COMPUTER SCIENCE | RESEARCH ARTICLE

Keypoint-based passive method for image manipulation detection

Choudhary Shyam Prakash1*, Hari Om1, Sushila Maheshkar2 and Vikas Maheshkar3

Abstract: Due to the availability of media editing software, the authenticity and reliability of digital images are important. Region manipulation is a simple and effective method for digital image forgeries. Hence, the potential to identify the image manipulation is current research issue these days and copy-move forgery detection (CMFD) is a main domain in image authentication. In copy-move forgery, one region is simply copied and pasted over other regions in the same image for manipulating the image. In this paper, we have proposed a method based on Harris corner and Adaptive non-maximal Suppression (ANMS) for manipulation detection in an image. Initially, the input image is taken and then Harris corner detection algorithm is used to detect the interest points and ANMS is adopted to control the number of Harris points in an image. This gives a proper number of interest points for the different size of images and gives the assurance for finding the manipulated region in manageable time. For each extracted interest points we calculate the descriptors using SIFT then for the matching process of obtained descriptors, we use the outlier rejection with the nearest neighbour. Here, RANSAC is used to find the best set of matches to identify the manipulated regions. Experimental results show the robustness against different transformation and post-processing operations.

Subjects: Computer Engineering; Computer Graphics & Visualization; Computer Science; General

Keywords: Image forensics; copy-move forgery; SIFT descriptor; duplicate region detection; ANMS

ABOUT THE AUTHOR

Choudhary Shyam Prakash is currently a Ph.D. scholar in the Department of Computer Science and Engineering at IIT (ISM), Dhanbad, India. He received his B.Tech from Aryabhata Knowledge University Patna, India in 2012. His research interests includes Image processing, Image Forensics.

PUBLIC INTEREST STATEMENT

Images are considered as one of the important sources of information. It is being used as courtroom witness, insurance claims, medical imaging, etc. Hence, its authenticity is utmost important. There are many sophisticated tools/software available to manipulate an image without leaving any noticeable proof behind. To ensure the authenticity of an image, image forgery detection methods have been proposed. In this paper, we have proposed a keypoint-based method to detect the forgery in an image. Scale Invariant Feature Transform (SIFT) and adaptive non-maximal suppression (ANMS) are used to extract the feature keypoints. After matching these feature points, forged regions are located in an suspicious image.
1. Introduction
In today's scenario, manipulating an image is an easy task due to the availability of sophisticated image editing tools like Photoshop, GIMP (GNU image manipulation program), the NIK collection (offered by Google) and many more software which make image editing easy. Due to this fact relying on any image is hard to believe. Sometimes the perfection of manipulation is very high, so that it cannot be detected by naked eyes. In addition, the images are accepted as the popular source of information, and it plays a vital role when it is presented as the courtroom witness, insurance claims, scientific scams etc. Hence, the authenticity of an image is a big question that can be answered through different image manipulation detection techniques.

Digital image forensics can be categorized as intrusive (active) method and non-intrusive (passive) methods (Farid, 2009). In intrusive methods, digital watermark or digital signature is embedded into the images to authenticate the originality of an image. If the extracted watermark or signature is matched with the original one then it is said to be authentic otherwise not. In non-intrusive methods, there is not such kind of external media presented in the image. We have to rely only on the intrinsic properties of image to detect the authenticity of an image.

In all manipulation techniques, copy-move (region duplication) is a common manipulation technique, which is used to conceal some parts of the image by copy-pasting in the same image. An example of copy-move manipulation can be seen from the Figure 1. The main intention of copy-move manipulation is to suppress some important object in the image by copy-pasting a set of pixels from one part to another part of an image and it is hard to find the manipulation by naked eyes. For better visual effect and undetectable manipulation, the copy-moved region may go through several post-processing operations like rotation, scaling, illumination adjustments, blurred, noised and compressed to hide the record of any manipulation. Hence to develop an effective manipulation detection algorithm, these operations should be considered.

1.1. Motivation and scope
In this paper, we proposed a method to locate the manipulated region accurately with the help of Harris corner detection and adaptive non-maximal suppression. As we know, the computational cost of matching is directly proportional to the number of interest points extracted from an image. The main motivations of our work are:

(1) In the proposed method, we have adopted ANMS for better distribution of keypoints throughout the image, which helps to detect the duplicated regions even if the image is post-processed by different operations. The density of keypoints using Harris corner and proposed method can be seen from the Figure 2. In Chen et al. (Chen, Lu, Ni, Sun, & Huang,
2013) method, they have considered geometrical transformation (like rotation, scaling and flipping) and image degradation (like JPEG compression and Gaussian noise). ANMS not only results in detecting the above-mentioned attacks but also in addition if translation, distortion and combination as geometrical transformation and brightness change, image blurring, colour reduction and contrast adjustment as image degradation. By using ANMS, the robustness against the post-processing has increased than Chen et al. (Chen et al., 2013) method.

(2) In Harris corner, in general the time complexity for detecting the keypoints is $O(n^2)$ where $M \times M$ is the size of image. Chen et al. (Chen et al., 2013) used Harris corner method and step sector statistics and overall complexity of their method is $O(mn + n^2)$ whereas in our method we have used ANMS, which has complexity of $O(n^2)$. So overall proposed method has less complexity than Chen et al. (Chen et al., 2013) method.

The rest of the paper is organized as follows. In Section 2, we discuss the related work, which have been proposed yet with their merits and demerits. In Section 3, proposed method is discussed in detail. Then, we present the detailed experimentation in Section 4. In Section 5, we draw the conclusion of the paper.

2. Related works
As far as the literature has been concerned, it is found that the detection techniques for copy-move manipulation deal with the additional post processing operations like rotation, flipping, resizing, edge blurring, white Gaussian noise and JPEG compression. These post processing operations are applied to conceal the originality of the image and when it once applied it is hard to detect the manipulation.
Hence to tackle these problems, there are two different techniques adopted to spot the manipulations: block-based techniques and keypoint-based techniques.

2.1. Block-based technique

In block-based techniques, block of pixels is used to find the duplicated regions. Most of the techniques fall in this category. Fridrich et al. (Fridrich, Soukal, & Lukas, 2003) proposed a technique for exact match, and it is considered as the basic algorithm for duplicate region detection. Initially, a square sliding window is considered for extracting the block from image and then stored in a matrix. Then lexicographical sort is applied on the matrix and two consecutive identical rows/columns are considered as the identical region. But this method is not sufficient when additional post-processing is applied on the image. To deal with this problem, author proposed a robust method in which DCT was applied on each block. But still it is unable to detect the duplicated regions when Gaussian noise, rotation and resizing post-processing operations are applied on the image.

Popescue and Farid (Popescu & Farid, 2004) proposed a method to reduce the dimension of the block using Principle Component Analysis (PCA). Their method shows better results under Gaussian noise addition and JPEG compression. Luo et al. (Luo, Huang, & Qiu, 2006) used RGB channel and extract seven features based on the average of the pixel intensity. Their method achieved higher accuracy rate against JPEG compression, Gaussian blurring and additive Gaussian noise but it still fails in rotation, resizing and flipping operations performed on image.

Mahadian and Saic (Mahdian & Saic, 2006) used Blur moment invariants for generating a feature vector of each block. They have calculated 24 blur moment invariants, hence the dimension of feature vector is 72 as the blur moment invariant calculated on each RGB (Red, Green, Blue) channel separately. Further to reduce the dimensions, they applied PCA. By this method they got the effective results in detection of duplicated regions even in the presence of JPEG compression, Gaussian blurring and additive Gaussian noise. But they did not consider the problem of rotation, resizing and flipping.

Zhang et al. (Zhang, Feng, & Su, 2010) used Discrete Wavelet Transform (DWT) and phase correlation for extracting the feature of blocks. Their method shows robustness under JPEG compression and blurring only. Li et al. (Li, Wu, Tu, & Sun, 2007) used DWT and singular value decomposition (SVD) to reduce the dimension and found a robust block representation. Their approach only works for JPEG compression when quality factor is greater than 70%. Kang and Wei (Kang & Wei, 2008) proposed a method in which SVD is used to extract the feature vector from the overlapping blocks. SVD is invariant to the noise which is useful for copy-move detection. However, rotation and resizing are not considered in their method.

Bravo-Solario and Nandi (Bravo-Solorio & Nandi, 2011) used correlation coefficients of Fourier Transform (FT) as the similarity measure between blocks of pixels in log-polar form. They overcome efficiently the problem of flipping operation on forged image but resizing and rotation still persisted in their method. Bayram et al. (Bayram, Sencar, & Memon, 2009) used Fourier-Mellin Transform (FMT) for extracting the features from block and Counting Bloom Filters (CBF) to sort the extracted features. Their method is able to detect the forgery under JPEG compression, resizing and rotation but the efficiency is decreased due to the use of CBF as it deals only with very similar blocks.

Wang et al. (Wang, Liu, Li, Dai, & Wang, 2009) proposed a method in which circular sliding window and Gaussian Pyramid Decomposition (GPD) are used to extract the features from blocks. The method is able to detect forgery under JPEG compression, rotation, blurring and horizontal flipping transformation but fails under resizing operation.

Lin et al. (Lin, Wang, & Kao, 2009) proposed a method in which 9-Dimensional feature vector is used which contain the information about pixel intensities in some specific regions of the block. Instead of using lexicographical sort they used Radix sort which gives a linear time sorting. Their method is
robust to JPEG compression and Gaussian blurring only. Wang et al. (Wang, Liu, Zhang, Dai, & Wang, 2010) initially applied GPD on input image then Hu moment of each block is calculated to extract the features. Their method is robust to JPEG compression, Gaussian blurring and Gaussian noise.

2.2. Keypoint-based technique

As an alternative of block-based techniques the methods based on keypoints are proposed for forgery detection by many authors as shown in Table 1. In keypoint-based techniques, image keypoints are detected and matched in the whole image for detection of duplicated region. In Table 1, we have shown the different keypoints-based forgery detection techniques with their method and considered post-processing operations in their method.

Huang et al. (Huang, Guo, & Zhang, 2008) proposed a method in which Scale-Invariant Feature Transform (SIFT) descriptor is used for keypoint extraction. The advantage of using descriptor is that, it is reasonably invariant to noise, rotations, resizing and lightning adjustments. By comparing the descriptor, duplicated region is found but it fails for small duplicated regions. Pan and Lyu (Pan & Lyu, 2010a) (Pan & Lyu, 2010b) used SIFT descriptor which is able to detect the forgery region even if the image is processed with JPEG compression and Gaussian noise. Amerini et al. (Amerini, Ballan, Caldelli, Bimbo, & Serra, 2011) performed more detailed study on SIFT descriptor to detect the forged region. They have used hierarchical clustering procedure which also considered the geometric transformation in the forged region. The main reason for using hierarchical clustering is to reduce the false positive rates and detection of multiple duplicated regions.

Xu et al. (Xu, Wang, Liu, Li, & Dai, 2010) used speeded-up robust feature (SURF) instead of SIFT descriptor to locate the forged region. They have extracted the SURF descriptor from the image and divided it into randomly two groups for comparison of descriptors. They have used nearest neighbour algorithm in each group for saving the matching time. Their method considers the rotation, resizing, blurring and white Gaussian noise operation efficiently. But these kinds of methods may face the difficulties in locating forged homogeneous regions as they could not provide enough keypoints for a proper analysis. Shivkumar and Baboo (Shivakumar & Baboo, 2011) proposed a method in which SURF is used for keypoints extraction and stored in a Kd-tree.

### Table 1. Analysis of some of the popular keypoint-based techniques

<table>
<thead>
<tr>
<th>Techniques</th>
<th>Method</th>
<th>Post-processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang et al. (Huang et al., 2008)</td>
<td>SIFT</td>
<td>Rotation, Noise, JPEG compression</td>
</tr>
<tr>
<td>Pan &amp; Lyu (Pan &amp; Lyu, 2010a, Pan &amp; Lyu, 2010b)</td>
<td>SIFT</td>
<td>Rotation, Scaling, Illumination adjustment</td>
</tr>
<tr>
<td>Amerini et al. (Amerini et al., 2011)</td>
<td>SIFT</td>
<td>JPEG compression, Noise</td>
</tr>
<tr>
<td>L. Chen et al. (Chen et al., 2013)</td>
<td>SIFT &amp;</td>
<td>Rotation, Scaling, Noise, Flipping, JPEG compression</td>
</tr>
<tr>
<td>Xu et al. (Xu et al., 2010)</td>
<td>SURF</td>
<td>Rotation, Resizing, Blurring, Noise</td>
</tr>
<tr>
<td>Shivkumar &amp; Baboo (Shivakumar &amp; Baboo, 2011)</td>
<td>SURF &amp;</td>
<td>Rotation, Noise</td>
</tr>
<tr>
<td>Proposed method</td>
<td>SIFT &amp; ANMS</td>
<td>Translation, Rotation, Scaling, Distortion, Combination, Brightness change, Noise, Image Blurring, Colour Reduction, Contrast Adjustment, JPEG compression</td>
</tr>
</tbody>
</table>
The use of Kd-tree improved the matching process and generate less false negative rates. The authors have presented only visual results of two cases of resizing, rotation and Gaussian noise.

Christlein et al. (Christlein, Riess, Jordan, Riess, & Angelopoulou, 2012) explored many copy-move forgery approaches (Amerini et al., 2011; Bashar, Noda, Ohnishi, & Mori, 2010; Bayram et al., 2009; Bravo-Solorio & Nandi, 2011; Fridrich et al., 2003; Huang et al., 2008; Kang & Wei, 2008; Lin et al., 2009; Luo et al., 2006; Mahdian & Saic, 2006; Pan & Lyu, 2010a; Popescu & Farid, 2004; Ryu, Lee, & Lee, 2010; Shivakumar & Baboo, 2011; Wang et al., 2009, 2010; Xu et al., 2010) and analyzed their performance on different scenarios. Their experiments show that the methods (Bravo-Solorio & Nandi, 2011; Ryu et al., 2010; Wang et al., 2009) give good results under rotation with keypoint-based techniques. By using Zenerik moments, KPCA, PCA, or DCT, authors get good results under scaling operations. Finally, when those post-processing operations are combined, then keypoint-based techniques show the best precision results in different scenario like large rotation, resizing and high compression on the image.

As per the above discussion about the existing techniques for copy-move manipulation detection: block-based and keypoint-based techniques.

The block-based techniques are most popular in copy-move manipulation detection technique due to the significance with various feature extraction techniques and ability to obtain high matching performance. Common techniques adopted for feature extraction are based on frequency transform domain and lexicographical sorting. Then matching process is applied for detection of duplicated regions. Due to the invariance of frequency transform against post-processing operations like JPEG compression, noise and blurring, it is adopted for feature extraction. But it also has some limitations like it is not invariant to geometrical transformation (Rotation and Scaling). So researchers have applied moment and log-polar transform to handle the problem of geometrical transformation operations and increase the robustness of detection techniques. In addition PCA, DWT and SVD are used to reduce the dimension of the features and increase the processing speed.

Keypoint-based technique is also becoming popular in copy-move forgery detection (CMFD). In this category SIFT feature is most reliable and popular detection technique because of its good performance in geometrical transformation. However, SIFT also have some drawbacks as it is not capable of detecting copy-move region in texture either from flat or smooth region or highly identical features region and it has high false positive ratio when handling these kind of problem. Hence the main focus in keypoint-based detection techniques is to improve its accuracy while maintaining the robustness in geometrical transformation operations, and it is the main advantage of using keypoint-based techniques over block-based techniques.

### 3. Proposed method

The major goal of our proposed method is to perceive a method to recognize copy-move manipulated images and spot the duplicated region accurately. Since in copy-move manipulation, a part of image is copied and pasted on other part in the same image, so getting the accurate similar region is the key idea presented here as well as in other developed method. This can be achieved through extracting and matching the local features of the image to get the similar regions.

Here, we have proposed a manipulation detection method based on keypoint detection technique. There are many keypoints detection technique available like Harris corners, SIFT, MSER etc. In the proposed work, the Harris corner method is used to detect the keypoints.

#### 3.1. Harris corner detection

An interest point in an image is a point with well defined position and can be detected robustly (Moravec, 1980). It means it doesn’t change its characteristics even if the image gone through geometrical
transformation or illumination adjustment. It can be a corner, an isolated point of local intensity maximum or minimum, a line ending or a point on a curve where the curvature is locally maximal.

The Harris corner detection algorithm (Harris & Stephens, 1988) uses the fact that, the intensity of image changes a lot in multiple directions at a corner, while it changes greatly in a certain direction at an edge. To formulate this anomaly of change in intensity, a local window is used to get the result from shifting the local window. When the window is shifted in an arbitrary direction around a corner point, the image intensity changes greatly. Around the edge point, it will change greatly when the window is shifted perpendicularly. By using this perception, Harris corner detection algorithm uses a second-order moment matrix as the basis of its corner decisions.

Autocorrelation matrix $M$ for a given image $I$ at point $(x,y)$ can be given by

$$ M(x,y) = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2(x,y) & I_x I_y(x,y) \\ I_x I_y(x,y) & I_y^2(x,y) \end{bmatrix} $$

(1)

where $I_x$ and $I_y$ are the derivatives of pixel intensity, respectively, in the $x$ and $y$ direction at point $(x, y)$ i.e.

$$ I_x = I \ast [-1, 0, 1] = \partial I / \partial x $$

(2)

$$ I_y = I \ast [-1, 0, 1] = \partial I / \partial y $$

(3)

where operator $\ast$ denotes convolution. The weighting function $w(x,y)$ can be given by

$$ w(x,y) = g(x,y,\sigma) = \frac{1}{2\pi\sigma^2} \exp(-\frac{x^2+y^2}{2\sigma^2}) $$

(4)

It assigns more weight to those values near the centre of a local region. It can be uniform, but it is more typically an isotropic, circular Gaussian.

To find the interest point, we first compute the corner strength function which is harmonic mean of the matrix $O_{mn+n^2}$.

$$ f_M(x,y) = \frac{\text{det}M(x,y)}{\text{tr}M(x,y)} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} $$

(5)

where $\lambda_1$ & $\lambda_2$ are the eigenvalues of $M$. These points are selected as interest point where $f_M(x,y)$ is a local maxima in window of $3 \times 3$ neighbourhood and above a threshold $t = 10$. When the local-maxima is detected, its position is refined to sub-pixel accuracy by fitting a 2D quadratic to the corner strength function in the local window of $3 \times 3$ neighbourhood and finding its maxima. The orientation can be detected by two methods, the first method is to find the eigenvectors which denote the largest eigenvalue of the Harris matrix $M$, and the second method is simply using the gradient at that point as its orientation. Here, we have used second method.

### 3.2. Adaptive non-maximal suppression

It is a novel approach, which is used to control the density of features in the image and gives a better spatial distribution of features than previous approaches as shown in Algorithm 1. The computational cost of matching is directly proportional to the number of interest points extracted from an image, and it should be restricted to minimize the computational cost of matching. At the same time, it is essential that the interest points should be spatially well distributed over the image. To get the above-mentioned requirements, adaptive non-maximal suppression (ANMS) is used to select a fixed number of interest points from an image.

Based on $f_M$, interest points are suppressed to select a fixed number of interest points and those pixels are considered, which are maximum in a neighbourhood of radius $3 \times 3$. Initially, the suppression radius $r = 0$ is taken, and then it is increased until the desired number of interest
points $3 \times 3$ is obtained. The first in the list is the global maxima, which is not suppressed at any radius. As the suppression radius decreases from infinity, interest points are added to the list. However, if an interest point appears once, it will remain in the list. This is just because if an interest point is maximum in radius $r$ then it is also a maximum in radius $r' < r$. In practice, we make the non-maximal suppression more robust by requiring that a neighbour has a sufficient larger strength. Thus, the minimum suppression radius $r$ is given by

$$r_i = \min_j |x_j - x_i|, \text{s.t. } f(x_i) < C_{\text{robust}} f(x_i), x_j \in I$$

(6)

where $x_i$ is a 2D interest point image location and $I$ is the set of all interest points locations. $C_{\text{robust}} = 0.9$ is taken to ensure that a neighbour must have sufficient higher strength for suppression to take place.

**Algorithm 1** Adaptive Non-maximal Suppression

**Input**: Interest points detected by Harris corner method

**Output**: Interest points filtered by ANMS for better distribution of interest points.

1. **procedure** FUNCTION
2. **Function**($I$, $r$, $C_{\text{robust}}$) --- ANMS (Interest_point, No_of_Interest_points, $C_{\text{robust}}$)
3. **Interest_point** $\leftarrow$ SORTROWS (Interest_point, $-3$);
4. $R$ $\leftarrow$ zeros (size(Interest_point, 1), 1);
5. **for** $i$ $\leftarrow$ 1 to size(Interest_point, 1) **do**
6. $f_xi$ $\leftarrow$ Interest_point($i$, 3);
7. [indices, values] $\leftarrow$ find($f_xi < Interest_point(1: i, 3) \times C_{\text{robust}}$);
8. $I.j$ $\leftarrow$ Interest_point(indices, :);
9. **if** numel(I.j) $> 0$ **then**
10. $y$ $\leftarrow$ Interest_point($i$, 1);
11. $x$ $\leftarrow$ Interest_point($i$, 2);
12. $ys$ $\leftarrow$ $(y - I.j(:, 1))^2$;
13. $xs$ $\leftarrow$ $(x - I.j(:, 2))^2$;
14. $R(i)$ $\leftarrow$ min(sqrt(ys + xs));
15. **else**
16. $R(i)$ $\leftarrow$ $-1$;
17. **end if**
18. **end for**


3.3. Feature descriptor

By extracting the keypoints, we have only obtained the information about their position and sometimes their coverage area (usually approximated by a circle or ellipse) in the image. Now the objective is to get the description of the local image structure for efficient and reliable matching of features in the whole image. Previously, many feature vector have been developed including Gaussian derivatives (Schmid & Mohr, 1997), local intensity patches (Forstner, 1986; Harris, 1992), scale invariant feature transform (Zhang et al., 2010) and affine-invariant descriptor (Baumberg, 2000; Brown & Lowe, 2002; Tuytelaars & Van Gool, 2000) for matching of features. According to Mikolajczyk and Schmid (Mikolajczyk & Schmid, 2003) survey, SIFT features generally gives the best performance.

In SIFT, after applying ANMS algorithm, the set of interest points are obtained. A $16 \times 16$ window around these extracted points is applied as shown in Figure 3. Within this window to select the keypoints descriptors, the subdivision of $4 \times 4$ window is considered. The orientation and magnitude of the gradient at that point is calculated and put them into 8 buckets. This is done for each of the sixteen $4 \times 4$ windows for a total of 128 features descriptors. Then, the magnitude is subtracted from the orientation of the interest points followed by normalization. To avoid any values that are significantly higher than the rest, a threshold of 0.2 is used to normalize vector and renormalizes to the unit length (Shivakumar & Santhosh Baboo, 2011).

3.4. Detection algorithm

The proposed technique for image manipulation detection based on Harris interest points and ANMS is shown in Figure 4, and the algorithm is shown in Algorithm 2.

Initially, the input image is taken and then Harris corner detection algorithm is used to detect the interest points, which are described in Section 2. The number of interest points detected from Harris corner detection have high relevance. If the number of Harris points are small, it is difficult to detect the manipulated region, and if the number of Harris points are greater, then it would be time consuming to find the manipulated region. Hence, we have

---

19 \( I_r \leftarrow \text{cat}(2, \text{Interest\_point}, R) \);

20 \( I_r \leftarrow \text{SORTROWS}(I_r, -4) \);

21 \( I_r \leftarrow I_r(1 : \text{No\_of\_Interest\_points}, 1 : 3) \);

22 end procedure

---

Figure 3. SIFT Descriptor.
adopted ANMS to control the number of Harris points in an image. In this way, we can get a proper number of interest points for different image size and give the assurance for finding the manipulated region in manageable time.

Now for matching process of obtained descriptors, we use the outliers rejection with nearest neighbour. Because of smoothness of natural image generally the best match of a Harris point resides within its close spatial adjacency. Hence, it is important to restrict the searching of nearest neighbour of a Harris point. To restrict the searching, we use a certain distance like $10 \sqrt{2}$ (As an example of $10 \times 10$ size region) between two matched Harris points. Still, many Harris points can be matched with each other but we select only those points with distinct similarities. The feature vector $f'$ other than $f$ and $f''$, the distance between $f$ and $f'$ must satisfy $\| f' - f \|_2 < \epsilon \| f' - f'' \|_2$, where $\epsilon \in (0, 1)$ and $f$ is the Harris point location $(x, y)$ and $f'$ is the corresponding feature vector of Harris point at $(x', y')$. The matched Harris points are recorded pair to pair. For matching process of obtained descriptors, we use the outliers rejection with nearest neighbour. To find the first and second nearest neighbour for each descriptor, we generally use a basic distance metric, i.e. Euclidean distance. It can be given by

$$A_{ij} = \sqrt{\sum_{t=1}^{n} (D_i - D_j)^2}$$  \hspace{1cm} (7)

where $D_i$ and $D_j$ represent the two different descriptor vectors. Now we take the ratio of the distance, and if the obtained ratio is less than the defined threshold value 0.4 (optimal value for panorama stitching (Brown & Lowe, 2007)), then we consider those descriptor as matching descriptor. Here, we conduct our experiment by varying the threshold value. If the threshold value is less than 0.4, the matches will be stronger and subsequently the execution is faster. If the threshold value is greater than 0.4, then there will be more matches, and it increases the execution time. With our method, 0.4 is a good trade off between speed and breadth of points. For each descriptor, the distance to itself is set to be infinity, otherwise it will find itself as the nearest neighbour.

Additionally, to improve the speed of the matching we use filtering. There are few interest points which are close to each other by few pixels, since the texture doesn’t change too much, hence we
remove those matches that are 3 pixels or less apart. After matching process, we got many equivalent points that have similar features and now to find the copy-move region we use RANSAC (Fischler & Bolles, 1981) to find the best set of matches. RANSAC is an iterative method, which frequently selects 4 points and computes the sum of squared differences (SSD) of each set of equivalent points (Brown & Lowe, 2007). Then, it keeps the set of inliers, where the SSD is less than some epsilon values. There is an inadequacy with RANSAC is that, it has single inlier set and therefore it doesn’t scale well to multiple translations. Here we use multi-RANSAC (mRANSAC) to solve this problem. It is similar to RANSAC, except that it saves the top k inlier sets (we used k = 6). It helps to capture multiple transformations through which we can detect multiple copy move regions within an image.

By using RANSAC, the multiple copy-move regions can be detected as shown in Figure 5(a). However, as seen from Figure 5(b), only partially duplicated regions can be detected without applying RANSAC.

Algorithm 2 Proposed Algorithm

Input: Suspicious Image

Output: Detected tampered region

1 procedure

2 tor SIFT ← 1; /* Initialize Descriptor */

3 ENABLE ANMS ← TRUE/FALSE; /* If ANMS is used */

4 Im ← Readthesuspiciousimage;

5 /* Get Interest points */

Figure 5. Detection of copy-move attack (a) With RANSAC (b) Without RANSAC.
Interest_point_original ← HARRIS (Im);

if ENABLE_ANMS = True then

Interest_points ← ANMS(interest_point_original, No_of_Interest_points, c_robust);

else

Interest_point ← interest_point_original;
end if

/* Find Descriptors*/

Descriptors ← sift_keypoint_descriptor (Im, Interest_point);

/* Matching*/

Matches ← nn_outlier_rejection(descriptor1, descriptor2, nn_threshold); /*Use nearest neighbour outliers rejection*/

Matches ← filter_small_matches(matches, threshold);

/*Filtering the matches*/

if ENABLE_RANSAC then

if !LOOK_FOR_MULTIPLE then

Matches ← ransac(matches, iteration, error_threshold);

else

Matches ← ransac_multi(matches, iteration, error_threshold);

end if

end if

Tampered_region ← Matches;

end procedure

4. Experimental analysis

The performance evaluation of the proposed manipulation detection algorithm is conducted on two dataset Comofod (Tralic, Zupancic, Grgic, & Grgic, 2013) and Image Manipulation Dataset (Christlein et al., 2012). The forged images of the two datasets are of size 512 × 512 and 600 × 400 respectively. The experiments are performed on RGB images present in the datasets. All images are manipulated in such a way that it cannot be detected by naked eyes. All the experiments are performed on Matlab R2013a. In the following subsections, we have discussed the experimental results in detail.

4.1. Copy-move manipulation

In this section, the duplicate region is copied and pasted over other region in the same image without using any post-processing operations. In Comofod (Tralic et al., 2013) and Image
Manipulation Dataset (Christlein et al., 2012), there are 200 and 45 forged images in both of the dataset of different size, respectively. The example of detection results can be seen from Figure 6. The accuracy of detected copy-move attack is shown in Table 2. As it can be seen that the proposed method works good on both of the datasets and results are satisfactory.

4.2. Transformation duplication manipulation

In this section, we applied some of the popular transformations to the copied region before changing its location. We applied transformation like translation, rotation, scaling, distortion and combination on the Comofod (Tralic et al., 2013) dataset in our experiment. In translation, a copied region is simply translated to the other region without performing any transformation. In rotation, the duplicated region is copied and rotated with some angle \( \theta \in (0^\circ, 360^\circ) \) and then pasted over other region of the image. In scaling, the duplicated region is scaled up or down with some scaling factor and then pasted over other region of the image. In distortion, the copied region is distorted and then pasted over other region of the image. For combination two or more (like rotation and scaling) transformation are applied on a copied region and then pasted over other region of the image. The detection results can be seen from Figure 7, and it is observed that the proposed method can successfully detect the different transformations like translation, rotation, scaling, distortion and combination. The accuracy of the proposed method on different transformation is listed in Table 3, and it is observed that the Detection Accuracy Rate (DAR) of the detection results are quite satisfactory. Also the average False Positive Rate (FPR) is 10.5\% indicating the efficiency of the proposed algorithm.

4.3. Robustness against post-processing

In copy-move manipulation detection method, the strength of resisting the post-processing attacks is fundamental requirement. There are many types of post-processing operations, which can be applied to conceal the traces of tampering in an image. The most common post-processing...
Figure 7. Detection results of Transformations (a) Translation (b) Rotation (c) Scaling (d) Distortion (e) Combination.
methods are JPEG compression, noise adding and image blurring but there are also some other methods like brightness change, colour reduction and contrast adjustments, which are frequently used these days. In the following, we present a detail analysis of post-processing attacks which were applied to duplicate regions.

In this paper, six different types of post-processing attacks are considered on the Comofod (Tralic et al., 2013) dataset. Most of the attacks are implemented in MATLAB apart from JPEG compression which is applied through PIXresizer software.

4.3.1. Noise, colour reduction and contrast adjustment attacks

In noise adding post-processing attack, white Gaussian noise is used with zero mean and different value of variance on the images. When the variance is 0.009, the images have quite visible noise artifacts. In this case, our method shows 90\% DAR in detecting the manipulation. Upon reducing the variance to 0.005, DAR is observed to be 75\% and when the variance is reduced to 0.0005, DAR is 92.5\%.

Colour reduction of the original image can be obtained by the uniform quantization of intensity value of the original image. Here, the colour channel intensity is reduced to 32, 64 and 128 levels from 256 intensity levels. Images obtained after colour reduction are almost imperceptible degradations comparing to the original images. The proposed method is applied to detect the manipulation and the obtained results are much satisfactory in all the cases.

In contrast adjustment post-processing attack image contrast is modified by mapping intensity of the whole input image to a new interval of lower and upper bound. Hence, the output image intensity has been reduced. Contrast adjustments in the range (0.01, 0.95) had almost hardly impact on the post-processed image but in the range (0.01, 0.8) the image look much darker than the original image. Our method detected the manipulated region accurately with average DAR of 95\% in all cases. The detection results of noise, colour reduction and contrast adjustment operations are shown in Table 4, and it can be observed that the proposed method is much robust against all above-mentioned post-processing operations.

4.3.2. Blurring and brightness change attacks

Another post-processing attack, which is used here to validate the proposed method is image blurring. Images are convoluted with $3 \times 3$, $5 \times 5$, and $7 \times 7$ averaging filter for blurring. Blurred images have the same dimension as the original images. The proposed method is used to detect the manipulated regions. After blurring, proposed method performs well in detecting the manipulated regions efficiently.

Forged images may also be subjected to other post-processing methods like brightness adjustments. Here, the experimentation is also performed on the images, which are subjected to the brightness changes. Here, three ranges [(0.01, 0.95), (0.01, 0.9) and (0.01, 0.8)] are used for brightness change. In the range (0.01, 0.95), images are almost imperceptible impact on appearance of post-processing while in the range (0.01, 0.8), the brightness changes and images are visually significantly brighter than the original one.

<table>
<thead>
<tr>
<th>Table 3. Accuracy of proposed method on different transformations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transformations</strong></td>
</tr>
<tr>
<td>Translation</td>
</tr>
<tr>
<td>Rotation</td>
</tr>
<tr>
<td>Scaling</td>
</tr>
<tr>
<td>Distortion</td>
</tr>
<tr>
<td>Combination</td>
</tr>
</tbody>
</table>

Prakash et al., Cogent Engineering (2018), 5: 1523346
https://doi.org/10.1080/23311916.2018.1523346

Page 15 of 19
4.3.3. JPEG compression

It is a fact that most of the manipulated images are stored in JPEG format. Hence, a manipulation detection method must be able to detect the manipulation associated with this form of compression. Here, the JPEG compression is employed on the Comofod (Tralic et al., 2013) dataset to justify the robustness of proposed method with different quality factors (between 30 and 90 in interval of 10). There are 360 (40 × 9) forged images from the dataset. In most of the cases, DAR is above 80%, which shows that proposed method is robust enough to detect the manipulation against JPEG compression.

4.3.4. Comparison and discussion

For further demonstration of the robustness against the post-processing operations of the proposed method, it is compared with the existing artifacts as shown in Figure 8. It can be observed from the Figure 8(a) that proposed method shows a better accuracy rate against JPEG compression and false positive rate is much less than the existing methods as shown in Figure 8(b). From Figure 8(c,d), it can be observed that when the blurring operation is applied, the detection accuracy of the proposed method is much satisfactory and false positive rate is less than the existing methods. In brightness change operation, it can be observed from Figure 8(e) that the accuracy of the proposed method is above 80% in all cases, and the false positive rate is less than the

Table 4. Detection results of post-processing attacks

<table>
<thead>
<tr>
<th>Post-processing operations</th>
<th>Number of images</th>
<th>Number of detected image</th>
<th>DAR(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise Adding ( \mu = 0 ) ( \sigma = 0.009 )</td>
<td>40</td>
<td>36</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>30</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>37</td>
<td>92.5</td>
</tr>
<tr>
<td>Colour Reduction (Intensity level)</td>
<td>32</td>
<td>38</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>39</td>
<td>97.5</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>40</td>
<td>100</td>
</tr>
<tr>
<td>Contrast Adjustments</td>
<td>(0.01, 0.95)</td>
<td>40</td>
<td>38</td>
</tr>
<tr>
<td>(Lower bound, (0.01, 0.9))</td>
<td>40</td>
<td>39</td>
<td>97.5</td>
</tr>
<tr>
<td>(0.01, 0.8)</td>
<td>40</td>
<td>37</td>
<td>92.5</td>
</tr>
</tbody>
</table>

Figure 8. Comparison of different techniques with DAR and FPR curve against different attacks. (a)-(b) JPEG compression, (c)-(d) Image blurring and (e)-(f) Brightness change.
existing methods as shown in Figure 8(f). Additionally, we compared proposed method with Chen et al. (Chen et al., 2013) and Silva et al. (Silva et al., 2015) methods under different geometrical transformation like scaling, rotation and combination. Chen et al. (Chen et al., 2013) has proposed a method for duplicate region detection based on Harris corner points and step sector statistics. This method is robust enough to geometrical transformations (like, JPEG compression and Gaussian noise). The step sector statistics method is proposed to give a unique representation of the each circle region around the interest points. Then matching of those interest points are performed to get the duplicated region. Silva et al. (Silva et al., 2015) proposed a method, which is based on multi-scale analysis and voting process of a digital image. SURF is used to detect the keypoints and then their descriptors are localized in the image. Next, these keypoints are matched and clustered. Multi-scale analysis is employed to detect the manipulated regions. This method is robust to rotation, scaling and partially robust to compression.

To control the number of interest points detected through Harris corner method the thresholding is done, which changes with image size. Because of this, the density of interest points cannot be controlled. However, in the proposed method the ANMS (Adaptive non maximal suppression) is used for controlling the density of interest points for any image size. The comparison results of proposed method with L.Chen (Chen et al., 2013) and Silva et al. (Silva et al., 2015) method is shown in Figure 9, and it is observed that the proposed method gives quite satisfactory results in all post-processing operations as comparison to existing state-of-arts method.

5. Conclusion
Copy-move forgery is common and carried out easily and accurately. In this paper, a technique is proposed based on Harris corner and ANMS for forgery detection. By using ANMS, the keypoint density can be reduced and the distribution of keypoints throughout image is better. Here, a wide range of post-processing operations are applied to validate the proposed method. From experimentation, it is observed that the proposed method works for most of the geometrical transformations (including rotation, scaling, translation, distortion and combination) with satisfactory results. It indicates the robustness and consistency of proposed method with less time complexity of $O(n^2)$ than the L.Chen et al. (Chen et al., 2013) methods. In future, we will try to improve the performance of the proposed method, which can also deal with image degradations (like JPEG compression, Gaussian noise, brightness change, image blurring, colour reduction and contrast adjustments).
Funding
The authors received no direct funding for this research.

Author details
Choudhary Shyam Prakash
E-mail: h.shyamprakash2008@yahoo.com
ORCID ID: http://orcid.org/0000-0002-9305-3472
Hari Om
E-mail: hariomkindia@gmail.com
Sushila Maheshkar
E-mail: sushila.maheshkar@yahoo.com
Vikas Maheshkar
E-mail: vikas.maheshkar@gmail.com

1 Department of Computer Science and Engineering, Indian Institute of Technology (ISM), Dhanbad, 826004, India.
2 Department of Computer Science and Engineering, National Institute of Technology, Delhi, India.
3 Division of Information Technology, Netaji Subhas Institute of Technology, Delhi, India.

Citation information
Cite this article as: Keypoint-based passive method for image manipulation detection, Choudhary Shyam Prakash, Hari Om, Sushila Maheshkar & Vikas Maheshkar. Cogent Engineering (2018), 5: 1523346.

References


Cardiff.


Cleveland, USA.


MIT Press.


© 2018 The Author(s). This open access article is distributed under a Creative Commons Attribution (CC-BY) 4.0 license.