Research Article

Blind copy-move forgery detection using SVD and KS test

Belal Ahmed[1] · T. Aaron Gulliver[1] · Saif alZahir[2]

Received: 13 September 2019 / Accepted: 4 July 2020 / Published online: 15 July 2020

Abstract
In this paper, we present a new copy-move forgery detection method based on singular value decomposition (SVD) and the Kolmogorov Smirnov (KS) test. This work introduces a new method of detecting copy-move forgery in images with accuracy up to the pixel level using only 4 features per image block. The proposed method consists of three steps. First, an image is partitioned into blocks of size 16 × 16. Second, image features are extracted from each block using steerable pyramid and SVD transforms. Finally, the extracted features are sorted lexicographically and matched using the KS test. The performance of the proposed method is evaluated using the CoMoFoD database. Four post-processing techniques are considered, namely brightness change, contrast adjustment, color reduction, and image blurring. This method achieved a high precision of more than 95% for 3 of the 4 post-processing techniques. The fourth post-processing (i.e., image blurring), we achieved a precision of 75% which is considerably high for such forgery. In addition, the proposed method outperformed published methods when the images were subjected to brightness change, contrast adjustment, color reduction and image blurring. Finally, the performance of the proposed algorithm shown to provide better precision using fewer features compared to several well-known techniques in the literature.

Keywords Copy-move forgery · Singular vector decomposition (SVD) · Kolmogorov Smirnov (KS) test · Steerable pyramid

1 Introduction

Digital forensics is a branch of forensic science which deals with the problem of determining the authenticity of digital data. Digital data such as images play a significant role in digital forensics since they are a main source of information. With the advent of image editing software, digital images can easily be manipulated. Tampered images can be found everywhere and this has eroded confidence in the reliability of digital images.

Digital image forensics can be categorized into two groups: active and passive. In the active approach [1], a digital watermark or signature is embedded into the image to verify its integrity and authenticity. A digital watermark can be visually undetectable but can be used to detect changes in image pixels and locate where the changes occurred [2]. However, watermark removal software is readily available at no cost and the images are still vulnerable to forgery. For this reason, passive methods have been introduced which require no prior information to detect tampering [3].

Images can be modified in several ways such as image splicing, retouching, and copy-move forgery [4]. Image retouching alters an image in order to change the look of a subject [5]. Image splicing refers to copying a part of an image and pasting it onto another image. Most techniques to detect image splicing rely on the sharp edges and corners of the pasted region and the inconsistency in the color of the pasted region compared to the original image [6]. In contrast, copy-move forgery refers to copying a part of an image and pasting it onto the same image. Since the pasted region comes from the same image, the

✉ Belal Ahmed, bahmed@uvic.ca | [1]Department of Electrical and Computer Engineering, University of Victoria, Victoria, Canada. [2]Department of Computer Science and Engineering, University of Alaska Anchorage, Anchorage, USA.

SN Applied Sciences (2020) 2:1377 | https://doi.org/10.1007/s42452-020-3181-6

color is typically consistent which makes it hard to detect. Thus, copy-move forgery is more difficult to detect than other types and is the focus of this paper.

In this paper, a new copy-move forgery detection technique is introduced which is based on singular vector decomposition (SVD) for features extraction and Kolmogorov Smirnov (KS) test for decomposition. The proposed method can detect copy-move forgery in images with accuracy up to the pixel level by using only 4 features per image block of dimension 16×16. Four different post-processing techniques are considered, namely brightness change (BC), contrast adjustment (CA), color reduction (CR), and image blurring (IB). Comparisons are conducted with several well-known techniques in terms of accuracy, time complexity, and image post-processing using the well-known CoMoFoD dataset [7].

The remainder of this paper is organized as follows. Section 2 provides a review of existing approaches for copy-move forgery detection. Section 3 explains the structure of block-based methods and a review of the existing block-based methods. Section 4 discusses the proposed method. The experimental results, comparisons and analysis are given in Sect. 5. Finally, the conclusions are drawn in Sect. 6.

2 Copy-move forgery

Copy-move forgery has a long history beginning in the early 1900s when Alexander Malchenko (standing, on the left) was edited out from the image as shown in Fig. 1. This task was complicated due to the limited tools available at that time. In contrast, there are now tools that can easily perform this task such as Adobe Photoshop which is available at no cost.

Copy-move forgery is usually used to maliciously hide or add information as in Fig. 1. Since the forged region comes from the same image, the resulting image has at least one duplicated region. The goal of detection methods is to determine these regions. Searching for duplicated regions by comparing pixels in the image is a direct solution but is slow and computationally expensive. Further, post-processing techniques such as brightness change (BC), color reduction (CR), contrast adjustments (CA), and image blurring (IB) can be used on the image or just the copied region to make the forgery harder to detect. This makes copy-move forgery detection a challenging task.

Copy-move forgery detection can be keypoint-based or block-based. In block-based methods, an image is partitioned into fixed size overlapping or nonoverlapping rectangular or circular blocks. A feature vector is extracted for each block and these vectors are matched by calculating the distance between them. This distance can be Euclidean distance [8, 9], Hamming distance [10], Hausdorff distance [11], logical distance [12], correlation coefficient [13, 14], phase correlation [15, 16], or local sensitive hashing [17, 18]. The main concern with block-based methods is their computational complexity.

Keypoint-based methods detect and describe local features in an image using techniques such as the scale invariant feature transform (SIFT) and speeded up robust features (SURF) [19]. These features are used to find matching regions in the image and if two regions have similar keypoint features, one is assumed to be forged. However, these methods can fail when the forged regions have been modified using techniques such as image retouching [20]. For this reason, block-based methods are employed here.

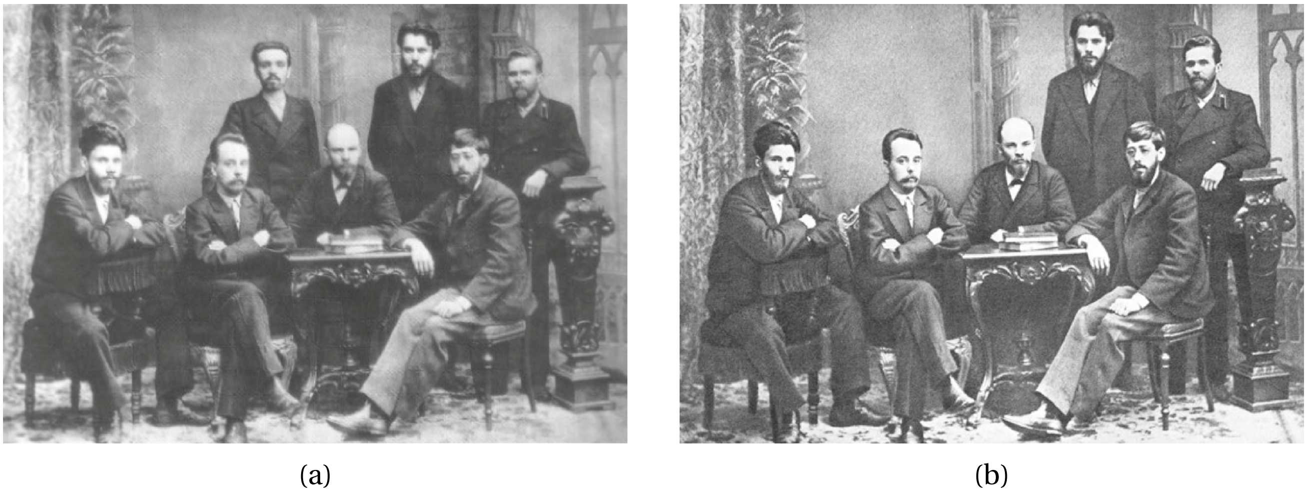


Fig. 1 Alexander Malchenko has been edited out: **a** original image, and **b** forged image

The next section presents a review of copy-move forgery detection using these methods.

3 Block-based methods

A number of block-based methods have been proposed to detect copy-move forgery [21, 22]. The goal is to find similar regions in a forged image.

3.1 Preprocessing

Some block-based methods require the image to be preprocessed before any further analysis. In [5, 15, 18, 22, 23] an image was converted to grayscale using the luminance formula $Y = 0.299R + 0.587G + 0.114B$, where R, G, and B represent the red, green, and blue luminance respectively, of the colour image. Each color is represented by a vector which provides the brightness intensity, for example 0 to 255 for 8-bit images.

3.2 Image partitioning

In block-based methods, an image is partitioned into fixed size overlapping or non-overlapping blocks. Square block partitioning was used in [5, 17, 21, 22, 24]. Circular partitioning was used in [9, 14] to reduce the dimension of each block compared to square blocks which lowers the computational complexity. After an image is partitioned into blocks, features can be extracted from each block to find those with similar features. Other methods operate directly on the blocks without feature extraction such as those in [25].

3.3 Feature extraction and matching techniques

Copy-move forgery detection is used to find replicated regions in an image. However, the copied region can be processed prior to pasting into another part of the image to make it harder to detect. The detection accuracy depends on finding matching features in a copy-move pair of blocks in an image even if the image has undergone post-processing such as blurring, color reduction, brightness change, or contrast adjustment [26]. To accomplish this goal, several procedures have been proposed.

In [21], a method for copy-move forgery detection was given which is based on the discrete cosine transform (DCT). In this approach, DCT coefficients are extracted for each block. To reduce the complexity, the coefficients are sorted lexicographically and then quantized to reduce the effects of noise, JPEG compression, brightness change or other post-processing. After quantization, the coefficients for each block are compared and a block is said to be copied if these coefficients are comparable. To reduce the false positive rate, a region is detected as forged only if there is a cluster of copied blocks in the region. However, this may result in small forged regions going undetected.

In [27], the DCT coefficients were considered as eigenvalues for each block. Then, the distances between the eigenvectors of all blocks were calculated and lexicographically sorted to reduce the false positive rate [21]. If the distance between two blocks is less than a certain threshold, the block is considered to be a duplicate. This method is robust to noise and JPEG compression but not to other post-processing operations. Comparing block features makes block-based partitioning methods complex. As a result, several techniques have been developed to lower the complexity by reducing the length of the feature vectors. Cao et al. [24, 28] reduced the feature vector length by using circular blocks instead of square blocks. Each block is divided into four regions. DCT is applied to each region so each is represented by DCT coefficients. The mean of the coefficients in each region is used as a feature resulting in 4 features per block. This method is robust to noise and blurring, and can detect multiple forgeries in the same image. However, it is not robust against rotation or scaling.

Several block-based algorithms have been proposed to reduce the feature vector length while being robust to post-processing operations. A log-polar transform was used in [14, 27, 29]. This transform maps the image blocks to log-polar coordinates [30]. This approach is robust to post-processing operations such as rotation, scaling, mirroring, illumination modification, and chrominance modification, but not against other techniques such as noise, JPEG compression or blurring.

SVD is a matrix factorization technique that can be used to extract features from an image. Li et al. [31] proposed an algorithm based on SVD and the Discrete Wavelet Transform (DWT). The DWT is used to decompose an image into a series of coefficients corresponding to the image spatio-frequency subbands and SVD is employed to extract the feature vectors. Then these vectors are sorted lexicographically to detect duplicated regions. Zhang and Wang [32] introduced a method which combines SVD with a k-dimensional (KD) tree. A KD tree is commonly used to search for nearest neighbours. First, SVD is used to extract a feature vector for each block, and these features are organized using a KD tree for fast searching. Similar blocks are matched using the Euclidean distance

$$D(u,v) = \left(\sum_{i=1}^{r} ((u(i) - v(i))^2 \right)^{\frac{1}{2}},$$

where $u = (u_1, u_2, \ldots, u_r)^T$ and $v = (v_1, v_2, \ldots, v_r)^T$ are length r feature vectors. Xiaobing and Shengmin [1] used an approach similar to that in [32], but only the k largest singular values were considered and the remaining values discarded to reduce the feature vector size. Kang and Cheng [1] also retained only the largest k values. Regions were matched using the cumulative contribution which is defined as

$$\eta = \left(\sum_{i=1}^{k}(\lambda_i)\right) \Big/ \left(\sum_{i=1}^{r}(\lambda_i)\right),$$

where λ is the feature vector sorted from largest to smallest and r is the number of singular values. However, this method is not robust to post-processing operations.

The size of the feature vectors is an important factor in the complexity of detection algorithms. Steerable pyramid is a technique used for image decomposition. In this method, an image is smoothed using a smoothing filter and subsampled by a factor of 2 along each coordinate. This process is repeated multiple times which looks like a pyramid if illustrated graphically. In [5], steerable pyramid decomposition was applied to image blocks to reduce the size from 16 × 16 to 4 × 4. Then, similar blocks were extracted using the Gaussian copula which is a distribution constructed from a multivariate normal distribution.

3.3.1 Kolmogorov–Smirnov (KS) test

Kolmogorov–Smirnov test is used to compare two samples based on a distribution function [33]. Given samples $x_1, \ldots, x_n$ of a random variable with cumulative distribution function (CDF) F, consider the problem of testing $H_0 : F = F_n$ versus $H_1 : F \neq F_n$, where F_n is some specified distribution function. For example, in the univariate case H_0 can be tested using the KS test statistic $sup_{x \in R} \mid F_n(x) - F(x) \mid$, where $sup_{x \in R}$ is the supremum of the set of distances, F is the CDF of the reference distribution, and F_n is the empirical distribution of the samples[34]. The test statistic measures the similarity between distributions and equals 0 when they are identical.

3.3.2 Steerable pyramid

Steerable pyramid is a mathematical tool that is multi-orientation, multi scale image decomposition technique. This transform was first introduced in the literature in early 1990s. It is a wavelet-based representation [35]. Steerability refers to the ability of the wavelets to rotate to any orientation by forming suitable linear combinations of a primary set of equiangular directional wavelet components [36, 37]. In steerable pyramid decomposition, an image is decomposed into lowpass and highpass subbands, using steerable filters L_0 and H_0. Then, the lowpass band breaks down into a set of bandpass subbands $B_0, \ldots, B_k$ and lower lowpass subband H_1. The resulting lower lowpass subband is subsampled by a factor of 2 along the x and y directions. This process is repeated until we arrive at the desired scale of decomposition. Figure 2 shows an image of size 128 × 128 pixels decomposed to its steerable pyramid subbands. This example shows 4 orientations and 3 scales. Each scale has 4 orientations. The first scale is 128 × 128 where the second and third scales are 64 × 64 and 32 × 32 respectively. Finally, the last subband is 16 × 16 pixels [36].

3.3.3 Singular value decomposition (SVD)

Singular value decomposition has been used in several fields such as data compression, signal processing and pattern analysis [1, 38]. An $N \times M$ matrix $A \in R^{N \times M}$ of rank j can be factored in the form

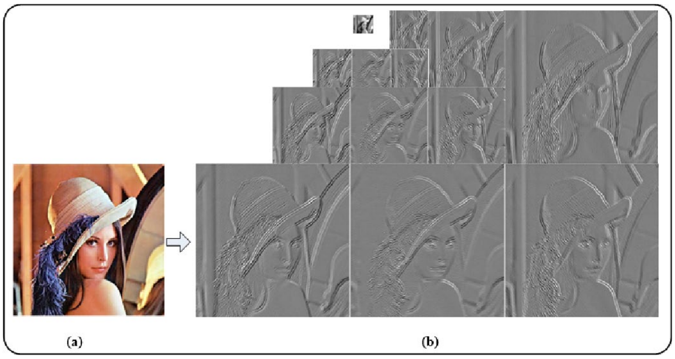
$$A = P \Sigma Q^T \tag{1}$$

where

$$\Sigma = \begin{bmatrix} \Sigma_j & 0 \\ 0 & 0 \end{bmatrix} \tag{2}$$

is an $N \times M$ diagonal matrix and $\Sigma_r = diag(\sigma_1, \sigma_2, \ldots, \sigma_j)$ is a square diagonal matrix with positive diagonal entries

Fig. 2 Lena Image and its decomposed steerable pyramid subbands



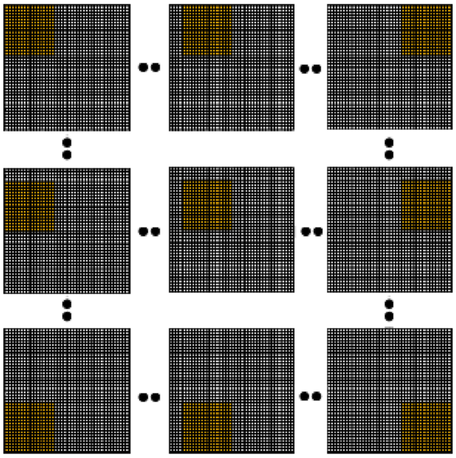


Fig. 3 Image partitioning into 16 × 16 overlapping blocks.

called the singular values of A, and orthogonal matrices $P \in R^{N\times N}$ and $Q \in R^{M\times M}$.

4 The proposed method

In the proposed method, an image is partitioned into blocks of size 16 × 16 using a 16 × 16 sliding window which is shifted by one pixel per step as shown in Fig. 3. This results in 246, 016 blocks for an image of size 512 × 512. To reduce the computational complexity, each block is decomposed into a 4 × 4 block using steerable pyramid decomposition. A 4 × 4 block has 4 nonoverlapping 2 × 2 sub-blocks. SVD is applied to each 2 × 2 sub-block to extract a single singular value which is the corresponding feature as shown in Fig. 4. This results in 4 features per 4 × 4 block. Thus, each original 16 × 16 block is represented by a vector of only 4 features. Figure 5 shows a block diagram of the feature extraction process. The indices of the original blocks are stored with the feature vectors so that each pixel is associated with a feature vector.

The feature vectors are sorted lexicographically so that similar vectors are close which simplifies the search process. The KS test is applied to the sorted feature vectors. The empirical and reference CDFs F_n and F are represented by the two feature vectors being compared. If the test statistic is below or equal to the threshold, the two feature vectors are said to match and belong to the same distribution. Conversely, i it is above the threshold, the vectors are said to differ and belong to different distributions. Pixels belonging to the same distribution share the same features and so one group is said to be forged.

Pixels in the same region often share the same features such as brightness and color. Thus, applying the KS test to the corresponding feature vectors may increase the false positive rate. To avoid this problem, a minimum distance between pixels is employed. The distance between two pixels is defined as

$$D = |x_2 - x_1| + |y_2 - y_1|$$

where (x_1, y_1) and (x_2, y_2) are the coordinates of the pixels. If $D \geq d$ where d is a minimum distance threshold, then the pixels are said to be in different regions. This threshold

Fig. 4 Feature extraction from 2 × 2 sub-blocks using SVD

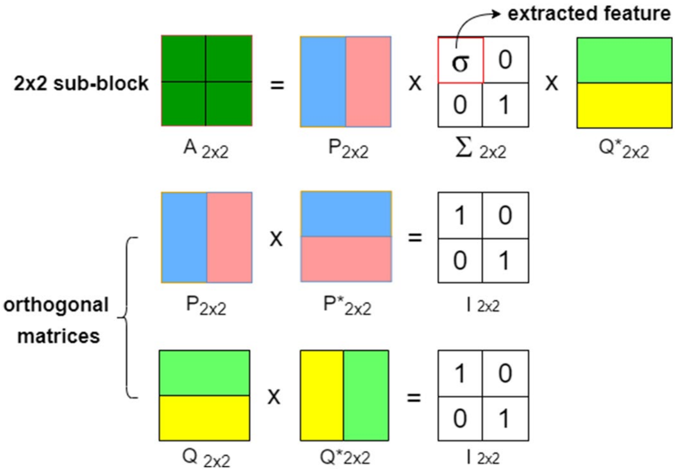
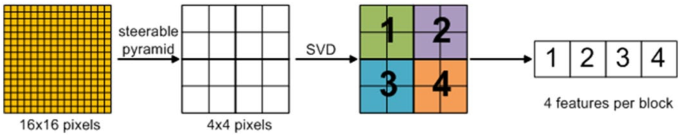


Fig. 5 The overall feature extraction process



should be high enough to avoid detecting pixels in the same region, but if it is too high forged pixels in different regions may be missed. A good threshold should have a low false positive rate and a low false negative rate. For this reason, several values were evaluated considering the false positive and false negative rates and the best value was found to be 50 pixels. A block diagram of the proposed algorithm is shown in Fig. 6.

5 Results and discussion

In this section, experimental results are presented to evaluate the effectiveness of the proposed method for image forgery detection. This method is compared with state of the art image forgery detection methods in terms of complexity and robustness to post-processed techniques.

5.1 Image dataset

The CoMoFoD image database [7] is used here to evaluate the image forgery methods. This database consists of two groups of images, one with 200 images of size 512 × 512 and the other with 60 images of size 3000 × 2000. The small scale images are classified into 5 categories: translation, rotation, scaling, distortion, and combination of all previous. Each category consists of 40 images and all images are processed using 6 post-processing techniques. These techniques are brightness change (BC), contrast adjustment (CA), color reduction (CR), and image blurring (IB). The proposed method has been evaluated on the 40 images of the translation category and 4 post-processing techniques BC, CA, CR, and IB. The proposed method has also been evaluated on one high resolution image of size 3000 × 2000. All images have 8 bit color vectors. Forged regions have been added to these images and modified using 4 different post-processing techniques [7]. For BC, the range of the color intensity values was changed by mapping the intensity to values between lower and upper bounds. Intensity values that fall outside these bounds are set to the corresponding minimum or maximum value. For CA, the image contrast is adjusted by mapping the range of color intensity values to a new interval bounded by lower and upper bounds. For BC and CA, the bounds are (0.01, 0.95), (0.01, 0.9), and (0.01, 0.8), denoted by 1, 2 and 3, respectively. For CR, the intensity values for each color are quantized to a smaller range. The ranges are (0, 32), (0, 64), and (0, 128) denoted by CR1, CR2 and CR3, respectively. For IB, the images are blurred by adding Gaussian noise to the intensity values. The mean of the noise is $\mu = 0$ and the variances are $\sigma = [0.009, 0.005, 0.0005]$ denoted by IB1, IB2 and IB3, respectively [7]. This is known as Gaussian blur. Note that IB1 is the highest level of blurring, whereas BC3, CA3, and CR3 are the highest levels of change for the other techniques. Figure 7 shows examples of forged images from CoMoFoD database. Column 1 is showing the original image, column 2 is showing the binary mask to show the duplicated regions, and columns 3 to 6 are showing examples of forged images post-processed with techniques BC, CA, CR, and IB respectively. The post-processing techniques showing in this table are at the highest level.

5.2 Performance of the proposed method

Figure 8 shows the performance of the proposed algorithm on 5 images from the CoMoFoD database. These images were selected because they show the performance for different sizes and numbers of forged regions. The first two columns show the original and forged images. Columns 3 to 6 show the results with the images after post-processing has been applied to the forged region, i.e. the detection results after BC1, CA1, CR1 and IB1 post-processing. The original region is circled in green and the forged region is circled in red. Thus in the first image, the bird was copied along with part of the adjacent area to make the color more consistent with the surrounding area after pasting. This makes the forged region hard to detect. In all cases, the proposed algorithm was able to detect the forged regions. The proposed method was able to locate and mask the forged region accurately even with a small region as in image 4 and multiple regions as in image 5.

To evaluate the robustness of the proposed algorithm against post-processing, all 3 levels of each technique are considered. Figure 9 shows the results for four images from the database. This shows that the forged regions were detected in all 12 cases. The accuracy of the proposed algorithm for the 200 images in the database is evaluated using precision and recall. Typically, precision and recall are calculated at the image level, i.e. based on how many images are correctly classified as forged, but to provide more accurate results, the precision and recall are calculated per pixel here. In other words, precision and recall are

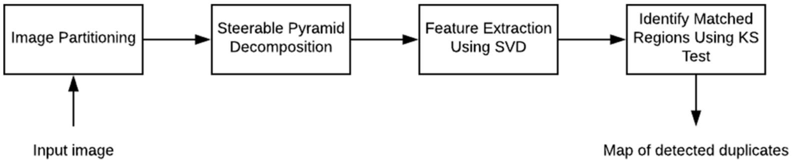


Image Partitioning
Steerable Pyramid Decomposition
Feature Extraction Using SVD
Identify Matched Regions Using KS Test
Input image
Map of detected duplicates

Fig. 6 Block diagram of the proposed algorithm

Fig. 7 Examples of forged images from CoMoFoD database.

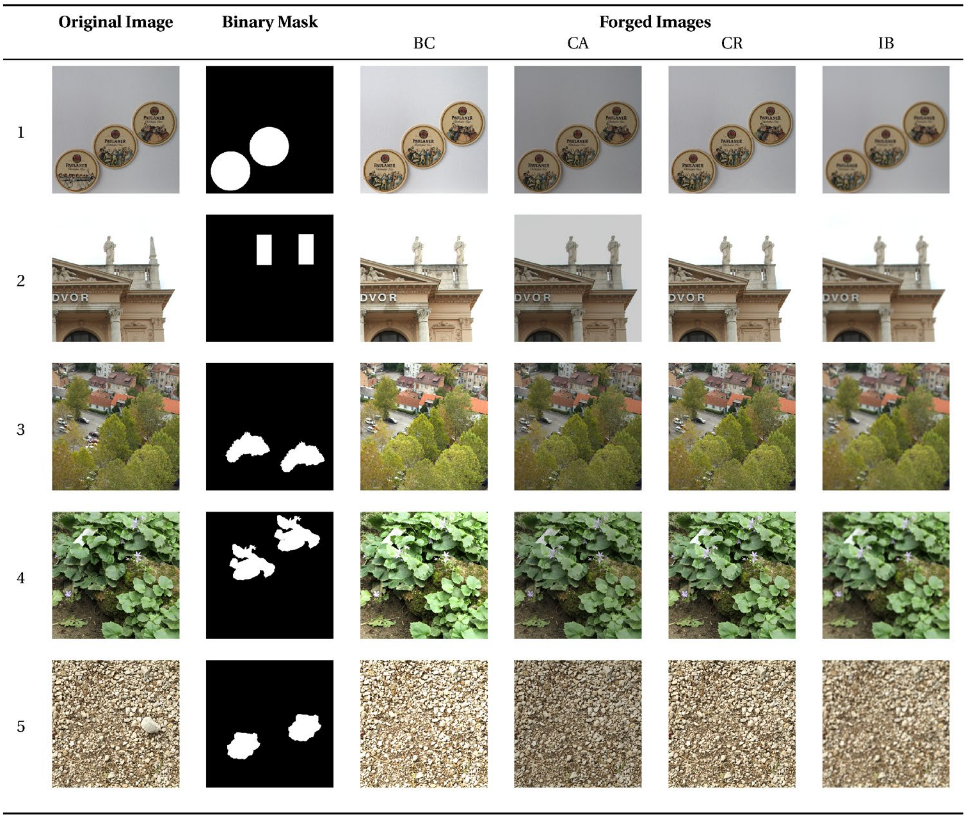


Fig. 8 The results obtained using the CoMoFoD database.

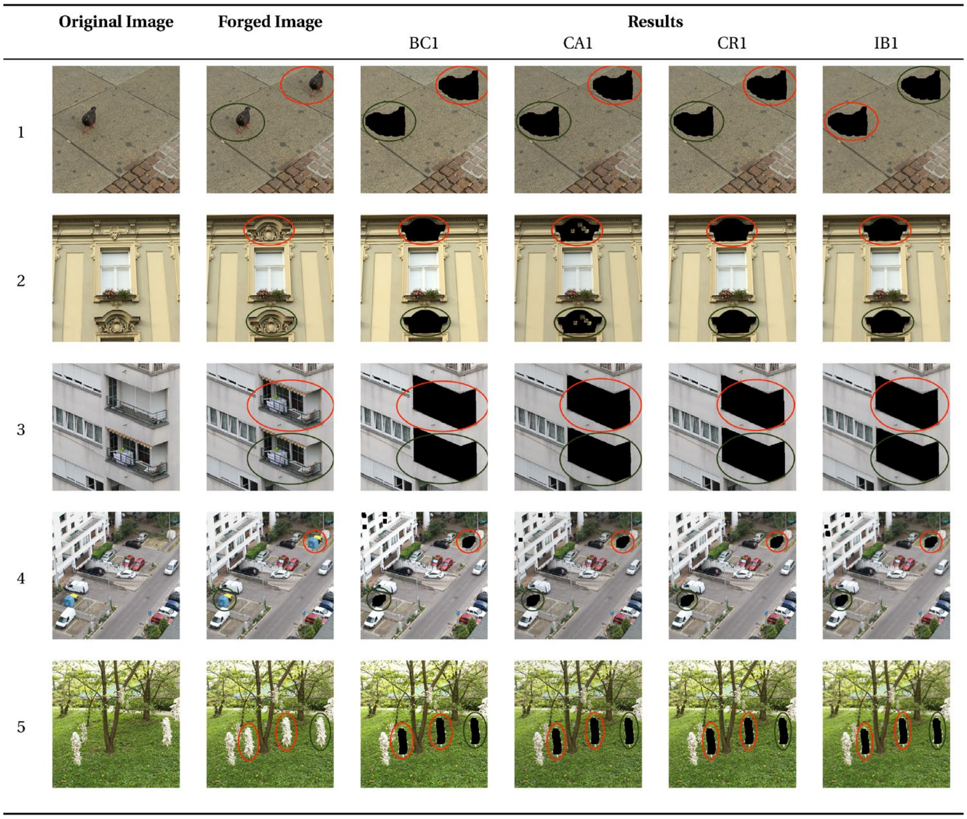
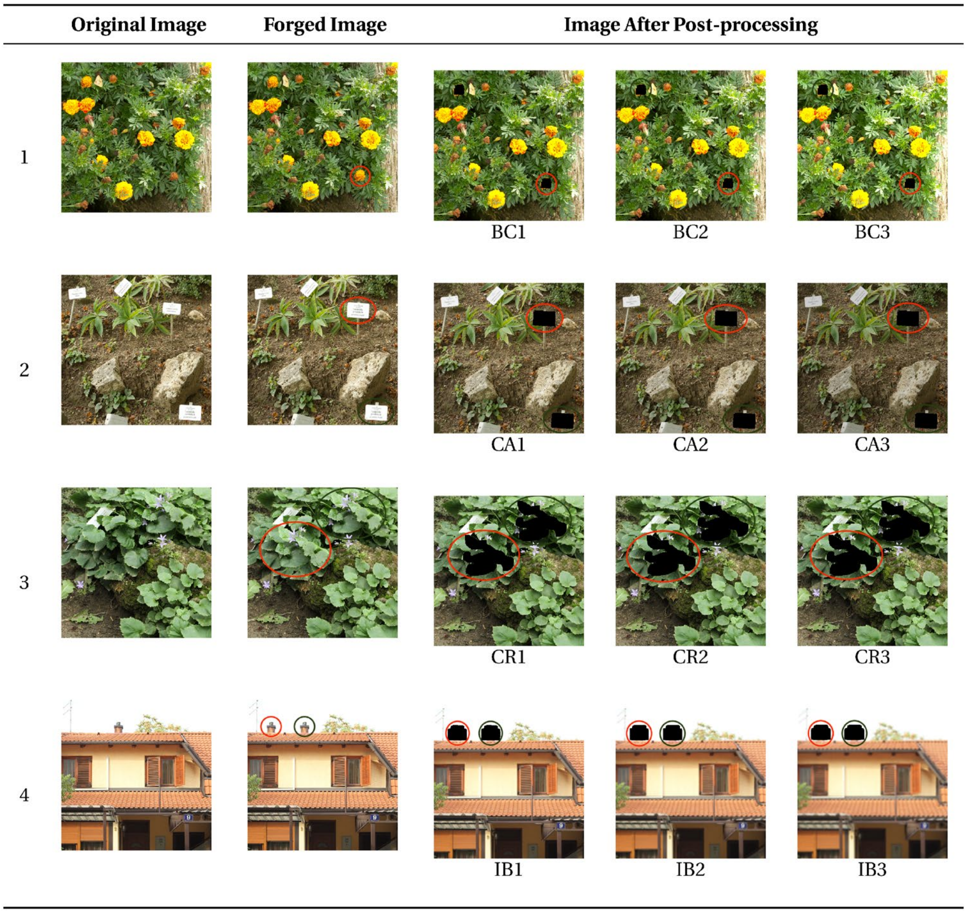


Fig. 9 Four images altered using different levels of post-processing.



calculated based on the percentage of pixels in the image that are correctly classified.

The precision, recall and F1_score are given by

$$\text{Precision} = \frac{\text{Forged region} \cap \text{Detected region}}{\text{Detected region}} = \frac{TP}{TP + FP}, \tag{3}$$

$$\text{Recall} = \frac{\text{Forged region} \cap \text{Detected region}}{\text{Forged region}} = \frac{TP}{TP + FN}, \tag{4}$$

$$\text{F1\_score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{5}$$

where *TP* is the true positive rate which is the number of forged pixels detected as forged, *FP* is the false positive rate which is the number of pixels incorrectly detected as forged, and *FN* is the false negative rate which is the number of forged pixels incorrectly detected as original. Precision is a good measure of the false positive rate whereas recall is a good measure of the false negative rate. F1_score is the weighted average of precision and recall which means it takes both false positives and false negatives into account. A good detection technique should have high precision, high recall, and high F1_score.

To calculate *TP*, *FP* and *FN*, the results are binarized using the Otsu method [39] for comparison with the binary masks from the database. In a binary mask, the intensity values of the forged pixels are set to 1 and the intensity values of the original pixels are set to 0. Figure 10 illustrates the precision and recall for the three levels of each of the four post-processing techniques. The results obtained are calculated by averaging over all 40 images chosen for evaluating this method. These results show that the precision and recall rates are almost constant over the levels with a maximum variation of ± 2% except for blurring which reduces the precision to 34% at the highest level. This consistency reflects the robustness of the proposed algorithm even with image post-processing. The precision achieved is more than 95% for three of the post-processing techniques, and except for the highest level of blurring it is greater than 75%.

Table 1 presents *FP* for the three levels of each post-processing technique. This shows that *FP* increases as the level of post-processing increases and reaches a maximum of just over 1% except for image blurring which is 4% for the highest level of blurring ($\mu = 0, \sigma = 0.0005$).

Table 2 presents the results for one image with the proposed and three other methods [5, 21, 22]. The binary

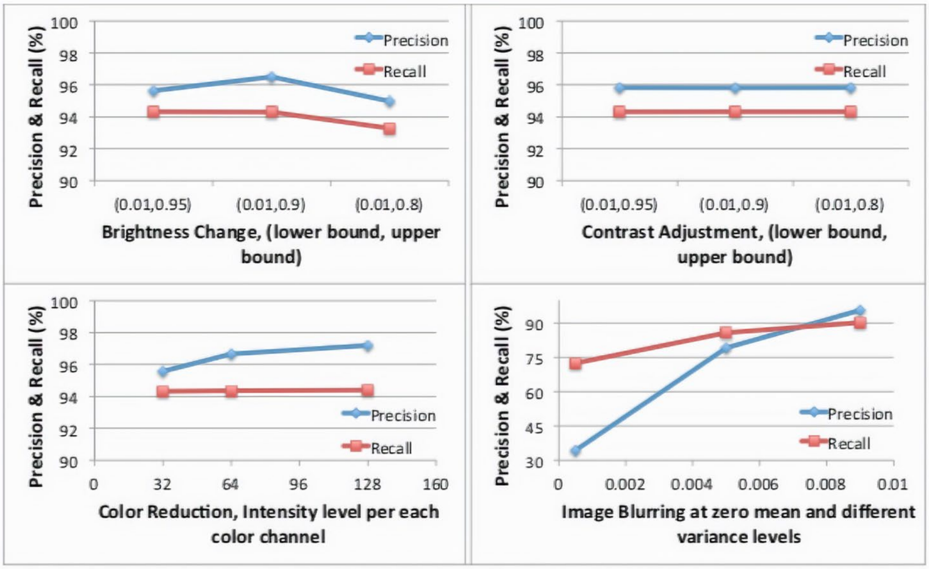
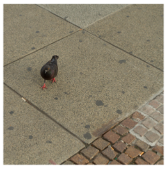
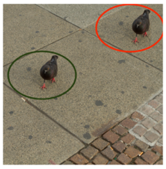
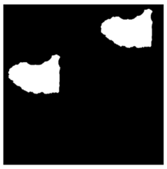
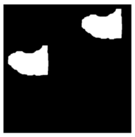
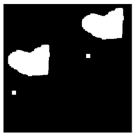
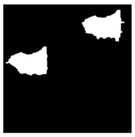
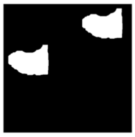


Fig. 10 Precision and recall with image post-processing **a** brightness change, **b** contrast adjustment, **c** color reduction, and **d** image blurring

Table 1 FP for different levels of post-processing

Post-processing technique	Parameters	FP (%)
BC (lower bound, upper bound)	(0.01, 0.95) (0.01, 0.9) (0.01, 0.8)	0.60 0.76 1.08
CA (lower bound, upper bound)	(0.01, 0.95) (0.01, 0.9) (0.01, 0.8)	0.57 0.57 0.57
CR (lower bound, upper bound)	(0, 32) (0, 64) (0, 128)	0.58 0.56 0.54
IB	$\mu = 0, \sigma_2 = 0.009$ $\mu = 0, \sigma_2 = 0.005$ $\mu = 0, \sigma_2 = 0.0005$	1.11 1.80 4.08

Table 2 Copy-move forgery detection results for the proposed and three published methods [5, 21, 22].

Method	Original image	Forged image	Binary mask	Results	Feature size
Ref. [23]					64
Ref. [21]					32
Ref. [5]					16
Proposed					4

mask is shown along with the result for the other three methods for comparison with the proposed method. The forged region and the region where it was copied from are shown in white and the reminder of the image is shown in black. The feature size per block for each method is given in the last column. Table 3 presents the results for one image of size 3000 × 2000 from the CoMoFoD high resolution database. The proposed method was able to detect the forged regions with an F1_score of 68.5%. The high precision and low recall mean that the proposed method has a low false positive as well as a high false negative ratio. Conversely, the method [21] was not able to detect the forgeries in this image due to the high complexity of the feature vectors used per block (i.e. 64 features) which exceeded the maximum array size of the software used.

The size of the feature vectors has a significant effect on the complexity. Thus, the main goal of the methods in [5, 21, 22] is to reduce the feature size while providing good detection performance. However, applying post-processing to an image may result in losing important features that can be used to find similar image blocks. For this reason, reducing feature size may result in the loss of more information which will affect the robustness against post-processing techniques. The last column in Table. 3 shows that the proposed method was able to accurately detect forged regions using only 4 features per block compared to the other methods which require 64, 32, and 16 features per block. The performance using the 200 small size images in the CoMoFoD database is given in Tables 4 and 5. In terms of precision, recall, and F1_score, the proposed method is the best while the technique in [21] is better in terms of the computation time. This is because the Big-O complexity of computing the DCT coefficients is $O(N)$ while the Big-O complexity of SVD is $O(N^2)$. However, the SVD features are better suited to image post-processing as the proposed method performed better than the other techniques for all four post-processing techniques using the lowest number of features.

6 Conclusion

In this paper, a new copy-move forgery detection algorithm was presented which is based on SVD and the KS test. The singular values are used as features. The feature vectors are sorted lexicographically to reduce the false matching rate. The KS test is then used to extract similar features. The performance of this method was tested using the CoMoFoD database. The results obtained show that the proposed method can detect forgeries at a pixel level and has a minimal false positive rate of less than 4%. Furthermore, this method is robust to the post-processing techniques such as brightness change, contrast adjustment, color reduction, and image blurring. Although this method had an insignificant longer computation time compared to one method we used, but the proposed method scored the highest precision, recall, and F1_score which averaged 95% for brightness change, contrast adjustment, and color reduction. For image blurring, it achieved the highest precision, recall, and F1_score which are 70%, 82.7%, and 75%, respectively. The proposed method uses only 4 features per block which is smaller than those used in the other methods. We also evaluated our method using a high

Table 3 The results obtained for a 3000 × 2000 image from the CoMoFoD high resolution database.

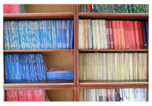
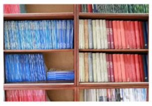
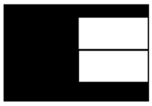
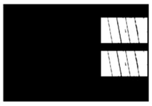
Original Image	Forged Image	Binary Mask	Results	Precision	Recall	F1_score
				99.8	52.2	68.5

Table 4 Comparison of the proposed and two published methods [5, 21]

Method	Brightness change			Contrast adjustment			Computation time (s)
	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)	
Ref. [5]	59.3	46.4	49.7	57.1	44.5	47.7	2495
Ref. [23]	57.5	76.9	58.4	53.7	78.8	55.0	51
Proposed	95.7	93.9	94.8	95.9	94.2	95.0	604

Table 5 Comparison of the proposed and two published methods [5, 21]

Method	Color reduction			Image blurring			Computation time (s)
	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)	
Ref. [5]	59.1	46.4	49.5	39.2	30.7	32.6	2495
Ref. [21]	57.0	78.8	58.5	53.0	70.6	51.0	51
Proposed	96.5	95.3	95.9	70.0	82.7	75.8	604

resolution image of size 3000 × 2000 and we obtained an F1_score of 68.5% which is promising.

Acknowledgements The authors would like to thank Radwa Hammad for her comments and advice that greatly improve the manuscript. They would also like to thank the reviewers for providing insightful suggestions and comments.

Compliance with ethical standards

Conflict of interest The authors declare that they have no competing interests.

References

1. Kang X, Wei S (2008) Identifying tampered regions using singular value decomposition in digital image forensics. In: Proceedings of IEEE international conference on computer science and software engineering, pp 926–930
2. Joglekar N, Chatur P (2015) A compressive survey on active and passive methods for image forgery detection. Int J Eng Comput Sci 4(1):10187–10190
3. Lin C, Chen C, Chang Y (2015) An efficiency enhanced cluster expanding block algorithm for copy-move forgery detection. In: Proceedings of IEEE international conference on intelligent networking and collaborative systems, pp 228–231
4. Ardizzone E, Bruno A, Mazzola G (2015) Copy-move forgery detection by matching triangles of keypoints. IEEE Trans Inf Forensics Secur 10(10):2084–2094
5. alZahir S, Hammad R (2017) Blind copula based copy-move forgery detection algorithm. In: Proceedings of IEEE international conference on consumer electronics, pp 436–437
6. Ahmed B, Gulliver TA, alZahir S (2020) Image splicing detection using mask-RCNN. In: Signal, image and video processing, pp 1–8
7. Tralic D, Zupancic I, Grgic S, Grgic M (2013) CoMoFoD—new database for copy-move forgery detection. In: Proceedings of IEEE international symposium ELMAR-2013, pp 49–54
8. Liu G, Wang J, Lian S, Wang Z (2011) A passive image authentication scheme for detecting region-duplication forgery with rotation. J Netw Comput Appl 34(5):1557–1565
9. Wang J, Liu G, Li H, Dai Y, Wang Z (2009) Detection of image region duplication forgery using model with circle block. In: Proceedings of IEEE international conference on multimedia information networking and security, pp 25–29
10. Yao H, Qiao T, Tang Z, Zhao Y, Mao H (2011) Detecting copy-move forgery using non-negative matrix factorization. In: Proceedings of IEEE international conference on multimedia information networking and security, pp 591–594
11. Chaitawittanun N, Munlin M (2015) An efficient clustering technique for copy-paste attack detection. Int J Comput Electr Autom Control Inf Eng 8(2):394–402
12. Singh VK, Tripathi R (2011) Fast and efficient region duplication detection in digital images using sub-blocking method. Int J Adv Sci Technol 35:93–102
13. Wang T, Tang J, Luo B (2013) Blind detection of region duplication forgery by merging blur and affine moment invariants. In: Proceedings of IEEE international conference on image and graphics, pp 258–264
14. Bravo-Solorio S, Nandi AK (2011) Automated detection and localisation of duplicated regions affected by reflection, rotation and scaling in image forensics. Sig Process 91(8):1759–1770
15. Khan ES, Kulkarni EA (2010) An efficient method for detection of copy-move forgery using discrete wavelet transform. Int J Comput SciEng 2(5):1801–1806
16. Nguyen HC, Katzenbeisser S (2012) Detection of copy-move forgery in digital images using radon transformation and phase correlation. In: Proceedings of IEEE international conference on intelligent information hiding and multimedia signal processing, pp 134–137
17. Li Y (2013) Image copy-move forgery detection based on polar cosine transform and approximate nearest neighbor searching. Forensic Sci Int 224(1–3):59–67
18. Ryu SJ, Kirchner M, Lee MJ, Lee HK (2013) Rotation invariant localization of duplicated image regions based on Zernike moments. IEEE Trans Inf Forensics Secur 8(8):1355–1370
19. Warbhe AD, Dharaskar R, Thakare V (2016) A survey on keypoint based copy-paste forgery detection techniques. Procedia Comput Sci 78:61–67
20. Ulutas G, Muzaffer G (2016) A new copy move forgery detection method resistant to object removal with uniform background forgery. In: Mathematical problems in engineering, vol 2016. Art. ID 3215162
21. Fridrich AJ, Soukal BD, Lukàš AJ (2003) Detection of copy-move forgery in digital images. In: Proceedings of digital forensic research workshop
22. Popescu A, Farid H (2004) Exposing digital forgeries by detecting duplicated image regions. Technical report TR2004-515. Department of Computer Science, Dartmouth College, Hanover, NH
23. Sharma K, Abrol P (2016) Non-overlapping block-based parametric forgery detection model. Int J Comput Appl 133(3):17–24
24. Cao Y, Gao T, Fan L, Yang Q (2012) A robust detection algorithm for copy-move forgery in digital images. Forensic Sci Int 214(1):33–43
25. Langille A, Gong M (2006) An efficient match-based duplication detection algorithm. In: Proceedings of IEEE Canadian conference on computer and robot vision, pp 64–64
26. Nathalie Diane WN, Xingming S, Moise FK (2014) A survey of partition-based techniques for copy-move forgery detection. Sci World J 2014. Art. ID 975456
27. Bravo-Solorio S, Nandi AK (2009) Passive forensic method for detecting duplicated regions affected by reflection, rotation, and scaling. In: Proceedings of IEEE European signal processing conference, pp 824–828
28. Cao Y, Gao T, Fan L, Yang Q (2012) A robust detection algorithm for region duplication in digital images. Forensic Sci Int 214(1–3):33–43
29. Myna A, Venkateshmurthy M, Patil C (2007) Detection of region duplication forgery in digital images using wavelets and log-polar mapping. In: Proceedings of IEEE international conference on computational intelligence and multimedia applications, pp 371–377
30. Al-Qershi OM, Khoo BE (2013) Passive detection of copy-move forgery in digital images: state-of-the-art. Forensic Sci Int 231(1–3):284–295
31. Li G, Wu Q, Tu D, Sun S (2007) A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD. In: Proceedings of IEEE international conference on multimedia and expo., pp 1750–1753
32. Zhang T, Wang R-d (2009) Copy-move forgery detection based on SVD in digital image. In: Proceedings of IEEE international congress on image and signal processing, pp 1–5
33. Chakravarti IM, Laha RG (1967) Handbook of methods of applied statistics. Wiley, New York
34. Justel A, Peña D, Zamar R (1997) A multivariate Kolmogorov–Smirnov test of goodness of fit. Stat Probab Lett 35(3):251–259

35. Dash SS, Jena UR (2017) Texture classification using steerable pyramid based Laws' masks. J Electr Syst Inf Technol 4(1):185–197
36. Unser M, Chenouard N, Van De Ville D (2011) Steerable pyramids and tight wavelet frames in L2(Rd). IEEE Trans Image Process 20(10):2705–2721
37. Freeman WT, Adelson EH (1991) The design and use of steerable filters. IEEE Trans Pattern Anal Mach Intell 13(9):891–906
38. Klema V, Laub A (1980) The singular value decomposition: its computation and some applications. IEEE Trans Autom Control 25(2):164–176
39. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):62–66

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.