EasyChair Preprint

№ 6828

An Image Forensic Technique Based on SIFT Descriptors and FLANN Based Matching

Megha Gupta and Priyanka Singh

October 10, 2021

An Image Forensic Technique Based on SIFT Descriptors and FLANN Based Matching

Megha Gupta and Priyanka Singh
Dhirubhai Ambani Institute Of Information And Communication Technology
Gandhinagar, Gujarat, India
Email:201911047@daiict.ac.in, priyanka_singh@daiict.ac.in

***Abstract*—Doctored images are prevalent everywhere since the easy availability of photo-editing tools. The research in image forensics focuses mainly on developing techniques that can help discriminate between doctored and legitimate content in an image. There are various kinds of forgeries possible in an image. Here, we present a robust algorithm for copy-move forgery detection(CMFD). We exploit the simple linear iterative clustering (SLIC) algorithm to divide the source image into non-overlapping, irregular-sized blocks and then use Scale Invariant Feature Transform (SIFT) to determine the feature keypoints with their descriptors. After that, keypoints between blocks are matched using Fast Library for Approximate Nearest Neighbors (FLANN). Forged regions are chalked out accurately employing some morphological operations and analysis using correlation coefficient. To prove the effectiveness of the proposed algorithm, we have tested it on four standard datasets and found out the proposed scheme is performing satisfactorily well. It is helpful after scaling, rotation, and JPEG compression operations too.**

***Index Terms*—Copy-move forgery, SLIC, SIFT, FLANN matching, Adaptive over-segmentation(AOS)**

I. Introduction

We live in a time when technology is advancing at a breakneck pace, and with every new development comes a new set of problems. In this case, the progress of technology has given everyone the capability to edit and manipulate images with ease, leading to many tampered images that are hard to trace, causing a loss of integrity of the image. This has plunged us into a digital dystopia where many crimes are happening unchecked, causing societal disruption. Crucial images that can serve as evidence for crime scenes are doctored, leading to bad decisions. This alarms for the continuous research in the field of image forensics to cope up with these research challenges.

Though many schemes are already present in the literature to deal with image forgeries, we still hear so many such forgery cases. Like a duplicate image where a gathering of warriors was copied to cover George Bush [Malathi et al., 2019]. A doctored image of a Malaysian politician Jeffrey Wong Su showing him as a knight to the Queen of England in July 2010 [Express, 2010]. He faced eviction from his party after such doctored images were made public. Another recent news was a fake photo shared on Facebook in 2020 to falsely claiming that the people in the photo are coronavirus victims in China. However, in reality, it was a photograph of an art project in Germany in 2014 [Garcia, 2020]. With high-end editing technologies, it has become challenging to keep pace with the kind of possible forgeries and their revelation [Express, 2017].

Over the last few years, their has been extensive research on digital image forgery detection [Al-Qershi and Khoo, 2013]. The image forgery detection methods can be classified into passive or active approach. Additional information apart from the image, such as pre-extracted or pre-embedded attributes, are required to detect a forgery in the active approach. It includes watermarking or digital signatures. Some characteristics are added to an image when the digital watermark is created. [Jarusek et al., 2019]. In the passive approach, no such information is needed. Hence, also referred to as the blind forgery detection. A general categorization of the image tampering detection schemes is shown in Fig. 1 [Antony and Devassy, 2018].

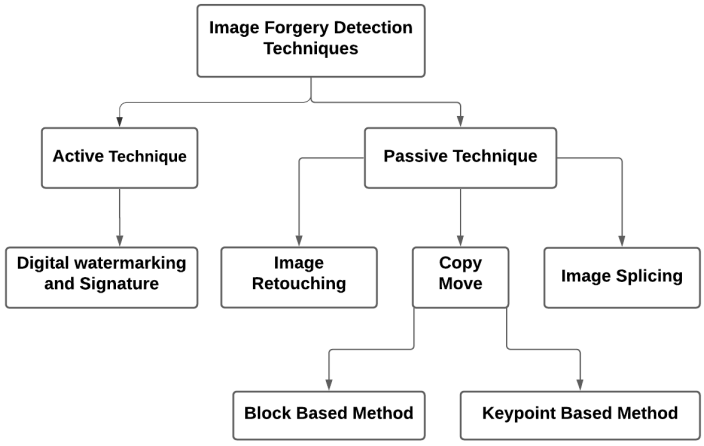

Fig. 1: Classification of image tampering detection approaches [Antony and Devassy, 2018]

There are numerous kinds of possible image forgeries but mainly can be categorized into: Image retouching, copy-move forgery, image splicing forgery. This paper focuses on copy-move forgery as it a commonly found forgery. A portion of an image is copied and pasted into other portions of the same image to hide some crucial details or mislead the viewer in this forgery.

The CMFD schemes can be grouped into keypoint-based techniques and block-based techniques. In block-based methods, the source image is segmented into blocks, and then the forged

regions are extracted by matching these blocks. Discrete Cosine Transform (DCT) [Popescu and Farid, 2004], PCA [Luo et al., 2006], and SVD [Mahdian and Saic, 2007] are the generally used block-based methods. Descriptors of keypoints are used in keypoint-based methods. Then, forged regions are extracted according to keypoint matching on the image. SIFT [Huang et al., 2008] and SURF [Bo et al., 2010] are commonly used keypoint-based techniques.

Cao et al. proposed a method for discovering CMFD by making fixed-sized blocks of the image. After that, computing Discrete Cosine Transform (DCT) of all blocks [Cao et al., 2012]. Keypoint descriptors are extracted for each block and sorted in lexicographic order, and matched to detect forged regions. Another scheme using the Polar Cosine Transform (PCT) and the approximate nearest neighbour was used in [Li, 2013]. Post-verification was performed to remove the false matches. However, segmenting the image into overlapping blocks increases the computational complexity.

Amerini et al. proposed a technique that used the J-Linkage algorithm to perform robust clustering in the geometric transformation space to recognise copy-move forgery [Amerini et al., 2013]. This method was better in precision rate and reliability. In [Pun et al., 2015], image forgery detection using the adaptive over-segmentation and keypoint matching was introduced. This scheme discarded blocks that contained only 1 or 0 key points leading to high false positives.

In this paper, we overcome this factor by exploiting the FLANN for matching blocks even with one key point. We applied the SLIC method to segment the image into irregular-sized non-overlapping blocks. After that, we applied SIFT for extraction of the feature keypoints along with their descriptors. The proposed scheme was examined on four standard datasets and worked efficiently, giving an F1 score of 97.7 % for MICC-F-2000, 94.57 % for MICC -600, 97.33 % for CPH 97.48 % for the CMFDB-GRIP dataset.

The remaining paper is divided into the following sections: Section II describes the related work. Section III has the description of the proposed technique. Section IV contains various experimental scenarios along with the details of the dataset. Section V includes conclusion and future directions.

II. Related Work

Many schemes are available in the literature to detect image forgery. Here, we briefly discuss some of the available schemes and their limitations.

Fridrich et al. proposed CMFD for JPEG images [Fridrich et al., 2003]. It was based on segmenting the host image into overlapping blocks and computing DCT. Similar DCT coefficients were identified and marked as forged regions of the image. Another related approach was proposed by N.D.Wandji et al. where they arranged the DCT based features in lexicographic order [Wandji et al., 2013]. This approach could detect the duplicated regions when there was more than one copy-move forged areas in the image. It was robust for shift, scale, blur, noise addition, slight rotations, and JPEG compression. The computational complexity was high for this method as it segmented the image into fixed-size of blocks.
An image forgery detection based on statistical correlations that appear in the case of forgery was proposed in [Popescu and Farid, 2005]. When a forger tries to create a forged image, he stretches, resizes or rotates the copied part to fit it properly into the target image. This attempt to resample the forged image on a new sampling lattice introduces specific correlations. These correlations were detected to validate the integrity of the image and used for the identification of forged parts. This scheme worked only on JPEG, GIF and TIFF images with minimal compression.
H. Huang et al. detected forged regions based on SIFT [Huang et al., 2008]. SIFT descriptors of an image are robust against changes in orientation, rotation, scaling etc. The SIFT method was used to recognized the key points and used these descriptors for matching. It gave good results even for a noisy or compressed image. Another scheme proposed by Zhang et al. detected multiple duplicated regions [Zhang et al., 2008]. It worked well for noisy and compressed images but unsuitable for rotational attacks.
SIFT is quite popular in many image processing techniques like facial recognition and object detection. Vijayan et al. combined FLANN feature matching with SIFT descriptors [Vijayan and Kp, 2019]. SIFT is robust to image distortion and the FLANN matcher matches the feature points. Bo et al. exploited the SURF descriptors for image forgery detection [Bo et al., 2010]. SURF descriptors of an image are invariant to changes in rotation, orientation, scaling etc. However, SURF does not give better performance than SIFT. Automatic image forgery detection based on approximate nearest neighbor leading to high performance was proposed in [Muja and Lowe, 2009]. Adapting k-means clustering to obtain superpixels was exploited by Achanta et al. in [Achanta et al., 2012].
Another brute force algorithm was proposed in [Antony and Devassy, 2018]. In this scheme, the characters were compared from left side to right side, until all characters were matched.

III. Proposed Methodology for image forgery detection

This section details the proposed method for detection of copy-move forgery. Fig. 2 shows an overview of the proposed scheme. The details of each step is discussed as follows:

Step 1: Adaptive over-segmentation of the host image

We have exploited the AOS method for dividing the source image into non-overlapping blocks of varying size. The AOS method is better for detecting forged regions than the traditional block-based approaches. Block-based methods can detect forgery at the block level, and hence, recall ratio is usually poor for such methods [Pun et al., 2015]. Also, the computational cost is directly proportional to the size of the image. We used SLIC to divide the image into significant irregular superpixels [Achanta et al., 2012]. The initial size of the superpixel is very crucial in accurately detecting the forged regions. The DWT method is used to obtain the initial superpixel size, as follows:

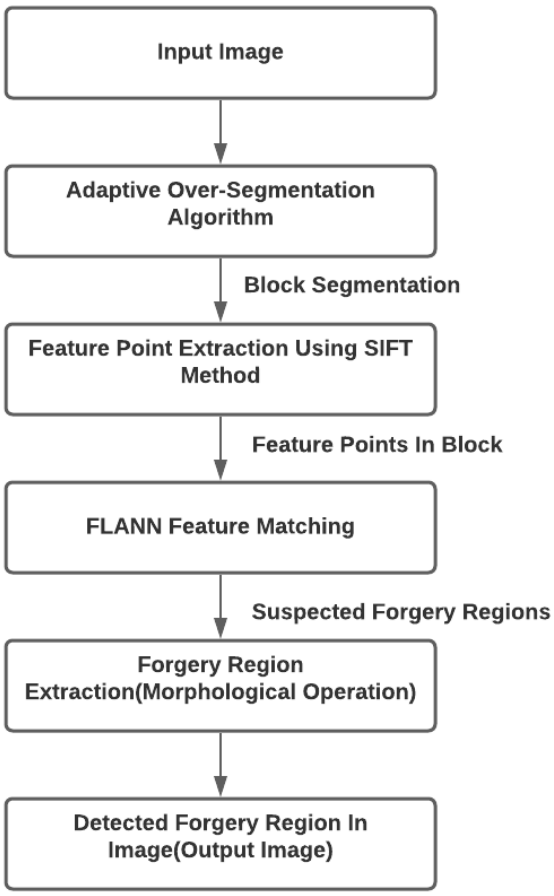

Fig. 2: Architecture of the copy-move image forgery detection method

We used SLIC to divide the image into significant irregular superpixels [Achanta et al., 2012]. The initial size of the superpixel is very crucial in accurately detecting the forged regions. The DWT method is used to obtain the initial superpixel size, as follows:

1) Apply 4-level DWT, on the source image and obtain the coefficients of the high-frequency energy and low-frequency energy:

$$Er_{LF} = |\sum CA_4| \tag{1}$$

$$Er_{HF} = \sum_i (|\sum CH_i| + |\sum CV_i| + |\sum CD_i|) \quad i = 1, 2, 3, 4 \tag{2}$$

where, Er_{HF} and Er_{LF} represent the high-frequency and low-frequency components.

2) Determine the low frequency distribution percentage Pt_{LF} :

$$Pt_{LF} = \frac{Er_{LF}}{Er_{LF} + Er_{HF}} * 100 \tag{3}$$

Algorithm 1 Image forgery detection scheme

INPUT: Host Image
OUTPUT: Detected Copy-move Forgery Regions

1: Apply 4 levels of DWT algorithm with haar as wavelet function to find out coefficients of high frequency energy and low frequency energy and then use these coefficients to find superpixel size.
2: Use the SLIC algorithm to segment the image into non-overlapping blocks of varying size using superpixel size found in the above step.
3: Use the SIFT algorithm to find location and descriptor of keypoints inside the superpixels.
4: Use FLANN algorithm for finding matching keypoints. Use the MatchThreshold, which we have taken 0.75 here, as threshold for distance between descriptors of keypoints to decide if they match or not.
5: Calculate TR_B according to the distribution of correlation coefficients. Filter the blocks matched in the previous step according to TR_B to find final matching points.
6: Again divide the source image into non-overlapping blocks using SLIC algorithm with superpixel size 20 for images with resolution greater than 1500 and 10 for low-resolution images.
7: Match the local color features of keypoints detected previously with neighbouring blocks in the new blocks. Include the blocks with similar color features in the forgery regions.
8: Apply morphological operation on forgery regions to generate final forgery regions.

3) Determine the initial size of superpixels S :

$$S = \begin{cases} \sqrt{0.02 * I * J} & \text{Pt}_{\text{LF}} > 50\% \\ \sqrt{0.01 * I * J} & \text{Pt}_{\text{LF}} < 50\% \end{cases} \tag{4}$$

where, I and J represent the size of the input image.

Step 2: Block feature extraction using SIFT
Feature points are take out from every image block using SIFT. SIFT features are based on the local features in an image and proven to be invariant to image processing operations such as compression, blurring, scale, and rotation [Huang et al., 2008]. It locates the keypoints in these image blocks. Thus, each image block comprises of these SIFT feature descriptors and the irregular block region information.

Step 3: Matching key points using FLANN
After obtaining the feature points, they are matched using FLANN. It is efficient for matching features with high dimensions or large datasets compared to other matchers like the brute force matcher. FLANN matches a feature based on the euclidean distance. Two keypoints are labelled as match only if the distance between their descriptors is smaller than the match threshold. We have set the match-threshold as 0.75 in the experiments.

The correlation coefficient map is created based on feature matching. The correlation coefficient CC tells the number of matched keypoints between two blocks. If N blocks are generated by adaptive over-segmentation, then there will be $\frac{N(N-1)}{2}$ correlation coefficients in the generated map.

After detecting the matched features, block-matching threshold TR_b is used to match the corresponding image blocks, which filters the false matched features, especially for those tampered parts that are similar to the background of the image. Every image has its own block-matching threshold, which depends on the features of the image.

To determine the block-matching threshold of an image, the first step is to sort the correlation coefficients in ascending order. We define $CC_s = \{CC_1, CC_2, CC_3, \ldots, CC_t\}$ where these values are sorted. Then, the first derivative, mean value of the first derivatives and second derivative of CC_s are calculated. The smallest correlation coefficient whose second derivative is greater than the average of first derivative vector is taken as the block-matching threshold:

$$f''(CC_s) > \overline{f'(CC_s)} \tag{5}$$

For the matched block pairs, the feature points are located and marked to localize the suspected forged regions.

Step 4: Localization of Forged Regions

To obtain the tampered regions, the superpixels are combined according to the local colour feature. In the experiments, the initial size S of the superpixels is 20 for high-resolution images, say 5000×5000 and to 10 for low-dimensional images, say 1500×1500.

To detect the suspected areas more accurately, for every suspected area, the local color feature of the neighboring blocks in varied orientations such as {45, 90, 135, 180, 225, 270, 315, 360} are computed [Pun et al., 2015]. Thereafter, it is compared with the local color feature of the corresponding forged area. If the difference between these local color features is less than the threshold TR_{sim}, then the neighboring blocks are merged with the suspected area. In the experiments, TR_{sim} is set to 15. Finally, small gaps in the merged regions are filled using the close morphological operation with a circular structuring element.

IV. Experimental Results

We conducted various experiments to validate the performance of the proposed algorithm on standard datasets. The experiments were conducted on a machine with Intel(R) Core(TM) i7-10750H CPU @2.60GHz, 64-bit processor, and 16GB RAM with Nvidia GeForce RTX 2060 on MATLAB R2019b. We used MICC-F-2000 [Amerini et al., 2011], MICC-F-600 [Amerini et al., 2011], Image Manipulation Dataset [Christlein et al., 2012], Copy-Move Hard (CMH) [Cozzolino et al., 2014], and CMFDdb_grip [Silva et al., 2015] as the standard datasets to test the proposed scheme. A brief description of these datasets is as follows:

1) **MICC Dataset:** This dataset is one of the earliest and the most commonly accessible datasets. This dataset consists of two subsets as MICC-F-2000 and MICC-600. Various operations such as scaling, rotation, compression have been performed to create tampered images. MICC-F-2000 consists of 686 and MICC-F-600 consists of 207 images.
2) **Copy-Move Hard (CMH) Dataset-** This dataset consists of images tampered with scaling and rotation operations set to various values. This dataset consist of four subsets i.e. CMH_{p1}, CMH_{p2}, CMH_{p3}, and CMH_{p4}. CMH_{p1} consists of images where the cloned areas are just copied and then moved, CMH_{p2} consists of images with a rotation of the copied area (orientations in the range of 90 to 180), CMH_{p3} consists of images with resizing of the copied part with a scaling factor in the range of 80% to 154%, and CMH_{p4} consists of images having both rotation and resizing at different values. This dataset consists of 108 images.
3) **CMFDdb_grip Dataset:** This dataset was made by Cozzolino et al. for measuring the performance of their CMFD algorithm. This dataset consists of images that have a slight change in the size of copy-move regions. Also, the size of images in this dataset is not very large. This dataset consists of 80 images

We evaluated the proposed scheme against various experiment scenarios. The following are details of each of these scenarios briefly:

1) The rotation operation is used on the tampered part in this scenario. Some portion of an image was copied, rotated by some angle and then pasted somewhere else on the original image. Results for some images having this scenario are shown in Fig. 3.
   - Dimension of the given image is 3264×2448 and the tampered part is inserted at the coordinate $(198, 1579)$ after rotating by 30°. Dimensions of copied part is 583×153.
2) The second scenario involves tampering the original image by compressing the JPEG quality. This implies that the forged image will be of a lower JPEG quality with respect to the original quality of the image. Results for some images considering this scenario are shown in Fig. 4.
   - In the given image, dimension of original image was 1024×768. The size of original image was 999 KB which was converted to 337 KB after JPEG compression reducing original image to 33%. The tampered part is inserted at $(591, 664)$.
3) The third scenario involves having a tampered image with scaling-down operation. A portion of an image is copied, scaled to a smaller size and then pasted on the other portion of the image. Results for some images considering this scenario are shown in Fig. 5.
   - In the given image, dimension of original image was 2048×1536. The tampered part is scaled by 0.57.

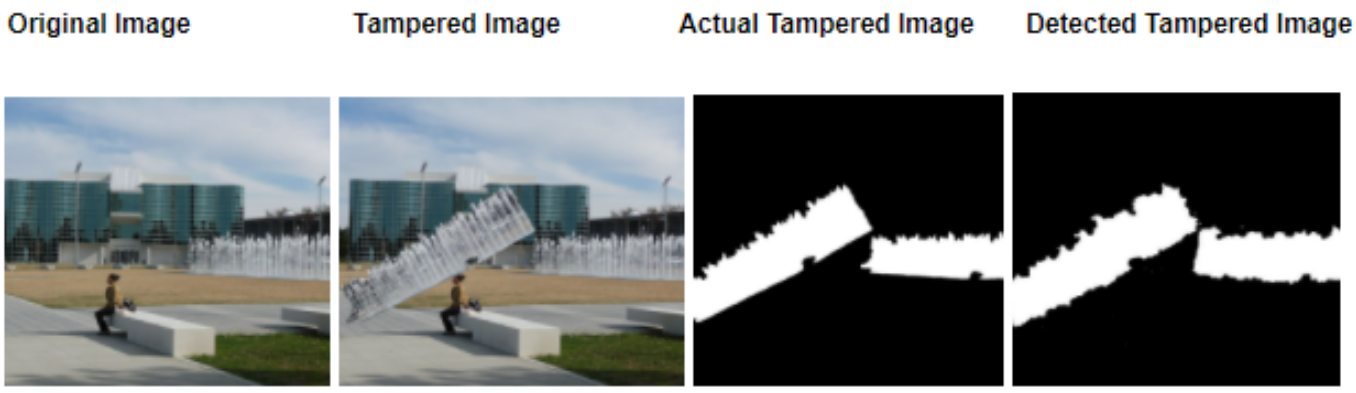

Fig. 3: Result of images with tampered part rotated

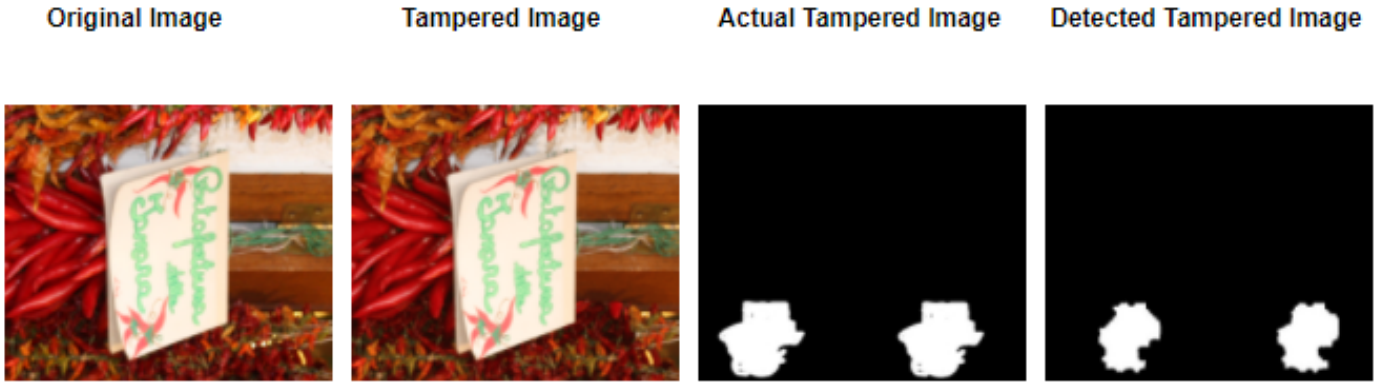

Fig. 4: Result of images after JPEG compression

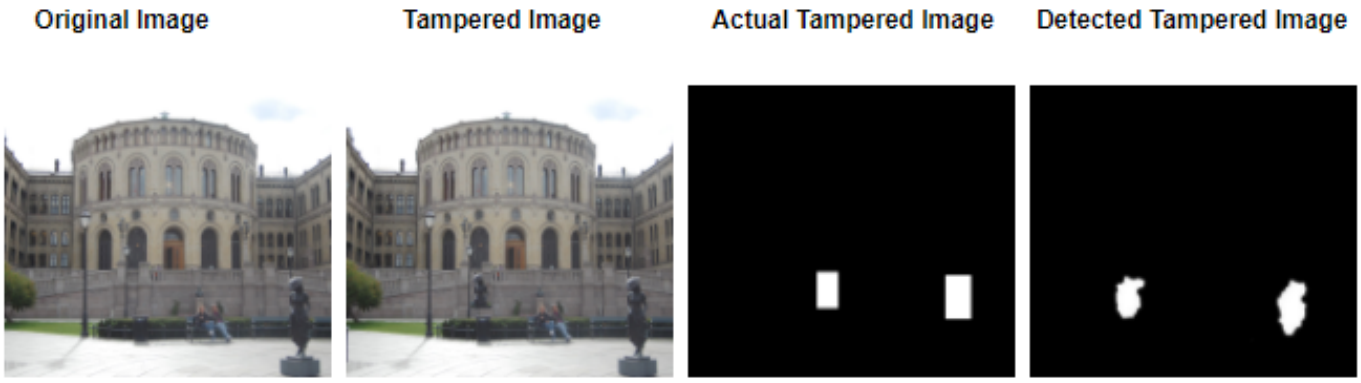

Fig. 5: Result of images with tampered part scaled down

The tampered part is inserted at $(807, 979)$. Size of original part was 142×214 which was scaled to 81×122.

4) The fourth scenario involves having a tampered image with scaling-up operation. A portion of an image is copied and then scaled to a larger size before pasting on the other portion of the image. Results for some images considering this scenario are shown in Fig. 6

- In the given image, dimension of original image was 2048×1536. The tampered part is scaled by 1.23. The tampered part is inserted at $(1164, 745)$. Size of original part was 187×111 which was scaled to

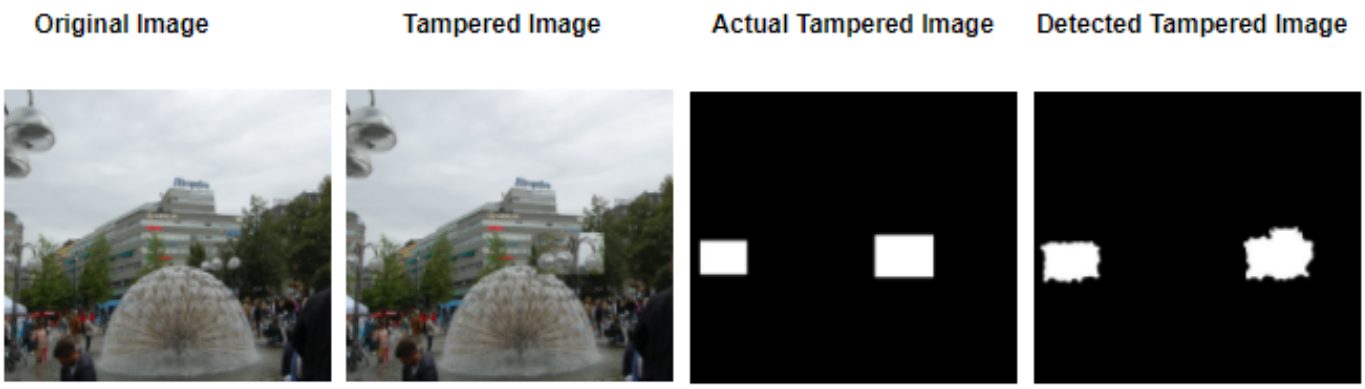

Fig. 6: Result of images with tampered part scaled up

230×137.

To judge the performance of the presented scheme in the experiment scenarios, we used recall and precision as the evaluation metrics [Amerini et al., 2011], [Christlein et al., 2012]. We defined precision as the ratio of the number of rightly detected tampered pixels to the number of identified forged pixels. It represents the probability that the parts detected by the algorithm are essential [Pun et al., 2015].

$$precision = \frac{|\Omega \cap \Omega'|}{|\Omega|} \tag{6}$$

where Ω represents the recognised tampered regions from the proposed technique from the dataset and Ω' represents the true tampered regions of the dataset.

The recall is the ratio of the number of rightly identified tampered pixels to the number of tampered pixels in ground truth tampered image. It represents the probability that relevant regions are identified [Pun et al., 2015].

$$recall = \frac{|\Omega \cap \Omega'|}{|\Omega'|} \tag{7}$$

F_1 measure is a reference number that is a combination of recall and precision to give a metric for the tampered detection [Singh et al., 2016].

$$F_1 = 2 \times \frac{recall.precision}{recall + precision} \tag{8}$$

Table I summarizes the results of the proposed scheme on the standard datasets MICC-F-2000, MICC-600, CPH, and CMFDdb_grip using the aforementioned metrics.

TABLE I: Forgery detection results for the proposed scheme

Datasets	Precision (%)	Recall (%)	F1 (%)
MICC-F-2000	95.81	98.67	97.07
MICC-600	93.82	96.24	94.57
CMH	96.30	94.79	97.33
CMFDdb_grip	95.80	99.34	97.48

The comparison between results of the proposed technique with the state-of-the-art algorithms are tabulated in Table II.

TABLE II: Comparison results of the proposed scheme

Datasets	Proposed Method F1(%)	[Pun et al., 2015] F1(%)
MICC-F-2000	97.07	75.74
MICC-600	94.57	86.41
CPH	97.33	92.97
CMFDdb_grip	97.48	97.55

V. Conclusion

This paper proposed an image forensic technique for detecting copy-move forgery in images. It extracted SIFT features for irregular sized non-overlapping blocks of the forged image and thereafter, exploited FLANN matching to detect forged regions based on keypoints matching between these blocks. The proposed method proved to be effective in performance under varied attack scenarios such as down-sampling, JPEG compression and geometric transforms. As a future study, the proposed method can be extended to tackle other types of forgery such as image splicing, contrast changes, noise addition, etc.

References

[Achanta et al., 2012] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282.

[Al-Qershi and Khoo, 2013] Al-Qershi, O. M. and Khoo, B. E. (2013). Passive detection of copy-move forgery in digital images: State-of-the-art. *Forensic science international*, 231(1-3):284–295.

[Amerini et al., 2013] Amerini, I., Ballan, L., Caldelli, R., Del Bimbo, A., Del Tongo, L., and Serra, G. (2013). Copy-move forgery detection and localization by means of robust clustering with j-linkage. *Signal Processing: Image Communication*, 28(6):659–669.

[Amerini et al., 2011] Amerini, I., Ballan, L., Caldelli, R., Del Bimbo, A., and Serra, G. (2011). A sift-based forensic method for copy–move attack detection and transformation recovery. *IEEE transactions on information forensics and security*, 6(3):1099–1110.

[Antony and Devassy, 2018] Antony, N. and Devassy, B. R. (2018). Implementation of image/video copy-move forgery detection using brute-force matching. In *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1085–1090. IEEE.

[Bo et al., 2010] Bo, X., Junwen, W., Guangjie, L., and Yuewei, D. (2010). Image copy-move forgery detection based on surf. In *2010 International Conference on Multimedia Information Networking and Security*, pages 889–892. IEEE.

[Cao et al., 2012] Cao, Y., Gao, T., Fan, L., and Yang, Q. (2012). A robust detection algorithm for copy-move forgery in digital images. *Forensic science international*, 214(1-3):33–43.

[Christlein et al., 2012] Christlein, V., Riess, C., Jordan, J., Riess, C., and Angelopoulou, E. (2012). An evaluation of popular copy-move forgery detection approaches. *IEEE Transactions on information forensics and security*, 7(6):1841–1854.

[Cozzolino et al., 2014] Cozzolino, D., Poggi, G., and Verdoliva, L. (2014). Copy-move forgery detection based on patchmatch. In *2014 IEEE international conference on image processing (ICIP)*, pages 5312–5316. IEEE.

[Express, 2010] Express, I. (2010). Malaysian politician faked photo of knighthood.

[Express, 2017] Express, I. (2017). 15 hoax stories that went viral.

[Fridrich et al., 2003] Fridrich, A. J., Soukal, B. D., and Lukáš, A. J. (2003). Detection of copy-move forgery in digital images. In *in Proceedings of Digital Forensic Research Workshop*. Citeseer.

[Garcia, 2020] Garcia, L. (2020). 5 tips we can all check coronavirus information.

[Huang et al., 2008] Huang, H., Guo, W., and Zhang, Y. (2008). Detection of copy-move forgery in digital images using sift algorithm. In *2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, volume 2, pages 272–276. IEEE.

[Jarusek et al., 2019] Jarusek, R., Volna, E., and Kotyrba, M. (2019). Photomontage detection using steganography technique based on a neural network. *Neural Networks*, 116:150–165.

[Li, 2013] Li, Y. (2013). Image copy-move forgery detection based on polar cosine transform and approximate nearest neighbor searching. *Forensic science international*, 224(1-3):59–67.

[Luo et al., 2006] Luo, W., Huang, J., and Qiu, G. (2006). Robust detection of region-duplication forgery in digital image. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 4, pages 746–749. IEEE.

[Mahdian and Saic, 2007] Mahdian, B. and Saic, S. (2007). Detection of copy–move forgery using a method based on blur moment invariants. *Forensic science international*, 171(2-3):180–189.

[Malathi et al., 2019] Malathi, J., Nagamani, T. S., Lakshmi, K. V., and devi, P. R. (2019). Survey: Image forgery and its detection techniques. *Journal of Physics: Conference Series*, pages 1–7.

[Muja and Lowe, 2009] Muja, M. and Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340):2.

[Popescu and Farid, 2004] Popescu, A. C. and Farid, H. (2004). Exposing digital forgeries by detecting duplicated image regions. *Dept. Comput. Sci., Dartmouth College, Tech. Rep. TR2004-515*, pages 1–11.

[Popescu and Farid, 2005] Popescu, A. C. and Farid, H. (2005). Exposing digital forgeries by detecting traces of resampling. *IEEE Transactions on signal processing*, 53(2):758–767.

[Pun et al., 2015] Pun, C.-M., Yuan, X.-C., and Bi, X.-L. (2015). Image forgery detection using adaptive oversegmentation and feature point matching. *IEEE Transactions on Information Forensics and Security*, 10(8):1705–1716.

[Silva et al., 2015] Silva, E., Carvalho, T., Ferreira, A., and Rocha, A. (2015). Going deeper into copy-move forgery detection: Exploring image telltales via multi-scale analysis and voting processes. *Journal of Visual Communication and Image Representation*, 29:16–32.

[Singh et al., 2016] Singh, S., Agrawal, S., and Singh, G. (2016). Accuracy detection of digital image forgery by using ant colony optimization technique. In *MATEC Web of Conferences*, volume 57, page 01014. EDP Sciences.

[Vijayan and Kp, 2019] Vijayan, V. and Kp, P. (2019). Flann based matching with sift descriptors for drowsy features extraction. In *2019 Fifth International Conference on Image Information Processing (ICIIP)*, pages 600–605. IEEE.

[Wandji et al., 2013] Wandji, N. D., Xingming, S., and Kue, M. F. (2013). Detection of copy-move forgery in digital images based on dct. *arXiv preprint arXiv:1308.5661*.

[Zhang et al., 2008] Zhang, J., Feng, Z., and Su, Y. (2008). A new approach for detecting copy-move forgery in digital images. In *2008 11th IEEE Singapore International Conference on Communication Systems*, pages 362–366. IEEE.