DETECTING IMAGE REGION DUPLICATION USING SIFT FEATURES

Xunyu Pan and Siwei Lyu

Computer Science Department
College of Computing and Information
University at Albany, SUNY
Albany, NY 12222, USA
{xypan,lsw}@cs.albany.edu

ABSTRACT

Region duplication is a common form of image manipulation where part of an image is pasted to another location to conceal undesirable contents. Most existing methods to detect region duplication are based on finding exact copies of pixel blocks, which cannot handle cases when a region is scaled or rotated before pasted to a new location. In this work, we describe a new detection method based on matching image SIFT features [5]. The robustness of the SIFT features with regards to local transforms renders this method able to detect general region duplications with efficient computation. The effectiveness of this method is demonstrated with experimental results, both qualitatively and quantitatively in terms of the detection accuracy and the false positive rate.

***Index Terms*—** image forensics, region duplication, SIFT

1. INTRODUCTION

The availability and sophistication of digital imaging technology (e.g., cameras, computers, software) and their wide use on the Internet have made digital images a main source of information. Yet, thanks to the development of technologies for manipulating digital images, fraudulent digital images are also appearing with growing frequency and sophistication.

Region duplication or region cloning is a very common practice of image tampering, where a continuous portion of pixels in an image are pasted to a different location to conceal undesirable objects or contents in the original image (Fig.1). In recent years, several methods have been proposed to detect region duplication for the purpose of image forensics [2, 7, 6]. These methods are based on finding pixel blocks that are exact copies of each other in an image[1]. Such methods are most effective for the detection of region *copy-move*, where a region of pixels is pasted without any change to another location in the image. However, such simple operation may not achieve desirable result, and in practice, the duplicated region is often scaled or rotated slightly to better fit it into the surroundings at the target location. As such operations will change the pixel values, a direct matching of pixel blocks is unlikely to be effective for their detection.

Fig. 1. Left. *An image (courtesy of A. Popescu and H. Farid [7]).* **Right.** *Its forgery with region duplication to remove one person.*

In this work, we describe a new method to detect region duplication based on local image statistical features known as the *scale invariant features transform* (SIFT) [5]. From the matched SIFT features, the transform between two image regions containing such features is estimated. Different image regions are then compared by adjusting their relative transform, and their correlations are used to output a map showing regions with high likelihood to be duplicated from other regions. There are two main merits of the proposed method: as the SIFT features have been shown to be robust to rotation and scaling, this method is able to detect general region duplications when the duplicated regions undergo such geometric transforms. Furthermore, matching the SIFT features instead of pixel blocks makes the result less susceptible to noise and JPEG compression, which helps to improve the robustness of the overall detection. We demonstrate the effectiveness of the proposed method with experiments, both qualitatively with visual inspection of the detection results and quantitatively using the detection accuracies and the false positive rates.

2. PREVIOUS WORK

Denote the subset of pixel locations corresponding to the original region (*source*) as Ω_S, and that of the duplicated region (*target*) as Ω_T. Assuming no changes in the intensities, the region duplication operation leads to the following relation

[1] Though repeated patterns in an image (e.g., textures) can lead to false positives for region duplication, it is assumed here that in the original image there is no *exact* copies of such patterns.

between the pixels at the source and the target as $I(\Omega_T) = I(T_\theta(\Omega_S))$, where T_θ is a spatial transform of pixel locations with parameter θ between region Ω_S and Ω_T. The central task in detecting region duplication is to recover Ω_S and Ω_T, along with the spatial transform T_θ.

Most existing region duplication methods [2, 7, 6] assume a region is pasted to a new location without any change, i.e., T_θ is an identity. This special case of region duplication is often known as region *copy-move*, for the detection of which it is sufficient to compare pixel blocks and find exact copies. As a brute-force match of all pixel blocks of a given size in an image will have a running time quadratic to the size of the image, most methods focus on using low dimensional representations of blocks, e.g., PCA [7, 6] or DCT [2], and fast lexicographical sorting to improve efficiency. However, in practice, direct copy-move may not achieve desirable tampering, and the pixel regions are typically undergone further processing before or after being copied, such as scaling, rotation and boundary smoothing. The latter case has been recently discussed in [4]. However, for region duplications that involve scaling and rotating of the region before pasting, which can significantly disturb the pixel blocks, detection methods based on direct matching pixel blocks are unlikely to be effective.

3. REGION DUPLICATION DETECTION WITH SIFT

Here, we present a new region duplication detection method based on the image SIFT features. Specifically, to detect the locations of potential duplicated regions, we first detect SIFT keypoints in an image and compute the SIFT features for such keypoints (see Section 3.1 for details). We then segment the image into non-overlapping *examination blocks*, and for each SIFT keypoint in an examination block, we find its closest correspondence in the whole image. Using the matched SIFT keypoints, we estimate a potential transform T_θ for scaling or rotation between an image region and its duplication. The final likelihood of two regions to be the result of region duplication tampering is then evaluated with the correlation between the two regions with their relative transform being adjusted. Thus, the overall detection in the proposed method consists of four steps, (i) collecting SIFT features, (ii) SIFT feature matching and pruning, (iii) estimating region transforms and (iv) identifying duplicated regions using correlations adjusted with the estimated transforms. In the following, we will describe in details each of these steps.

3.1. Image SIFT Features

Distinctive local statistical image features are essential to problems such as image matching, object detection/recognition and tracking. Desirable properties of such features include efficient computation, robustness to local geometrical distortion, illumination variations, noise and other degradations. In recent years, several effective local image features have been proposed, of which the *scale invariant feature transform* (SIFT) [5] is shown to be very reliable and effective.

As described in [5], the first step in collecting SIFT features is to identify *keypoints* that are locations with distinct image information and robust to scaling and rotation. This is achieved by searching for locations that are stable local extrema in the image scale space, followed by a computation of the dominant local orientation at the key points. Note that the number of keypoints is usually much less than the number of pixels, thus subsequent computation will not be wasted at locations with little image information. At each keypoint, a SIFT feature vector is generated from the normalized histograms of local gradients in a neighborhood of pixels of that keypoint. The size of the neighborhood is determined by the scale of the keypoint, and all gradients are aligned with the dominant orientation at the keypoint. These steps ensure that the obtained local descriptors are invariant to rotation and scaling. Furthermore, the obtained histograms are normalized to unit length, which renders the SIFT features also robust to changes of global contrast. With the setting in [5], the final SIFT features are 128 dimensional vectors at each keypoint.

3.2. Matching and Pruning of SIFT keypoints

After all SIFT features are collected from an image, we find matches of SIFT keypoints in each small non-overlapping pixel blocks (known as examination block) in the whole image. Specifically, for each SIFT keypoint in an examination block, we compute the l_2 distances between its 128-dimensional SIFT feature with those of other keypoints not belong to the same examination block, and find its nearest neighbor using the Best-Bin-First (BBF) algorithm [1].

Matches of SIFT keypoints found this way are much noisy with many mismatches due to the nature of the procedure. To remove mismatches, we further prune keypoint matches that are not consistent with a priori spatial transform between regions. Specifically, matches using keypoints in the examination block that matches with multiple keypoints in the image are removed. We also check the bounding box of the correspondences of the keypoints in an examination block, and if its size is beyond certain range of the size of the examination block, the whole examination block is dropped from subsequent processing. After the pruning step, we record the number of correct matches for each examination block, and take the examination block with the maximum number of correct matches, ω_S, and the bounding block containing its matched SIFT point ω_T, as the basis to estimate the transform between the original region and its duplicate.

3.3. Estimating Region Transforms

After pruning, we estimate the potential transform between the original and the duplicated regions using ω_S and ω_T before estimating the shift of the two regions.

Copy-move. If there is no extra transform on the duplicated region, all pixels in the duplicated region are related to those in the original region with a common shift vector. Under this circumstance, we first compute the l_2 distances between each pair of matched SIFT keypoints. Keypoints corresponding to translated regions will have the same l_2 distance, though in practice due to noise and imaging artifacts there will be a distribution of such distances. We then build a histogram of such distances, and collecting keypoint pairs $P_S \subseteq \omega_S$ and $P_T \subseteq \omega_T$ with distances of maximum frequency of occurrence. The shift vector is then estimated as the difference between the means of P_S and P_T.

Scaling. If the duplicated region is scaled before being pasted to other location, the l_2 distance between a pair of keypoints in the duplicated region is a multiple of that of their correspondence in the original region. In other words, for any two SIFT keypoints $(\vec{x}, \vec{y}) \in \omega_S$ and their correspondences $(\vec{x'}, \vec{y'}) \in \omega_T$, $\|\vec{x} - \vec{y}\|/\|\vec{x'} - \vec{y'}\| = \text{const}$. However, due to imaging conditions and the matching procedure, there is a distribution of such scaling factors for a real image, which we estimate by computing pairwise l_2 distances for all SIFT keypoint pairs of ω_S and ω_T. We then form histograms of the ratios of such l_2 distance between corresponding pairs in ω_S and ω_T. The ratio with the maximum frequency is used as an estimation of the scale factor. Furthermore, keypoint pairs falling into that bin are used to estimate the translation between the original and the duplicated region as in the case of copy-move.

Rotation. The case where the duplicated region is rotated before pasted to the target location is slightly more complicated. Instead of directly estimating the rotation transform, which is numerically less stable given the noisy nature of the matched SIFT keypoints, we estimate the transform between two local coordinate systems of the original and the duplicated region. Specifically, we pick three non-collinear keypoints $(\vec{x}, \vec{y}, \vec{z}) \in \omega_S$ and their correspondences in $(\vec{x'}, \vec{y'}, \vec{z'}) \in \omega_T$ that have the strongest matches (measured by the l_2 distances of their corresponding SIFT features). The two sets of vectors, $(\vec{x}-\vec{y}, \vec{z}-\vec{y})$ and $(\vec{x'}-\vec{y'}, \vec{z'}-\vec{y'})$, form a local coordinate system for pixel locations in ω_S and ω_T, respectively. Each pixel location can be written as a linear combination of the two vectors, with the two linear combination weights being the two coordinates. As rotation doesn't change these coordinates, we compute coordinates of each pixel location in ω_S for $(\vec{x} - \vec{y}, \vec{z} - \vec{y})$. Their transformed correspondences are obtained by using the same set of coordinates in the coordinate system given by $(\vec{x'} - \vec{y'}, \vec{z'} - \vec{y'})$. After adjusting the rotation, translation between regions are estimated as the shift between the means of the two set of SIFT keypoints.

3.4. Showing Duplicated Regions

With the estimated region transform, we can establish the correspondence between all pixels in the original region and their counterparts in the duplicated region. From such a correspondence, we create a map of region correlations to identify the original and the duplicated regions[2]. In doing so, we first segment the image into overlapping *contour blocks*. Using the estimated transform, we compute the correlation coefficient between each contour block and its correspondence which generates a correlation map. We then process the correlation map to obtain an estimated contour of the original and the duplicated regions by first applying a Gaussian filter of 7×7 to smooth the correlation map and to remove the artifacts at the edge. Next, we choose a threshold to binarize the correlation map, i.e., if the correlation coefficient at is larger than the threshold, its value is reset to one, otherwise zero. This is followed by removal of regions with areas smaller than a pre-given threshold so to reduce the effect of noise. Finally, the contours of the potential original and duplicated regions are connected with mathematical morphological operation [8].

4. EXPERIMENTS

In this section, we describe experimental evaluations of the proposed region duplication detection method. Images used in our experiments are from [7]. The result reported are obtained with 32×32 examination blocks and 4×4 contour blocks with 3 pixel overlapping. The correlation threshold for contour blocks is 0.5, 0.4 and 0.2, and the area threshold is 300, 500 and 700 pixels, for simple copy-move, scaling and rotation, respectively.

JPEG	Q = 60	Q = 70	Q = 80	Q = 90	Q = 100
64×64	86.23/1.74	85.04/1.66	89.79/1.68	90.49/1.75	92.76/2.15
96×96	91.42/0.93	92.35/0.96	93.02/0.95	93.85/1.02	95.05/1.20
SNR	20 dB	25 dB	30 dB	35 dB	40 dB
64×64	89.76/1.78	92.06/2.03	92.43/2.08	92.55/2.07	92.61/2.13
96×96	92.84/1.02	94.24/1.13	94.62/1.18	94.70/1.18	94.78/1.19

Table 1. *Average detection accuracies and false positives (both in percentage) for different sizes of region duplication under JPEG compression and additive white noise.*

Shown in Fig.2 are several examples of region duplications with the image shown in Fig.1, with cases where the duplicated regions also undergo scaling (middle) and rotation (right). The detected duplication regions using the proposed method are shown with highlighted contours correspondingly in the bottom row. Note that region duplication with rotation (right) can be particularly challenge to identify visually or with matching of pixel blocks, yet the proposed method can reliably recover potential duplicated regions.

To further demonstrate the robustness of our method in the face of image degradations, we compute the detection accuracy and false positive rate with different JPEG compression rate and additive white noises (Table 1). Specifically, denote $\tilde{\Omega}_S$ and $\tilde{\Omega}_T$ as the detected region corresponding to the source and target, respectively, the detection accuracy and false positive rate are defined here as $p_{\text{acc}} = \frac{1}{2}\left(\frac{|\tilde{\Omega}_S \bigcap \Omega_S|}{|\Omega_S|} + \frac{|\tilde{\Omega}_T \bigcap \Omega_T|}{|\Omega_T|}\right)$ and

[2]Note the algorithm cannot differentiate the "original" from the "duplicate", but only suggests that these two regions are copies of each other.

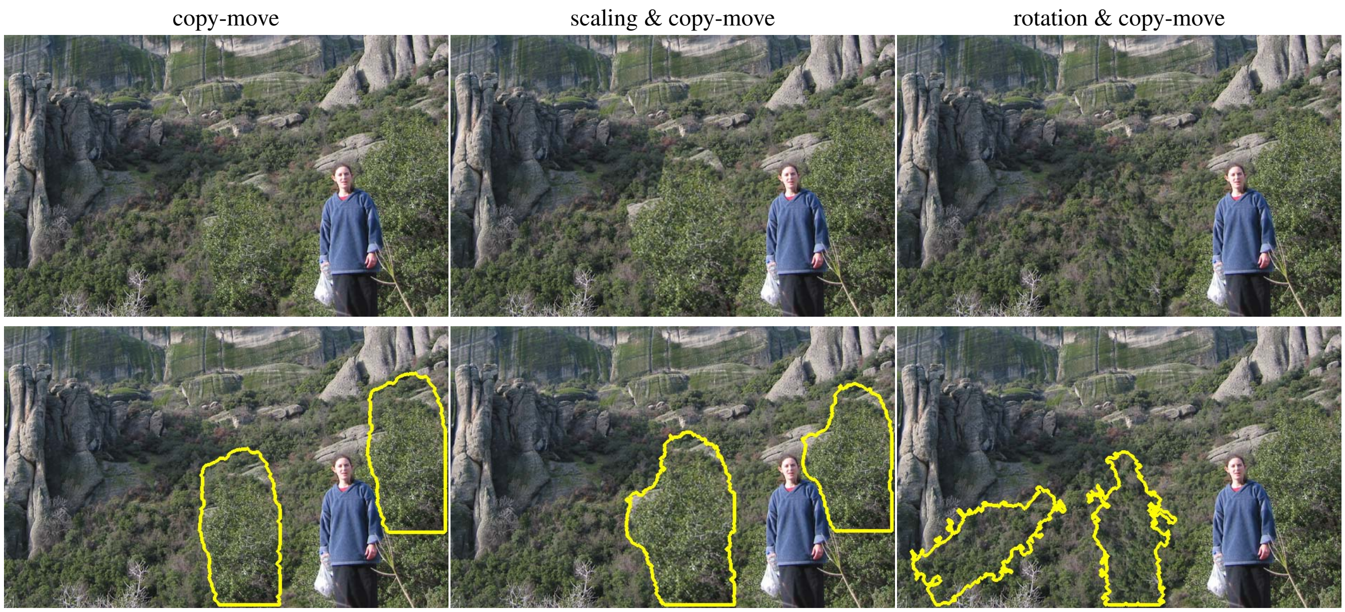

Fig. 2. **Top**: *three forgeries of image shown in Fig.1 with region duplication using translation, scaling and rotation.* **Bottom**: *detected duplicated regions using the proposed method with contours highlighted.*

$p_{\text{fpr}} = \frac{1}{2}\left(\frac{|\tilde{\Omega}_S - \Omega_S|}{|\tilde{\Omega}_S|} + \frac{|\tilde{\Omega}_T - \Omega_T|}{|\tilde{\Omega}_T|}\right)$, of which the former quantifies the fraction of duplicated region being correctly detected, and the latter evaluates the fraction of pixels that are not of the duplicated regions but are misclassified. Our experiments are performed with a set of images with randomly chosen square regions (of size 64×64 and 96×96 pixels) that are randomly pasted to other locations in the same image. As shown in Table 1, both rates are fairly stable with regards to different additive noise levels (measured with SNR) and JPEG qualities, which is attributed to the robustness of the SIFT features for such artifacts in images.

5. DISCUSSION AND FUTURE WORK

In this paper, we describe an effective method to detect image region duplication. Our method is based on local image SIFT features, which makes it applicable to the detection of general region duplications with region scaling and rotation that challenge methods based on matching pixel blocks. Experimental results demonstrate that this method is effective and robust in the presence of additive noise and different JPEG qualities.

A very recent work [3] has also proposed the use of SIFT features for detecting region copy-move. Compared to the method proposed here, the work in [3] is quite limited in that, first, it does not consider detection of general region duplication, and second, only matched keypoints are shown as detection results while our work provide a complete estimation of region contours.

There are several extensions of this work we are currently working on. First, though we are currently detecting region duplication with only geometric changes, the robustness of SIFT features to local luminance and contrast changes can also be use to detect duplicated regions with such variations. Second, we are also working on extending the current method to the detection of region duplication under general linear affine transforms, which include scaling and rotation as special cases. Finally, by incorporating temporal correlation, we hope similar methodology can also be developed for more effective detection of region duplications for videos [9].

6. REFERENCES

[1] J. Beis and D. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *CVPR*, 1997.

[2] J. Fridrich, D. Soukal, and J. Lukas. Detection of copy-move forgery in digital images. In *Digital Forensic Research Workshop*, 2003.

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

[4] D. Letscher. Detecting filtered cloning in digital images. In *ACM Workshop on MM&Sec*, 2007.

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

[6] W. Luo, J. Huang, and G. Qiu. Robust detection of region-duplication forgery in digital images. In *ICPR*, 2006.

[7] A.C. Popescu and H. Farid. Exposing digital forgeries by detecting duplicated image regions. Technical Report TR2004-515, Department of Computer Science, Dartmouth College, 2004.

[8] S. Suzuki and K. Abe. Topological structural analysis of digital binary images by border following. *CVGIP*, 30(1):32–46, 1985.

[9] W. Wang and H. Farid. Exposing digital forgeries in video by detecting duplication. In *ACM Workshop on MM&Sec*, 2007.