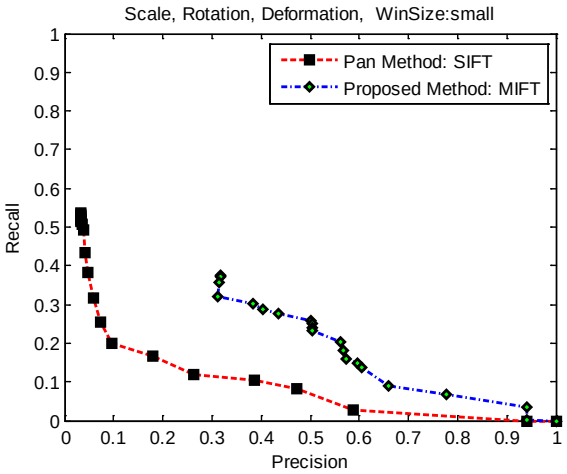


Figure 15. Comparison between the proposed method and the method of [14] assuming deformation, scale and rotation..

IV. Conclusions

In this paper, we have considered the problem of copy-move image forgery detection. Our emphasis was on detecting and extracting duplicated regions with higher accuracy and robustness. We have performed extensive experiments using a large dataset of real images to evaluate the proposed approach. In particular, we have investigated the effect of different transformations in creating the image forgery on detection accuracy. Comparisons with related methods indicate that the proposed methodology can extract duplicated regions more accurately. It should be mentioned that like with similar methods employing keypoint-based features for matching, the proposed approach will not work well if the duplicated region corresponds to a flat surface where no interest points can be detected.

Acknowledgment

This work is supported by grant 10-INF1140-02 under the National Plan for Science and Technology (NPST), King Saud University, Riyadh, Saudi Arabia. George Bebis is a Visiting Professor in the Department of Computer Science at King Saud University, Saudi Arabia.

References

[1] W. Luo, J. Huang, and G. Qiu, “Robust detection of region-duplication forgery in digital images,” in Proc. Int. Conf. on Pattern Recognition, Washington, D.C., 2006, pp. 746–749.

[2] B. Mahdian and S. Saic, “Detection of copy move forgery using a method based on blur moment invariants,” Forensic Sci. Int., 2007, vol. 171, pp. 180–189.

[3] S. Kumar and P. K. Das, “Copy-Move Forgery Detection in digital Images: Progress and Challenges”, International Journal on Computer Science and Engineering, Vol. 3, No. 2, pp. 652-663, February 2011.

[4] J. Fridrich, D. Soukal, and J. Lukas, “Detection of Copy-Move Forgery in Digital Images,” Department of Electrical and Computer Engineering, Department of Computer Science SUNY Binghamton, Binghamton, 2003, NY 13902-6000.

[5] J. Zhang, Z. Feng, Y. Su, “A new approach for detecting copy–move forgery in digital images,” IEEE Singapore International Conference on Communication Systems, 2008, pp. 362–366

[6] H. Farid, “A survey of image forgery detection,” in IEEE Signal Processing Magazine, vol. 2, no. 26, pp. 16–25, 2009.

[7] A.C. Popescu and H. Farid, “Exposing digital forgeries by detecting duplicated image regions,” Technical Report TR2004-515, Dartmouth College, Aug. 2004.

[8] Z. Lin, R. Wang, X. Tang, and H-V Shum, “Detecting doctored images using camera response normality and consistency,” in Proc. Computer Vision and Pattern Recognition, San Diego, CA, 2005.

[9] 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 IEEE Int. Conf. Multimedia and Expo, Beijing, China, 2007, pp. 1750–1753.

[10] S. Khan, A. Kulkarni, “An Efficient Method for Detection of Copy-Move Forgery Using Discrete Wavelet Transform,” International Journal on Computer Science and Engineering (IJCSE), 2010, 1801 - 1806.

[11] G. Muhammad, M. Hussain, K. Khawaji, and G. Bebis, "Blind Copy Move Image Forgery Detection Using Dyadic Uncedimated Wavelet Transform," 17th International Conference on Digital Signal Processing, Corfu, Greece, July 2011.

[12] G. Muhammad, M. Hussain and G. Bebis, "Passive copy move image forgery detection using undecimated dyadic wavelet transform", Digital Investigation, vol. 9, pp. 49-57, 2012.

[13] H. Huang, W. Guo, and Y. Zhang, “Detection of copy-move forgery in digital images using SIFT algorithm,” IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 2008.

[14] X. Pan and S. Lyu, “Region duplication detection using image feature matching,” IEEE Trans. Inf. Forensics Security, vol. 5, no. 4, pp.857–867, Dec. 2010.

[15] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, “A SIFT-based forensic method for copy-move attack detection and transformation recovery,” IEEE Trans. Inform. Forensics Security, vol. 6, no. 3, pp. 1099–1110, Jun. 2011.

[16] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” IJCV, 60(2): 91–110, 2004.

[17] M. A. Fischler and R.C. Bolles. “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,“ Communications of the ACM 1981, 24-381–395.

[18] X. Guo, X. Cao, J. Zhang, and X. Li, “Mift: A mirror reflection invariant feature descriptor,” In Proc. ACCV, 2009.

[19] J. Canny, “A Computational Approach To Edge Detection,” in IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6): 679–698, 1986.

[20] CASIA Image Tampering Detection Evaluation Database, Ver. 2.0, 2010, http://forensics.idealtest.org.

[21] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," Journal of Machine Learning Technologies 2 (1): 37-63, 2011.

[22] A. Vedaldi, and B. Fulkerson, ”An Open and Portable Library of Computer Vision Algorithms, ” 2008. http://www.vlfeat.org/.