Keypoint based comprehensive copy-move forgery detection

Anjali Diwan  |  Rajat Sharma  |  Anil K. Roy  |  Suman K. Mitra

DA-IICT, Gandhinagar, Gujarat, India

Correspondence
Anjali Diwan, DA-IICT, Gandhinagar, Gujarat, 382007, India.
Email: anjali.diwan@ieee.org

Abstract
Verifying the authenticity of a digital image has been challenging problem. The simplest of the image tampering tricks is the copy-move forgery. In copy-move forgery copied portion of the image is pasted on another part of the same image. Geometrical transformations are used on the copied portions of the image before pasting it for the tampered image to look realistic and visually convincing. To make it more complex, other processing approaches may also be applied in the forged region for hiding traces of forgery. These processings are the scale, rotation, JPEG compression, and AWGN. In this paper, an approach based on features of the CenSurE keypoint detector and FREAK descriptor is proposed. This combination has novelty in itself as it has never been used for this purpose before to the best of authors’ literature studies. CenSurE detectors are fast and give stable and accurate output even in the case of rotated images, which we club with binary descriptor FREAK. Hierarchical clustering and Neighbourhood search is applied in such a way that it can locate and detect multiple copy-move forgeries. The authors are hopeful that the proposed approach may be used in real-time image authentication and copy-move forgery detection.

1  |  INTRODUCTION

Copy-move duplication is the most conventional forgery method in the field of digital images. In this method, one can copy one or more parts of a digital image and then paste it in any other location of the same image. The copy-move is described by using replication of the surrounding area of a digital image and inserting to a different area in the same image, generally to conceal unwanted areas of an image or duplicate an object to change the message conveyed by it. In the present digital era, we encounter vast amounts of images on digital media. These images are a significant source of information, which could be forged either for fun or for spreading misleading information. Some examples of copy-move forgery are shown in Figure 1(a).

The photo of prayer ceremony along the Ganges River, which was the original winner of the prestigious National Geographic Photo Contest (2012) [1]. It was disqualified when discovered that it had cloned out an unpleasant plastic bag from the edge of the frame. (b) Malaysian national news agency Bernama published a photo of a crowd of supporters of Prime Minister Najib Tun Razak. It was noted that portions of the crowd had been duplicated to appear larger, Bernama admitted to the mistake and withdrew the photo. These examples are proof that a big question mark arises on the integrity and authenticity of the available images. It indicates that fake news is a severe threat to our society and democracy [2]. It is the responsibility of the digital forensic community to develop a technique to authenticate digital images quickly.

The main aim of digital image forensic is to upgrade a recognizable proof framework to compute the credibility of the image and to release the possibility of forgery. In recent years, many recognition techniques have been developed. Among all, passive techniques are prioritised by researchers as they do not need any prior data, but the image itself. As forgery can not be recognised visually, passive strategies analyse the underlying statistical characteristic of the digital image. In the detection process, the digital image characteristics are retrieved and examined to identify the counterfeit image [3].

A forged region might be distinguished if a grouping area delivered by coordinating sets with a similar relative change is quite high [4]. In copy-move forgery process copying and pasting are done on the same image; hence, colour, noise component, and intensity range and other properties will be almost the same. It makes forgery detection difficult when no clue about the attack is available other than the image itself. To make forgery or tampering look real some tricks to hide the footprint...
FIGURE 1 (a) National Geographic photo-contest. (b) Malaysian news agency

of forgery are used by virtue of some geometrical transformations such as scale and rotation of the copied regions before pasting them somewhere else in the same image. Some image processing methods, that is, enhancing brightness and texture are also used to cover up the malafide intention of forging or tampering with an image [5].

In this paper we propose a novel method of copy-move forgery detection using CenSurE keypoint [6] and FREAK [7] descriptor. The work aims to address forgery when the copy-move region is affected with scale change (small to large) and rotation, along with various processing. To address transformation second stage of the algorithm is developed based on the hierarchical clustering approach. Multiple forgeries are handled by generating a k-NN search together with the clustering of keypoints. The notable contributions of this paper are:

- We have used the CenSurE keypoint detector, which was not used earlier. Binary descriptor FREAK is used as it has many inbuilt advantages over many other descriptors, it enhances the accuracy of detection in the presence of transformation and post-processing in images. Both CenSurE and FREAK works very well in complex conditions like geometric transformation and post-processing.

- We have designed a combination of the nearest neighbour search along with hierarchical clustering. Properties of agglomerative hierarchical clustering (AHC) are incorporated for enhancing accuracy in the presence of several attacks.

- We are accessing keypoints for geometrical transformation. This approach makes sure any geometrical transformation should not get undetected.

The paper is organised as follows: Section 2 addresses the current work in the copy-move forgery detection. Section 3 describes background knowledge. The proposed methodology is described in Section 4. Results and their detailed discussion are in Section 5. Section 6 has a conclusion and future scope.

2 RELATED WORK

To detect copy-move forgery in the digital image, researchers have developed numerous techniques. Broadly these techniques are classified as block-based techniques and keypoint based techniques.

In block detection based technique, image is divided into overlapping or non-overlapping image blocks. Features are extracted for each block and calculated. In block-based technique, different block feature matching approaches are required, like lexicographical sorting, K-d tree, radix sort, hash value, euclidean distance [3]. The first work for copy-move forgery detection was done in different matching processes are done like exhaustive block search, exact block matching, auto-correlation and robust match. However, this method gives forgery detection but leads to false matching for large textured images. Paclaviet al. [8] have divided the image into the non-overlapping block, and multifractal parameters are used for feature extraction of the block and classify it by metaheuristic method. In this approach, they are not detecting forgery in the presence of geometrical transforms. A.diwan et al. [9] have used a feature of LPP for localising the copy-move area of the image. They have accurately detected and localised forgery in the presence of JPEG, AWGN. Unlike other block-based methods, they do not need to sort image feature vectors, which reduces computational complexity. Hosny et al. [10] have proposed QPCETMs based technique that works on the sub-sampled images. The study of the image object before feature extraction is an exciting step, but this approach may not be useful for smooth texture or images with a dense texture. Zhong et al. [11] two-phase hashing feature is used for representing and searching image features. Block features are extracted using the normalised moment transform before applying the hash function. Gani et al. [12] Cellular Automata are applied to each DCT block feature of the image. The time complexity of the method is very high.

It has been observed that block-based techniques are often computationally expensive. Moreover, they are not invariant to geometrical transformation and other manipulation like flipping (180 degrees) shift, and blur. A keypoint based technique can address all these problems [3]. In this technique, keypoints of the images are used for finding forgery in images. There are many keypoint based forgery detection techniques that have been used by researchers in the past. Warif et al. [13] have used SIFT and mirror SIFT for addressing scale and rotation. They have used LoG for the reduction of computational cost. Muzaffer et al. [14] have proposed binaries SIFT, where each element
of SIFT is binaries. They claim to increase the overall speed of detection. Wan et al. [15] proposed to use the SIFT descriptor in such a way that forgery in smooth images could also be detected effectively. Beijing Chen [16] has used a Fractional quantisation moment and patch-match scheme to detect copy-move forgery. Ke Liu et al. [17] have used combined feature extraction using local intensity order pattern and SIFT keypoint. The nearest neighbour match is used for matching keypoints. Chen et al. [18] have used SIFT features and Invariant moments. A block of matched keypoints is further used for matching. Elhaminia et al. [19] used Markov random field for forgery detection. Over segmentation is done on images before the clustering of a similar region. Then SURF and PCT features are extracted from labelled keypoints. Wang et al. [20] have used SURF features and PCET for finding keypoints of images and detect tapered regions. They can detect geometrically transformed image but for higher scale change is a challenge which they can not detect. keypoints of the image are intersected with additional operations to improve detection accuracy in the presence of different attacks. Wang et al. [21] have used keypoint and segmentation for addressing images with a different texture. SURF keypoint, along with PCET, is used. Forged images with scale are difficult to detect using this method; also, the time complexity is a little high in specific steps. Selvaraj et al. [22] have worked on the SIFT feature along with sensitivity based clustering for copy-move forgery detection. Armas et al. [23], this work is based on the two approaches ELA and colour filter array, and they need modified and unmodified both images. With the incorporation of two different ways of analysis, they can detect copy-move along with splicing. Meena et al. [24] calculated the image features in two steps, first by SIFT keypoint detector for textured region and then by using FMT (Fourier miller transform) for the smooth region. FMT is used for the whole image if the number of in liners is less than a threshold.

There are many keypoint based techniques available that can detect forgery for various attacks. Nevertheless, they miss some of the attacks in copy-move forged images. We need an approach that should be efficient enough to detect a forgery in the presence of different attacks. In this paper, we propose an approach by which we can address all the attacks like scale, rotation, JPEG compression, and AWGN. The proposed approach is robust to multiple copy-move besides single copy-move forged images.

3 Basics of Feature Detector and Descriptor

In keypoint based copy-move forgery, extracted keypoints should be sparse, repeatable, and discriminating to maximise matching accuracy. Hence, the detector's repeatability measure can be used to find the best detector for copy-move forgery detection. The repeatability measures are defined as the extent to which the detected regions of the image are the same. It is based only on feature geometry. Consider a pair of images. The detector repeatability rate for the pair is defined as the ratio between the number of detection simultaneously present in both images, that is, repeated detection, and the total number of detection in the images. When copy-move parts of the images are the same corner detector can be used. However, when we have a copy-move part with different scale and rotation, we need invariant features, such that SIFT [25], SURF [26], and CenSurE. SIFT and SURF can address images with rotation and scale but up to a certain extent. We needed a keypoint detector with higher repeatability for significant angle rotation and scale. CenSurE performed better in the large degree of rotation. Though, it does not give good results in a massive-scale change.

In Figure 2, the repeatability of SIFT [25], SURF [26], Harris, and CenSurE are shown. The first graph represents the repeatability of detectors for change in the copy-move image scale. The second graph is for copy-move images with different degrees of angle rotation.

This section describes the fundamentals of the detector and descriptor used in the proposed approach. A small description of agglomerative hierarchical clustering is given concerning to keypoint.

3.1 CenSurE Keypoint Detector

Corner detectors are considered efficient in getting well localise keypoints in images. Though corner detector keypoints are accurate; however, they are not stable for scale change. Creating an image pyramid is another way by which image features are created. Nevertheless, scale features are poorly localised at a higher level of the pyramid. Hence, these features can be detected, but they are not effectively stable as corner detectors. To get stable scale-space features, we have used the CenSurE detector. This detector generates features so that all pixels of the image are integrated for all different scales. This detector determines large scale features by computing all features at all scales by selecting extrema across all locations and scales. There are three main steps for the CenSurE feature detector.

- The weaker responses are first filtered, and the response to a simplified bilevel Laplacian of Gaussian is computed to detect edges.
- At this stage, local extrema are detected.
- Harris measure is used to detect the local extrema with a strong corner response.

In copy-move forgery detection CenSurE features give higher accuracy than other detectors. We have experimentally evaluated some of the predominantly used keypoint detectors like SIFT [25], SURF [26], and ORB [27] and compared it with CenSurE.

3.2 FREAK Keypoint Descriptors

Image keypoint descriptors are the representation of an image that simplifies the image by extracting useful information. The human retina inspires FREAK; it uses a sequence of one-bit difference of Gaussian (DoG). The sampling grid used is circular, with inner circles symmetrically distributed. Density is higher
near the centre of the circle, and it decreases exponentially with distance. Each sampling point is smoothed with a Gaussian kernel where the radius of the circle illustrates the size of the standard deviation of the kernel. \( F \) is a binary string formed by a sequence of one-bit Difference of Gaussian (DoG), as shown in Equation (1).

\[
\sum_{0 \leq n \leq M} 2^n T(P_n) = F. \tag{1}
\]

Here pair of the receptive field is denoted by \( P_n \) and \( M \) is the size of the descriptor, which can be set as per requirement.

\[
T(P_a) \begin{cases} 
1 & \text{if } I(P_{a1}) - I(P_{a2}) > 0 \\
0 & \text{Otherwise}
\end{cases} \tag{2}
\]

Each pair of the receptive field is smoothed by Gaussian filter as given in Equation (2).

\( I(P_{a1}) \) is first field of pair \( P_a \).

FREAK is built using intensity comparisons of a set of 512 sampling pairs. Here two circles are selected randomly and then looking for the pairs that give more information. For each pair, if the first point has a greater intensity than the second, then one is written else 0 is written to the corresponding bit of the descriptor. Performance can be adjusted by changing the size of the Gaussian kernels or overlapping the receptive fields. More information is captured by overlaps, which tend to increase performance. Also, the pixels are being averaged, and are much more concentrated near the keypoint. This leads to a more accurate description of the keypoint. The way we have experimented with different keypoint detectors for our work. We have used some of the prominent descriptors to evaluate its response in terms of accuracy and time. Descriptor compared with FREAK are SIFT [25], SURF [26] and BRISK [28]. Detail description of the work, along with results, are shown in section V.

3.3 Clustering of keypoint

Agglomerative hierarchical clustering is an iterative classification method. This successive clustering operation produces a binary clustering tree (dendrogram), whose root is the class that contains all the keypoints. This dendrogram represents a hierarchy of partitions. It is then possible to choose a partition by truncating the tree at a given level, depending upon user defined constraints (the user knows how many classes are to be obtained).

In this technique, initially, each data point (keypoint) is considered as an individual cluster. At each iteration, the similar clusters merge with other clusters until one cluster or K clusters are formed. At first, dissimilarities are calculated between N keypoints, then two keypoints that, when clustered together, minimise a given agglomeration criterion, thus creating a class comprising these two keypoints. Then the dissimilarity between this class and the \( N_2 \) other keypoint is calculated using the agglomeration criterion. The two keypoint or classes of objects whose clustering together minimises the agglomeration criterion are then clustered together. This process continues until all the keypoints have been clustered.

It is vital to find a similarity between two clusters to merge them or divide them. There are specific approaches by which we can find similarities between clusters. These are the ways by which the linkage between the two clusters is established. These are single linkage, complete linkage, and average linkage. Complete linkage and average linkage approach are biased towards the global structure of clusters. Whereas the single linkage merge approach is local, and hence, it merges two clusters based on their similarity. Considering local information while finding similarity between two clusters for merging is advantageous for copy-move tapering detection.

In single linkage hierarchical clustering, the distance between two clusters is defined as the shortest distance between two
points in each cluster, as shown in Equation (3).

\[
L(r, p) = \min(D(x_r, x_p)). \tag{3}
\]

In complete linkage hierarchical clustering, the distance between two clusters is defined as the longest distance between two points in each cluster, as shown in Equation (4).

\[
L(r, p) = \max(D(x_r, x_p)). \tag{4}
\]

In average linkage hierarchical clustering, the distance between two clusters is the average distance between each point in one cluster to every point in the other cluster, as shown in Equation (5).

\[
L(r, p) = \frac{1}{n_r \cdot n_p} \sum_{i=1}^{n_r} \sum_{j=1}^{n_p} D(x_r, x_p). \tag{5}
\]

4 | METHODOLOGY

The proposed method is based on an effective and precise keypoint technique for image copy-move forgery recognition and localisation. The proposed algorithm is accomplished progressively better execution regardless of whether the copy-move imitation has just smooth or little territories or the produced image has been handled by scarcely any troublesome assaults (e.g. huge scale resizing and overwhelming commotion expansion). Figure 3 shows keypoint based digital image forgery detection strategy.

1. Define detector (CenSurE) and descriptor (FREAK)
2. Read ground truth and defining factor for clustering.
3. Detect keypoint from the image using \(K_s\) detector.
4. Use the FREAK descriptor for all the keypoints to find the k-NN feature.
5. Discard the nearest keypoint up to threshold so that it will not take similar points near it.
6. Operate agglomerative clustering.
7. Count the number of transformations on the image.
8. Plot forged region with points based on clustering procedures.

By completely utilising the strength qualities (counting the scale data and key direction) and the shading data of every keypoint, our proposed strategy achieves great exactness of location results at an extensively decreased computational expense.

4.1 | Proposed work

In the test image a set of keypoints \(X = x_1, x_2, ..., x_n\) are detected and their corresponding descriptor \(f_1, f_2, ..., f_n\) are extracted. Matching operation is performed among the keypoint descriptor vectors to identify the similar patches in the test image. The best candidate match is found by the Nearest neighbour search for each keypoint concerning all other keypoints of the image, so the Euclidean distance between the two descriptors is calculated. Matching two keypoints means, finding whether two descriptors are the same or not. Evaluating the distance between two descriptors by setting a global threshold gives the desired outcome. It happens due to the high dimensionality of the feature space, where some descriptors must be significantly discriminative than others. Hence, we used the ratio between the closest neighbour to that of the second closest one, which is compared with a threshold. Further we defined a similarity vector \(S = d_1, d_2, ..., d_n\) - 1 of the keypoint, which represents the shortest Euclidean distance with respect to other descriptors. Following the above idea, the keypoints are matched if the 2-NN test is satisfied as in the equation below:

\[
\frac{d_1}{d_2} > T \quad \text{where,} \quad T \in 1, 0. \tag{6}
\]

This procedure detects copy-move forgery effectively. It fails when a copied region is moved at multiple regions of the image, that is, multiple copy-move. Multiple copy-move forged image is one of the critical forgeries which we need to address for effective forgery detection. To deal with multiple copy-move generalised 2-NN (g2NN) search is used. g2NN is a generalised version of Equation (6). The g2NN test’s scheme is that the ratio between the distance of the candidate match and the distance of the second nearest neighbour. This match is considered to be low if the match is lower than the threshold and very high if the match is greater than the threshold. Our generalisation consists of iterating 2NN test on two similarity vector until the ratio is greater than \(T\), that is, \(T < \frac{d_1}{d_2}\). In our experiment, we fount that the value of \(T = 0.5\) gives the best result. Hence, applying a k-nearest neighbour (k-NN) search gives us the flexibility of selecting the number of keypoint to be considered. Along with this, we can use a variety of distance


### TABLE 1
Average values of F1-measure for different datasets are examined for different cluster linkage

| Dataset    | Single Linkage | Complete Linkage | Average Linkage |
|------------|----------------|------------------|-----------------|
| CMFD       | 97.61          | 96.22            | 95.99           |
| Grip       | 95.12          | 95.94            | 94.53           |
| Coverage   | 97.50          | 97.01            | 96.78           |
| F600       | 97.14          | 96.89            | 96.57           |
| F220       | 98.41          | 97.54            | 97.01           |
| CoMoFoD    | 96.15          | 96.38            | 95.25           |

criteria like euclidean distance, hamming distance, Manhattan distance, Minkowski distance.

#### 4.2 Impact of clustering on detection

Iterating over keypoint in $X$, we obtain a set of matched points. Isolated points are eliminated, and rest are considered for further processing. To prevent false alarm in high textured images, we have designed further processes. The clustering of the keypoint is done, as explained in Section 3. We are studying clusters and find average distance among clusters. If clusters are very far in the tree, they will not be allowed to join. In the clustering process, deciding a final number of clusters is a crucial task because a coefficient is designed, which depends on the average distance of the cluster. The average distance of the clusters is taken, and an appropriate threshold is selected. Hence, a threshold $T_b$ is defined to stop clustering after a specific value of $T_b$.

As clustering significantly affects forgery detection, we have experimented on various test images. We eliminated clusters with low matched keypoint to reduce false alarm. We consider that an image has forged with a copy-move attack if we can detect two (or more) clusters with at least four or more pairs of matched points linking a cluster. Rotation and scaling transformations can be computed by considering the centroids of the two matched clusters. How to link a cluster is another question to be addressed; experiments have been done on various linkage approaches. The result is shown in Table 1. One of the best suited approaches for our problem is single-linkage clustering. In single-linkage clustering, the similarity of two clusters is the similarity of their most similar members. This single-linkage merge criterion is local, and it merges two clusters based on their similarity. Other clustering methods are complete linkage, average linkage, and Ward’s linkage. These linkage approaches are biased towards globular clusters. In copy-move tapering, it is useful to take the locality of a cluster; hence we have used single-linkage in finding similar keypoints in the test image.

#### 4.3 Geometric transformation detection

After getting keypoints detection transformations is done. The type of geometrical transformation between copy and move region is decided. An affine homographic matrix is prepared based on the coordinate information of both copy and move region. We first calculate homography. Then rotation and scaling calculation is done by using homographic matrix decomposition. At the same time, translation is calculated by finding the centroid of the cluster.

Outlier can effect estimation badly, so a random sample consensus algorithm (RANSAC) is applied. This algorithm randomly selects a set (in our case, five pairs of points) from the matched points and estimates the homography $H$. All the left keypoints are transformed and compared in terms of spatial distance concerning their matched points. A threshold $\alpha$ is defined, and distance of point above this threshold is considered as outlier. As shown in Figure 4, experiment is carried out for a valid range of threshold for all the datasets, and the best value of $\alpha$, that is 22 is selected for the best result.

#### 4.4 Dataset and error metrics

The assessments should be done on a variety of copy-move forged images to evaluate the algorithm’s effectiveness. Christlein et al. [29] prepared the Image manipulation Dataset (we call it CMFD) based on 48 original images. The average size of the image is about $3000 \times 2300$, and about 10% of pixels are duplicated. Various kinds of attacks, such as geometrical
transformation (scale and rotation), AWGN, JPEG compression, are performed on the copied region. Images range from uniform to highly textured areas. GRIP [30] dataset contains 80 source images where tampered images are of different texture. Some of the images in GRIP are smooth like sky, and sandy desert. We have also used MICC-F600 [31] dataset which consist of 440 original images, where image resolution size is 800 × 533 to 800 × 600. This dataset consists of copied images with various kinds of attacks like scale, rotation, and combination of scale and rotation. Large angle rotation and images with multiple copy-move are also there. MICC-F200 [31] dataset have 110 original test images, where image resolution size is 722 × 480 to 3888 × 2592. CoMoFoD [32] dataset contains 260 forged image sets in two categories (small 512 × 512, and large 3000 × 2000). Additionally, we have used the Coverage dataset [33]; in this dataset, realistic images with high texture are used. It has 100 original and 100 forged images. Image resolution size is 400 × 486. Various attacks are enlisted in Table 2.

We are focusing on the authentication of the image, whether the image is forged or authentic. To find the robustness of the proposed method, analysis of the result is done by creating a confusion matrix. Confusion matrix consist of true positive (TP), false positive (FP), true negative (TN), and false negative (FN). These values are used for finding true positive rate (TPR), false positive rate (FPR), and F1-measure. TPR or sensitivity tells us what percentage of the forged image is correctly identified. FPR is a fallout rate, and it tells us what percent of authentic image detected as forget. The receiver operator characteristic curve (ROC) is a vital evaluation metric for calculating the method’s performance. The ROC curve graphically shows the connection between sensitivity and specificity. The perfect result can discriminate between the forged images with 100% sensitivity and 100% specificities.

\[
TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN},
\]

\[
F_1 = \frac{2TP}{2TP + FP + FN}.
\]

5 | EXPERIMENTS AND RESULTS

| Attacks | Criterion | CMFD | MICC-F600 |
|---------|-----------|------|----------|
| AWGN    | Standard deviation | 0.020:0.020:1.0 | 0.020:0.020:1.0 |
| JPEG F600 | Quality factor | 20:10:100 | 20:10:100 |
| Rotation | Degree | 2:2:10, 180, 60 | 2:2:10 |
| Scaling | Ratio | 0.91:0.02:1.09 | 0.91:0.02:1.09 |

All the experiments are performed using OpenCV-Python public library. The computer system used has the Intel processor with core(TM)-i7, 8 GB RAM, and CPU of 2.80 GHz.

5.1 | Analysis of detector and descriptor

Experimental results of different detectors and descriptors for simple copy-move are shown Figure 5(a)–(f). Simple copy-move is an ideal condition where there is no postprocessing is done. As a baseline, we evaluated how these detectors and descriptors perform for ideal conditions. Performance analysis for all the detectors and descriptors is done for their default parameter setting. The objective is to examine the response of the detector and descriptor for different images. We have tested them for six different detests, that is, CMFD, GRIP, COVER-AGE, MICC-600, and MICC-220, CoMoFoD. These datasets have images with divers texture (smooth to highly textured) and varied forgery region. SIFT, SURF, BRISK, and CenSurE detectors are used for keypoint extraction, and SIFT, SURF, BRISK, and FREAK descriptors applied on these keypoints.

5.2 | Discussion of experimental results

In this section, we have evaluated the performance of the proposed algorithm. As discussed earlier, forged images are generated by applying several transformations to the copied region of the image and then moving to another region of the same image, following six types of attacks which we think are the most common attack type and for which we have some results from previous works available to compare with. It is to note that we have not considered cropping as a potential attack because it is as good as copy-moving a small region of the image.

1. Simple copy-move: This is an ideal situation where copy-move images are not gone through any processing.
2. Rotation: Copied region of the image is rotated by a certain angle and then pasted.
3. Scale: The copied region is scaled up or down and then pasted.
4. Combined transform: Various operations are applied in the copied region of the image before pasting on the image.
5. JPEG compression: Copied region undergo lossy JPEG compression before pasting on some of the region of the image.
6. AWGN: Additive white Gaussian noise is applied to the copied region before pasting on the image.

Simple copy-move are the images without any geometrical transformation and post-processing. It is the ideal condition where part(s) of the image is copied and pasted on another region of the same image. We are addressing single as well as multiple copy-move; some examples of single and multiple copy-move images are shown in Figure 6 and 8 respectively.

As discussed in Section 4.1, we are addressing multiple copy-move forgeries explicitly. Results for all datasets are shown in
Evaluation results for SIFT, SURF, ORB and CenSurE detectors with SIFT, SURF, BRISK and FREAK descriptors, respectively, for different datasets. (a) CMFD, (b) GRIP, (c) COVERAGE, (d) MICC-F600, (e) MICC-F220 and (f) CoMoFod.

Some results for copy-move forgery detection.

Table 3. Execution time per image is also specified in the table. TPR and FPR of the six test datasets show that the proposed approach maintains its accuracy through various images with different texture used for forgery. However, results show that GRIP dataset gives the lowest TPR, and it happens due to some highly smooth images of the dataset. We have compared the proposed algorithm with some recent copy-move forgery detection approaches, and the result is shown in Table 4. We can see
FIGURE 7  ROC curves for copy-move image datasets. (a) CMFD, (b) Coverage, (c) MICC-F600, (d) MICC-F220, (e) GRIP and (f) CoMoFoD

TABLE 3  Results and execution time of simple copy-move forgery detection on CMFD dataset

| Method          | F1-Measure | Execution Time (s) |
|-----------------|------------|--------------------|
| Pun et al. [34] | 92.97      | 74.17              |
| Line et al. [35]| 85.98      | -                  |
| Pun et al. [36] | 93.87      | 110.62             |
| Meena et al. [24]| 96.97      | 18.75              |
| Proposed        | 97.61      | 15.75              |

FIGURE 8  Multiple copy-move forgery detection results: (a) forged image, (b) ground truth, (c) detected image

TABLE 4  Comparison of experimental results of the proposed algorithm for simple copy-move forgery detection on various datasets

| Dataset    | Proposed | Chen [37] |
|------------|----------|-----------|
|            | TPR      | FPR       | F1      | TPR | FPR | F1 |
| CMFD       | 97.67    | 4.16      | 97.61   | 100 | 0   | 100 |
| GRIP       | 92.85    | 3.57      | 95.12   | 90.00| 10.42| 91.72|
| Coverage   | 98.38    | 3.57      | 97.50   | -   | -   | -   |
| F600       | 97.22    | 5.00      | 97.14   | -   | -   | -   |
| F220       | 100      | 0.00      | 98.43   | -   | -   | -   |
| CoMoFoD    | 96.87    | 0.00      | 98.43   | -   | -   | -   |
a significant advancement in detection accuracy and execution time per image. Detecting forgery in images with smooth background where number of detected keypoints are considerably low is one of the challenge we are addressing. Some examples of images with uniform background are shown in Figure 9.

The ROC curve for all the six datasets has been given in Figure 7(a)–(f). ROC curve shows a trade-off between sensitivity and specificity. The curve closer to the top left corner indicates better performance than the curve on the right side. An area under an ROC curve is a measure of the usefulness of the method in general. The perfect test can discriminate between the forged image with 100 percent sensitivity and 100% specificity. The area under the ROC curve of the perfect result is 1. ROC curve for all the six datasets show that the detection accuracy of the proposed approach for the forged image is high, and the miss detection rate is low.

Geometrical transformations like rotation and scaling are used to make the forgery look real. Various post-processing operations are also used to hide the footprint of forgery in images. These operations are JPEG compression, and additive white Gaussian noise (AWGN). The combined effect of all the attacks is also used to create complex effects to hide forgery and make it look sensible. The proposed algorithm can detect a forgery in the presence of these combined attacks. To evaluate the robustness of the proposed algorithm, we have calculated the average F1-measure and execution time for each attack. Results for forgery detection in the presence of rotation, scale and combined transform types of attack are shown in Table 5. The results of forgery detection in the presence of JPEG compression and AWGN are shown in Table 6. The proposed algorithm keeps very stable computational complex-

### Table 5

| Attack      | Value | Proposed Approach F1 | Execution Time (s) | Pun [34] F1 | Execution Time (s) |
|-------------|-------|-----------------------|--------------------|-------------|--------------------|
| Rotation    |       |                       |                    |             |                    |
| (angle)     | 2     | 94.91                 | 15.75              | 92.79       | 110.12             |
|             | 4     | 94.05                 | 15.75              | 94.40       | 113.63             |
|             | 6     | 94.99                 | 16.99              | 93.02       | 106.47             |
|             | 8     | 93.90                 | 17.01              | 93.21       | 121.60             |
|             | 10    | 93.88                 | 17.56              | 92.34       | 123.92             |
| Scale       |       |                       |                    |             |                    |
| (factor)    | 2     | 93.49                 | 16.29              | 92.36       | 122.23             |
|             | 4     | 93.95                 | 17.55              | 92.67       | 122.15             |
|             | 6     | 92.46                 | 17.89              | 91.39       | 124.02             |
|             | 8     | 92.0                  | 17.01              | 91.98       | 119.75             |
|             | 10    | 92.88                 | 18.15              | 93.81       | 108.07             |

### Table 6

| Attack      | Value | Proposed Approach F1 | Execution Time (s) | Pun [34] F1 | Execution Time (s) |
|-------------|-------|-----------------------|--------------------|-------------|--------------------|
| JPEG        |       |                       |                    |             |                    |
| (quality)   | 100   | 94.87                 | 15.75              | 92.23       | 114.12             |
|             | 90    | 94.87                 | 15.72              | 90.31       | 111.31             |
|             | 80    | 93.9                  | 15.84              | 87.51       | 113.17             |
|             | 70    | 93.9                  | 15.01              | 85.82       | 113.64             |
|             | 60    | 92.88                 | 13.99              | 83.19       | 113.77             |
|             | 50    | 92.88                 | 13.24              | 80.75       | 114.82             |
|             | 40    | 90.18                 | 13.97              | 78.98       | 123.89             |
|             | 30    | 89.01                 | 12.48              | 73.76       | 126.90             |
|             | 20    | 87.23                 | 12.71              | 68.40       | 122.48             |
| AWGN        |       |                       |                    |             |                    |
| (deviation) | 0.02  | 91.9                  | 16.01              | 80.80       | 129.43             |
|             | 0.04  | 91.9                  | 16.99              | 74.49       | 130.96             |
|             | 0.06  | 91.9                  | 17.82              | 70.8        | 131.95             |
|             | 0.08  | 91.0                  | 19.31              | 57.60       | 131.96             |
|             | 0.10  | 90.88                 | 19.99              | 50.07       | 132.88             |
detect these attacks accurately as shown in Figure 11. We have shown results of some images undergone through combined transform. Results of rotation by large angles and also combined transformation (rotation and scale) are shown in Table 7.

### Table 7

| Attack                | Value | F1    | Execution Time (s) |
|-----------------------|-------|-------|--------------------|
| Rotation (angle)      | 20    | 89.88 | 18.52              |
|                       | 40    | 87.65 | 18.50              |
|                       | 60    | 87.65 | 19.22              |
|                       | 180   | 84.88 | 21.76              |
| Combined effect       |       | 96.97 | 20.84              |

### CONCLUSION

This paper discussed image tampering investigation exploiting a novel method in which we used features of CenSurE keypoint detector and FREAK descriptor. CenSurE keypoint detectors are considered to be one of the fastest keypoint detectors. To reduce the estimation time further we used a binary descriptor called FREAK that ensured much lesser computational time. It may be noted that our proposed procedure which consists of clustering followed by the nearest neighbour search reduces procedural time complexity. All these ultimately result into improved efficiency and accuracy of the image forgery detection. We carried out experiments on six open-source datasets, that is, CMFD, GRIP, Coverage, MICC-F600, MICC-200 and CoMoFoD, which have a broad range of geometrically transformed and post-processed images. We experimented with six types of attacks, namely, simple copy-move, rotation, scale, combined transformation, JPEG and AWGN along with multiple copy-move. Results of all these experiments proved that our proposed method was superior.

As expected smooth images are the most challenging for keypoint based method. Experiments done on a large number of smooth images of the GRIP dataset showed that were able to identify that images were tampered with. However, there is still a scope of improvement in the detection of forgery in highly smooth images. A combination of CenSurE detector and FREAK descriptor provides us stable, scale, and angle invariant keypoints.

Other than highly smooth images, we are getting poor results in some of the cases of combined attacks such as rotation followed by scale. Future work is mainly focused on improving the selection of image features with the help of integration of other forensic approaches to include forgery in highly smooth textured images and large scale and rotated images. We also could not include another attack called splicing which is a kind of image forgery where multiple image sources are used for creating a forged image and then pasted on some other region of the image to create a real impression. We are hopeful to report soon a more robust algorithm which would take care of smooth region images and would also detect splicing attack along with the copy-move.
REFERENCES

1. Fourandixs, Photo tampering throughout history. http://pth.izitru.com/. Accessed 6 Feb 2020
2. Pand, H.: Digital forensics in a post-truth age. Forensic Sci. Int. 289, 268–269 (2018)
3. Teerakanok, S., Uehara, T.: Copy-move forgery detection: A state-of-the-art technical review and analysis. IEEE Access 7, 40 550–40 568 (2019)
4. Soni, B., et al.: CMFD: A detailed review of block based and key feature based techniques in image copy-move forgery detection. IET Image Proc. 12(2), 167–178 (2017)
5. Santos, G.L., et al.: A bio-inspired methodology for digital imaging forensic detection. In: 2019 XV Workshop de Visão Computacional (WVC), pp. 96–102. IEEE, Piscataway (2019)
6. Agrawal, M., et al.: Censure: Center surround extremas for realtime feature detection and matching. In: European Conference on Computer Vision, pp. 102–115. Springer, Berlin (2008)
7. Alahi, A., et al.: FREAK: Fast retina keypoint. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–517. IEEE, Piscataway (2012)
8. Pavlovic, A., et al.: Copy-move forgery detection based on multifractals. Multim. Tools Appl. 78(15), 20 655–20 678 (2019)
9. Diwan, A., et al.: Detection and localization of copy-move tampering using features of locality preserving projection. In: 2019 Fifth International Conference on Image Information Processing (ICIIP), pp. 397–402. IEEE, Piscataway (2019)
10. Kosny, K.M., et al.: Copy-for-duplication forgery detection in colour images using OPCETMs and sub-image approach. IET Image Proc. 13(9), 1437–1446 (2020)
11. Zhong, J.-L., Pun, C.-M.: Two-pass hashing feature representation and searching method for copy-move forgery detection. Inf. Sci. 512, 675–692 (2020)
12. Gani, Q., Qadir, F.: A robust copy-move forgery detection technique based on discrete cosine transform and cellular automata. J. Inf. Security Appl. 54, 102510 (2020)
13. Wanf, N.B.A., et al.: Sift-symmetry: A robust detection method for copy-move forgery with reflection attack. J. Visual Commun. Image Rep. 46, 219–232 (2017)
14. Mazzaffer, G., et al.: Recent keypoint based copy move forgery detection techniques. In: 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), pp. 1–7. IEEE, Piscataway (2017)
15. Jin, G., Wan, X.: An improved method for sift-based copy-move forgery detection using non-maximum value suppression and optimized y linkage. Signal Process. Image Commun. 57, 113–125 (2017)
16. Chen, B., et al.: Fractional quaternion Zernike moments for robust color image copy-move forgery detection. IEEE Access 6, 56 637–56 646 (2018)
17. Liu, K., et al.: Copy move forgery detection based on keypoint and patch match. Multim. Tools Appl. 78(22), 31 387–31 413 (2019)
18. Chen, B., et al.: Fractional quaternion cosine transform and its application in color image copy-move forgery detection. Multim. Tools Appl. 78(7), 8057–8073 (2019)
19. Elhamnia, B., et al.: A probabilistic framework for copy-move forgery detection based on Markov random field. Multim. Tools Appl. 78, 25591–25609 (2019)
20. Wang, H., Wang, H.: Perceptual hashing-based image copy-move forgery detection. Security Commun. Netw. 2018, 6853696 (2018)
21. Wang, C., et al.: An image copy-move forgery detection method based on SURF and PCET. IEEE Access 7, 170 032–170 047 (2020)
22. Selvaraj, P., Karuppiah, M.: Enhanced copy past forgery detection in digital images using scale-invariant feature transform. IET Image Proc. 14(3), 462–471 (2020)
23. Armas Vega, E.A., et al.: Passive image forgery detection based on the demosaicing algorithm and jpeg compression. IEEE Access 8, 11 815–11 823 (2020)
24. Meena, K.B., Tyagi, V.: A copy-move image forgery detection technique based on Tetrolet transform. J. Inf. Security Appl. 52, 102481 (2020)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60(2), 91–110 (2004)
26. Bay, H., et al.: Surf: Speeded up robust features. In: European Conference on Computer Vision, pp. 404–417. Springer, Berlin (2006)
27. Rublee, E., et al.: Orb: An efficient alternative to sift or surf. In: 2011 International Conference on Computer Vision, pp. 2564–2571. IEEE, Piscataway (2011)
28. Leutenegger, S., et al.: Binary robust invariant scalable keypoints. In: 2011 International Conference on Computer Vision, pp. 2548–2555. IEEE, Piscataway (2011)
29. Christlein, V., et al.: An evaluation of popular copy-move forgery detection approaches. IEEE Trans. Inf. Forensics Secur. 7(6), 1841–1854 (2012)
30. Cozzolino, D., et al.: Copy-move forgery detection based on patchmatch. In: 2014 International Conference on Image Processing (ICIP), pp. 5312–5316. IEEE, Piscataway (2014)
31. Amerini, I., et al.: A sift-based forensic method for copy-move attack detection and transformation recovery. IEEE Trans. Inf. Forensics Secur. 6(3), 1099–1110 (2011)
32. Trafic, D., et al.: Comofod new database for copy-move forgery detection. In: Proceedings ELMAR-2013, pp. 49–54. IEEE, Piscataway (2013)
33. Wen, B., et al.: Coverage—A novel database for copy-move forgery detection. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 161–165. IEEE, Piscataway (2016)
34. Bi, X., Pun, C.-M.: Fast reflective offset-guided searching method for copy-move forgery detection. Inf. Sci. 418, 531–545 (2017)
35. Lin, C., et al.: Region duplication detection based on hybrid feature and evaluative clustering. Multim. Tools Appl. 78, 20739–20763 (2019)
36. Bi, X., Pun, C.-M.: Fast copy-move forgery detection using local bidirectional coherence error refinement. Pattern Recogn. 81, 161–175 (2018). http://www.sciencedirect.com/science/article/pii/S0031320318301183
37. Chen, H., et al.: Copy-move forgery detection based on keypoint clustering and similar neighborhood search algorithm. IEEE Access 8, 36 863–36 875 (2020)

How to cite this article: Diwan A, Sharma R, Roy AK, Mitra SK. Keypoint based comprehensive copy-move forgery detection. IET Image Proc. 2021;15:1298–1309. https://doi.org/10.1049/ipr.2021.01205