International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 04
February 2018

Blur-Invariant Copy-Move Forgery Detection Technique with Improved Detection Accuracy utilising SWT-SVD

Karravala Krishna Reddy (M.Tech Scolar)[1]
Sk.Ayesha (Asst Professor)[2]
Electronics and communication Engineering Department [1,2]
AMARA institute of engineering and technology, Guntur, Andhra Pradesh 522549, India[1,2]
krishnareddy0402@gmail.com[1], ayesha4a0@gmail.com[2]

ABSTRACT

Majority of the existing copy-move forgery detection algorithms operate based on the principle of image block matching. However, such detection becomes complicated when an intelligent adversary blurs the edges of forged region(s). To solve this problem, the authors present a novel approach for detection of copy-move forgery using stationary wavelet transform (SWT) which, unlike most wavelet transforms (e.g. discrete wavelet transform), is shift invariant, and helps in finding the similarities, i.e. matches and dissimilarities, i.e. noise, between the blocks of an image, caused due to blurring. The blocks are represented by features extracted using singular value decomposition (SVD) of an image. Also, the concept of colour-based segmentation used in this work helps to achieve blur invariance. The authors' experimental results prove the efficiency of the proposed method in detection of copy-move forgery involving intelligent edge blurring. Also, their experimental results prove that the performance of the proposed method in terms of detection accuracy is considerably higher compared with the state-of-theart.

Keypoints: SWT, SVD, DWT, Colour based segmentation, Blur invarience

I. INTRODUCTION

In today's cyber world, the easy availability of highly advanced equipment and technology, and their wide accessibility to every common man, has put the credibility of digital data highly at stake. Today, neither a credit card number, nor a social security number, not even a bank account number can be used as an evidence, trustworthy enough to confirm one's identity.

Digital images, being the major information carriers in today's digital world, act as the primary sources of evidence towards any event in the court of law as well as media and broadcast industries. Nonetheless, the relative simplicity of editing and manipulating digital images have made their validity and reliability largely questionable. In fact, seeing is no more believing, due to the fact that in today's digital age, there is an expanding number of vindictively altered pictures.

Utilising an extensive variety of effective software applications, digital image manipulations by an adversary have become extremely common and simple. This can be a critical assignment when images are utilised as fundamental proof to impact judgment, for instance, in the court of law. Digital Forensics involves the use of scientific methods towards the investigation, analysis and interpretation of evidences derived from digital sources for the purpose of facilitating the reconstruction of events, hence helping to anticipate illegitimate adversarial activities.

Digital image forensics deals with analysis of image contents for investigation and detection of forgeries to an image. In this work, we address the problem of detecting copy-move forgery or region duplication attack [1–10], which is one of the most primitive as well as prevalent forms of digital image forgeries,

where the forger copies region(s) of an image and pastes it onto itself at some other location(s), with the malicious target to obscure or repeat significant image object(s). The presence of homogeneous texture in natural images, such as water, sky, grass, sand, foliage and so on make it all the more vulnerable to this form of attack.

An example of this attack is demonstrated in Fig. 1. The detection of copy-move forgery or region duplication in an image is made more difficult by an intelligent adversary, through blurring of edges of the forged region, so that traditional pixel block matching algorithms fail to detect the forgery. In this paper, we propose a copy-move forgery detection algorithm which is robust to edge blurring of the duplicate region(s).

Majority of the existing region duplication detection techniques are block based [4, 5], i.e. they aim to find pixel blocks that are exact continuous copies of each other in an image. Such methods are effective in detection of copy-move forgery, where an image region is duplicated without any form of alteration to it. However, for region duplication that involves region transformations such as scaling, rotation, edge blurring and so on, such block-based methods do not prove to be equally efficient. In this case, the

Fig. 1 Copy-move forgery:

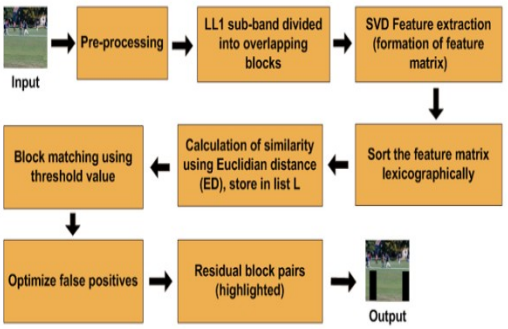

Fig. 2 Operational flowchart

keypoint-based algorithms are considerably helpful. Unlike the block-based algorithms, keypoint-based copy-move forgery detection methods [11, 12] rely on the identification and selection of high entropy image regions, called the keypoints. However, although these algorithms are robust against image transformations, they suffer from relatively high computational complexity. Our contribution in this paper is development of a forensic technique for blur-invariant copy-move forgery detection in digital images.

The proposed method decomposes an image into its frequency subbands using stationary wavelet transform (SWT) [13], and extracts features from the SWT subbands using singular value decomposition (SVD) [14]. The shift invariance, undecimated characteristics of SWT, and low computational complexity and stability of SVD, makes the proposed technique considerably efficient as compared with the state-of-the-art [1–10].

Also, in the proposed method, we introduce the concept of automatic threshold fitting to optimise manual effort. Colour-based segmentation [15] has been used in this work to achieve blur invariance. Further, to reduce the huge number of false positives produced when an image contains extensive regions of homogeneous texture, we have used a 8-connected neighbourhood checking [15]. The rest of the paper is organised as follows. A review of the state-of-the-art is presented in Section 2. Section 3 presents the proposed technique for detection of plain copy-move forgery, in detail. The proposed blur-invariant copy-move forgery detection technique has been presented

in Section 4. Our experimental results are presented in Section 5. Finally, we conclude in Section 6.

II .Related Work

In one of the pioneer researches in this area, Fridrich et al. [2] proposed region duplication detection based on the principles of exact block matching, autocorrelation, exhaustive block search and robust match [based on discrete cosine transform (DCT)]. The robust matching method proves to be most efficient, where the detection is based on matching of quantisation DCT coefficients, lexicographically sorted for computation efficiency. However, this method, when applied to images containing large identical textured regions, leads to a lot of false matches. Farid and Popescu [4] presented a computationally efficient copy-move forgery detection technique based on principal component analysis (PCA).

Here, the inherent dimensionality reduction characteristics of PCA have been used to reduce the number of features to half of that of [2]. However, due to dimensionality reduction, the efficiency reduces for lossy compressed or rotated images. Kang and Wei [5] proposed a region duplication detection method based on SVD, which is extremely effective in cases of duplicate regions induced with noise. Zhang et al. [6] proposed an algorithm based on discrete wavelet transform (DWT) for copy-move forgery detection, which again attains a considerably low computational complexity as compared with the other existing schemes.

A sorted neighbourhood approach based on DWT and SVD has been proposed by Li et al. [10], in which first DWT is applied to the image and then SVD is used to extract the features of the blocks of the low-frequency components, leading to dimension reduction of the blocks. Yang et al. [7] applied undecimated shift-invariant dyadic wavelet transform (DyWT) on a forged image by decomposing it into four frequency subbands, and have decomposed low-frequency subband into overlapping blocks. The Euclidean distance between each block pair is computed and the matching pairs are detected using a threshold.

This method optimises the number of false matches, even when image consists of extensive uniform regions. Another DyWT-based approach, capable of detecting lossy compressed duplicate image regions, where both low- as well as high-frequency components are utilised to get rid of false positives, is proposed by Muhammad et al. [9]. Bayram et al. [8] proposed a Fourier Mellin transform based method, using Bloom filters. This method proves to be scale and rotation invariant as well as computationally efficient, being able to detect forgery even in highly compressed images.

Mahdian and Saic proposed a blur moment invariant method for exposed images, degraded by blurring or noise. Most of the above methods suffer from false positives and are vulnerable to blurring. In this paper, we propose a blur-invariant region duplication detection method with reduced false positive rate (FPR) and improved detection accuracy (DA).

III .Proposed SWT-SVD based Copy-Move Forgery Detection

In this section, we discuss in detail the operSation of the proposed method for copy-move forgery detection, without blurring of the forged region. A block diagram representing the operational flow of the proposed technique is shown in Fig. 2. In the following subsections we discuss in detail the operations of the proposed method.

3.1 Pre-processing

The pre-processing step involves two operations. First, if the possibly forged input image is a colour one, it is converted from RGB to greyscale [15], using the following formula:

$$I = 0.299R + 0.587G + 0.114B \quad (1)$$

Second, SWT [13] is applied to the input grey scale image to obtain four subbands, viz. approximation (LL), horizontal (LH), vertical (HL) and diagonal or detail (HH), specifically denoted as LL1, LH1, HL1 and HH1, respectively, at scale 1. In the proposed

method, we use SWT because of the following inherent properties [20, 21] of it over traditional transforms as DWT. (i) The inherent translation invariant property of SWT makes it a perfectly suited technique for detection of copy-moved image portions. Due to this property, even if a signal is shifted, its SWT coefficients do not change. DWT does not preserve shift invariance. In copy-move forgery, the duplicate regions are not necessarily located in the same (relative) pixel positions of two blocks. If the descriptors vary with translation, they result into different representations, corresponding to these two blocks, and

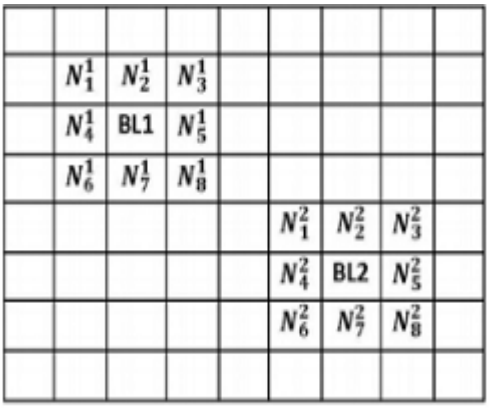

Fig.3 8-connected neighbours of blocks BL1 and BL2

Hence lead to incorrect inference by the region duplication detection algorithm. Since SWT is translation invariant, in such situations the descriptors are NOT altered, hence producing maximum accuracy in copy-move forgery detection. (ii) SWT performs efficient edge detection, due to which it enables efficient detection of blurring along the edges of duplicated region(s). (iii) SWT is applicable to discrete (image) signals of any arbitrary size. (On the contrary, DWT is suited only for images having dimensions in powers of 2.)

3.2 Feature Extraction using SVD:

The LL1 subband that gives the smoothed version of the image, say of size M × N pixels, is partitioned into overlapping blocks of size B × B pixels, which is assumed to be smaller than the size of the forged area to be identified, thus giving a total number of M − B + 1 × N − B + 1 blocks. However, to achieve a good DA, the block size should not be too small either. The more we increase the block size, the more features we obtain and the better accuracy we can expect. Experimental results pertaining to the variation of DA with unit forgery detection block size, has been presented in Section 5. SVD [14] is applied to each block to extract its corresponding singular values feature vectors. In the proposed method, SVD feature extraction proves to be effective, because of the following properties [2–4]:

(i) SVD is considerably less computationally intensive and stable technique which is invariant to translation.

(ii) The dimension of the image matrices need not be fixed for SVD. The matrix can be square as well as rectangular.

(iii) SVD efficiently represents the intrinsic algebraic properties of an image.

(iv) Singular vectors correspond to the geometrical characteristics (size, shape, position etc.) of objects within an image.

3.3 Sorting of Image Blocks

The feature vectors of the blocks that are stored row-wise in a matrix, called the feature matrix, resulting in a total number of

M − B + 1 × N − B + 1 rows, are lexicographically sorted each.

3.4 Calculation of Similarity

The Euclidean distance D(u, v) between a pair of rows (blocks), say u and v, where u = u1, u2, . . . , ur and v = v1, v2, . . . , vr is computed by D u,

$$D(u,v) = \left(\sum_{i=1}^{r}(u_i, v_i)^2\right)^{1/2} \quad (2)$$

provided blocks u and v are not more distant than a offset threshold, say T f, that signifies the maximum number of rows to compare, i.e. abs index u − index v ≤ T f. This helps to select only those similar blocks

which can be expected to have been copy-moved. We do not need to find the Euclidean distance of all the block pairs, but only the very similar blocks that are close to each other (as specified by T f), in the lexicographically sorted feature matrix, thus reducing the computation time. The computed Euclidean distance values, along with their corresponding block pairs, are stored in a list L.

3.5 Block Matching using Threshold

The list L now consists of all the block pairs that are to be further processed for detection of forgery. The length of L depends on the size of the image, the block size and also the offset threshold T f. L can be long if the image is big, the block size is small or T f is high. Hence, now L requires to be truncated appropriately so as to hold the relatively more similar block pairs than the rest. Thus, a similarity threshold Td, that helps to filter out less similar blocks from the list L, keeping block pairs having a greater probability to have been duplicated, is fitted. An automatic threshold fitting approach, followed in the proposed algorithm, has been discussed in Section 3.7. All the rows in L whose Euclidean distances are more than the similarity threshold Td are discarded, as they are considered to be non-duplicated parts of the images. Further verification is performed on the rest of the pairs that pass this stage of elimination, as follows. For a given block pair, say block1 with coordinates i, j and block 2 with coordinates k, l , the offset of coordinates between block 1 and block2 is given by C

$$C_{12} = \max[abs(i-k), abs(j-l)] \quad (3)$$

Block 1 and block2 are labelled as suspected duplicated regions if C 12 ≥ Ts where Ts is the minimum separation between duplicated regions. All such blocks that pass this filter are detected to be duplicated by the proposed method.

3.6 Optimising False Matches

During copy-move forgery detection, when an image is comprised of extensive regular textured regions, such as blue sky, green grass or scenery with a lot of greenery all around, a sandy desert or beach and so on, conventional copy-move forgery detection algorithms tend to produce huge FPR. This is due to the fact that in such cases, large parts of the image are naturally similar, and hence lead to incorrect detection results. To solve this problem, we adopt 8-connected neighbourhood checking [15]. Here, all the blocks that are detected to be duplicates are marked, and considered. For each marked block, its 8-connected neighbours, i.e. up, down, left, right and the four diagonals (as shown in Fig. 3), are checked. The number of neighbours that were also detected as duplicates by above subsection are counted. If this count is > x (some empirical value ≥ 1 and ≤ 7, say 4), then the original block is kept marked. Else, if the count is ≤ x, the original block is considered to be a false positive and hence unmarked.

All the residual marked blocks are output as duplicated or forged. Now the number of false positives has been optimised. After block matching step, it is observed that false matches arise due to similarities in region of uniform natural or textured image. According to our experimental results in Section 5, during copy-move forgery detection in standard test images, initially the FPR varied between 1.17 and 4.63%, which was reduced 0.43 to 2.16% by proposed 8-connected neighbourhood checking.

3.7 Threshold Fitting

Here, we present the detailed threshold fitting technique adopted in Algorithm 1. Manual threshold fitting: Threshold fitting is a crucial step in the proposed copy-move forgery detection algorithm. The threshold may be viewed as a barrier between the authentic and forged regions of the image. Hence, it requires utmost care while being chosen. A perfect fit may be obtained empirically, through selection of various random thresholds (in the valid range), then running the algorithm iteratively for all, and finally selecting the one that produces the most accurate forgery detection results. However, this naive process involves intense computational complexity and constant human interactions. Also, this threshold would vary from image to image. Hence, in this paper, we propose the concept of automatic threshold fitting as discussed next.

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 04
February 2018

The Institution of Engineering and Technology 2017 303 Automatic threshold fitting: In the proposed algorithm (according to Section 3.5), it is clear that the threshold value separates the list of block pairs into two parts. Empirical studies suggest that the part of L satisfying the threshold condition is only 0.1 − 0.3% of the entire list. This is our key observation, which has been used in this work to obtain the approximate threshold fit automatically. List L is sorted in ascending order to obtain list Lsort, and then the threshold value is chosen to be the Euclidean distance of the block pair that is located somewhere between 0.001 and 0.003th position of the sorted list. The threshold is computed as

$$T_d = ED_{sort_1} \times N \times 0.001; \quad (4)$$

where ED sort_1 is the Euclidean distance of the topmost block pair in sorted list L sort, and N is the total number of entities presented in L. Next, we present an example calculation of threshold Td values for a particular 256 × 256 test image, and the corresponding false match results, as shown in

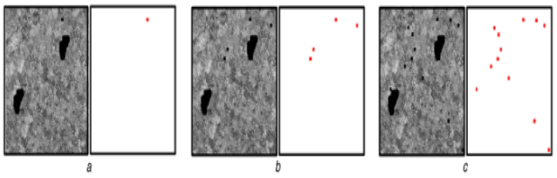

Fig. 4. The procedure for threshold calculation in the proposed algorithm is given in Algorithm 1 (Automatic_Threshold_Fitting).

Algorithm 1: Automated_Threshold_Fitting

1: **Input**: list (L) storing Euclidian distances of block pairs, N are the length of L;
2: **Output**: Threshold Td;
3: $L \leftarrow [ED_1, ED_2, \dots ED_N]^T$;
4: $L_{sort} \leftarrow [ED_{sort\_1}, ED_{sort\_2}, \dots, ED_{sort\_N}]^T \leftarrow Sort_{Assending}\ (L)$;
5: ***for*** $i = 1,2, \dots, N$ ***do***
6: $T_{d_i} \leftarrow [ED_{sort_i} \times N \times 0.001]$;
7: ***end for***
8:$T \leftarrow [T_{d_1}, T_{d_2}, T_{d_3}, \dots, T_{d_N}]^T$; /* *For demonstrating example* */
9: $T_d \leftarrow Td_1$; /* *Actual threshold adopted* */
10: *Return* T_{d_1};

Example: For the image shown in Fig. 4, the list L is obtained (according to the process described in Section 3.1–Section 3.5), and stored in an ascending order in list Lsort,which is

$$L_{sort} = \begin{bmatrix} 0.0899 \\ 0.1102 \\ 0.1187 \\ 0.1231 \\ 0.1307 \\ \vdots \end{bmatrix}$$

Depending on the Lsort entries, we populate the list T according to Algorithm 1 (Automated _Threshold_Fitting), as

$$T = \begin{bmatrix} 0.0899 \times 40520 \times .001 = 3.64 \\ 0.1102 \times 40520 \times .001 = 4.46 \\ 0.1187 \times 40520 \times .001 = 4.81 \\ 0.1231 \times 40520 \times .001 = 4.98 \\ 0.1307 \times 40520 \times .001 = 5.29 \\ \vdots \end{bmatrix}$$

The false matches corresponding to the first, third and fifth entries in T are shown in Fig. 4. It is evident from Fig. 4 that the least value of T d gives the lowest FPR. In our work, the first entry in list T is adopted as the threshold Td.

IV. Blur-Invariant Copy-Move Forgery Detection

Many times, an intelligent adversary intentionally blurs a region of an image while duplicating it, specifically its edges, so as to ensure that it does not stand out or seem out of place due to the abrupt variations along the edges. This makes the image imperceptible to human eyes, as well as helps to avoid detection of the forgery by conventional copy-move forgery detection algorithms such as [2, 4–6].

In this section, we discuss the proposed method for achieving blur invariance in copy-move forgery detection. Blurring any part of an image generates a noise in that area, different from the image's original noise, which is nothing but the undesired variation of colour information or brightness in the image, produced while capturing the image due to electronic

noise, and is different from the manual noise caused during blurring the forged part. The diagonal subband (HH) of an image obtained by applying SWT to it allows the detection of noise caused due to blurring. Key idea used: The proposed forgery detection method utilises two pieces of information for detecting copy-move forgery. They are i. Similarity between copied and moved regions in the smooth version of an image. ii. Noise inconsistency between the blurred region(s) and the remaining part of the image. In Fig. 5, we present a block diagram to depict the operational flow of the proposed blur-invariant copy-move forgery detection. In the following sub-sections, we discuss in detail the steps of the proposed method.

4.1 Pre-processing

If the possibly forged input image is a colour one, it is first converted to greyscale [9], using (1). Next, SWT [13] is applied to the input greyscale image to obtain four subbands, viz. approximation (LL1), horizontal (LH1), vertical (HL1) and diagonal or detail (HH1), at scale 1.

4.2 Colour-Based Segmentation for Blur Invariance

Colour-based segmentation is performed on the image in the L ∗ a ∗ b colour space, using K-means clustering. Segmentation is performed only when blurring is involved in the forgery. This is because, when a forged image involves blurring, the naturally homogeneous textures present in the image exhibit more similarity as compared with the copy-blurred-moved regions. However, this is not true for non-blurred forged images, since here, the copymoved parts are exact copies of each other. It was observed many Fig. 4 False positive results using first, third and fifth thresholds in list T 304 times that the fitted threshold is not able to take care of the false positives caused after blurring; hence this adversely affects the DA of the scheme.

To solve this problem, colour-based segmentation is applied. The image is segmented into, say n segments according to its colours, and the individual segments are handled independently. The individual segments are divided into overlapping blocks of fixed size and then threshold fitting is carried out on each of them.

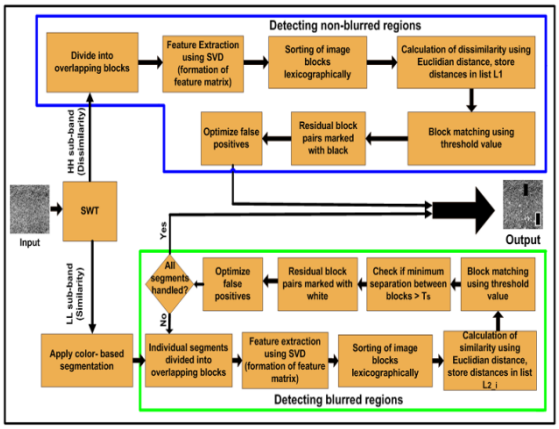

Fig. 5 Operational flowchart

Fig. 6 K-means clustering applied to Lena test image

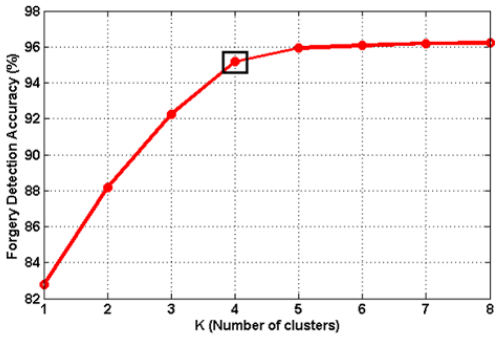

Fig. 7 Elbow method for deciding K

. However, now it is done for each individual segment, and not the entire image as a whole. To achieve colour-based segmentation using K-means clustering [5], we first convert an image to its L∗a∗b colour space. The L∗a∗b colour space is selected since it models the human visual system extremely efficiently. Also, this colour space being a

perceptually uniform orthogonal Cartesian coordinate system is appropriate for this problem. The L∗a∗b∗ space consists of a luminosity layer 'L∗', chromaticity layers 'a∗' and 'b∗', which indicate where a colour lies along the red–green and blue–yellow axes, respectively.

K-means clustering involves two parameters, viz., the number of clusters to be formed, and a distance metric to quantify the degree of closeness of two objects. Here, objects are nothing but the pixels represented by 'a∗' and 'b∗' values. The result of K-means clustering is used to label the pixels. Every cluster gets an index as returned by the K-means clustering algorithm, and every pixel in the image is labelled with its cluster index distinguished by different greyscale values. The results of K-means clustering applied to the Lena image, has been shownnFig. 6. Decision of K:

The value of K is decided by using Elbow method [6]. As we increase K, computational complexity increases. So we select the value of K (number of clusters) in such a way that is further adding cluster causes negligible improvement in forgery DA (for mathematical formulation of DA and other performance parameters, the reader may please refer to Section 5.1). Table 1 presents the correlation between interval in number of clusters and corresponding improvement of DA. In Fig. 7, we have shown the improvement in DA with increase in K. We observed that after the interval $K \in 4-5$, we obtain negligible improvement of DA by increasing K. In this work, we selected K = 4 following the Elbow method. The results for different values of K, i.e. K = 2, 3, …, 6, in terms of blur invariance, are shown in Fig. 8, for one test image.

4.3 Feature Extraction using SVD

The HH1 subband that captures the details of the image is partitioned into overlapping blocks. In practice, the region

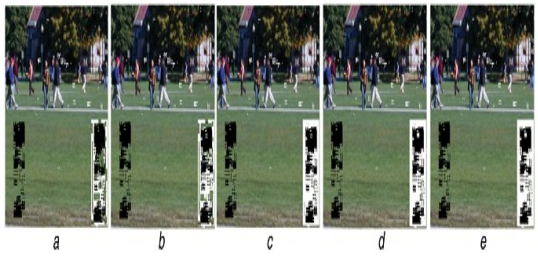

Fig. 8 Blur invariance achieved with K = 2, 3, …, 6 for a single test image

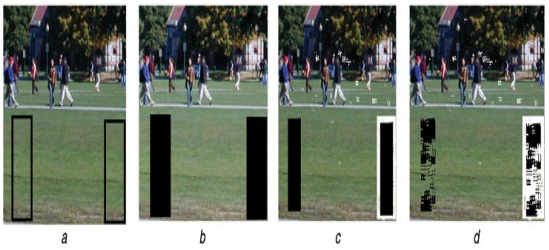

Fig. 9 Copy-move forgery detection results of the proposed method, with and without blurring (a) Manually forged input image, (b) Output obtained when there is no blurring involved, (c) Output obtained when there is blurring along the edges, (d) Output obtained when there is blurring along the edges as well as inside the forged area

undergoing blur is considerably smaller compared with the entire image. Hence, the chosen unit block size should be considerably smaller compared with the image size. However using extremely small unit blocks (say 2 × 2) hinders the process by increasing the computation time. So a correct tradeoff has to be arrived at. Next, the LL1 subband of the image is divided into overlapping blocks, not for the entire image, but segment-wise. Note here that, segmentation is required for the LL subband only, not the HH subband. This is because, segmentation is required here to avoid false positives while finding similarity (using LL); specifically, false positives caused due to blurring visa-vis natural similarities between image regions. However, the HH subband helps in noise detection, and hence used for recording dissimilarities among image regions. So, segmentation is not required here. Features of each block in HH1 and LL1 (for each segment) are extracted using SVD.

4.4 Sorting of Image Blocks

The feature vectors of the blocks are stored row-wise in a matrix, called the feature matrix, and are sorted lexicographically. This is done for blocks of the entire image (HH1) and of each segment too (LL1).

4.5 Calculation of Similarity and Dissimilarity using Euclidean Distance

The Euclidean distances between the pixel values of each pair of blocks in HH1 and LL1 are computed, provided the blocks are as far from each other as possible, the offset threshold say T f h and T f l signifying the maximum number of rows to compare. This is done to get the most dissimilar and most similar blocks that can be expected to have been blurred and possibly copy-moved, respectively. The computed Euclidean distances, along with their corresponding block pairs, are stored into lists L1 and L2_i, where i = 1, 2, …, n and n is the total number of segments.

Table 1 Improvement of DA with *K*

Interval of *K* values	DA improvement, %
1–2	5.41
2–3	4.08
3–4	2.91
4–5	0.77
5–6	0.14
6–7	0.10
7–8	0.04

4.6 Block Matching using Threshold

A dissimilarity threshold Tdh that helps to filter out less dissimilar block pairs from list L1, keeping block pairs having higher probability of being blurred, is fitted using automatic threshold fitting. Similarly, similarity threshold Tdl_i that helps to filter out less similar blocks of segment i from the list L2_i, keeping block pairs having higher probability of having been duplicated, is fitted. All rows in L1 whose Euclidean distance is less than the dissimilarity threshold Tdh are discarded. Similarly, all rows in L2_i whose Euclidean distances are more than the similarity threshold T dl_i are discarded. Further verification is performed to the rest of the block pairs that pass this stage of elimination. Two blocks, say block1 and block2 are output as duplicates, if the separation between them is ≥ Ts, where Ts is the minimum allowed separation between duplicated regions. This is carried out only for LL1 blocks, not HH1. If two blocks are too close to each other, specifically overlapping, then their feature vectors would also be very similar. Hence, the value of Ts depends on the block size. All such LL1 blocks, for each segment, that pass the above filter are marked in black in the output image. The residual block pairs in L2 are marked white in the output image.

4.7 Optimising False Matches

For both the blackened and whitened blocks, the 8-connected neighbourhood check is carried out to remove the false positives as discussed in Section 3.6. Finally, the resultant image is displayed with duplicate blocks shown in black, and white blocks indicating the blurred parts. Note here that, due to the colour-based segmentation adopted in the proposed method, all the blocks are not compared any more; only intra-segment comparisons are performed. The performance of the proposed method in terms of copymove forgery detection, with and without blurring involved, has been presented in Fig. 9. This figure shows a natural image with extensive green (grass) patch, from which a portion has been copymoved manually. In Fig. 9a, we have shown the forged image. The copy-move forgery detection results using the proposed SWT with SVD based method, for the manually forged input image, have been shown in Figs. 9b–d. Fig. 9b is the output obtained when there is no blurring at all, Fig. 9c is the detection result obtained when the edges of the duplicated region have been blurred and Fig. 9d is the result obtained when there is blurring along the edges as well as in the inner parts of the forged region.

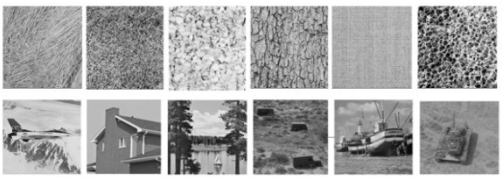

Fig. 10 256 × 256 standard greyscale (texture and natural) test images

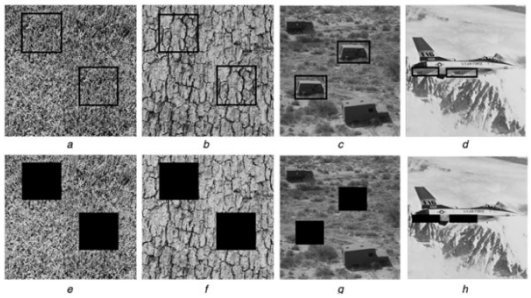

Fig. 11 Manually forged test images with no blurring and corresponding forgery detection results using the proposed SWT–SVD method (a)–(d) Manually induced forgeries, (e)–(h) Duplicate regions detected

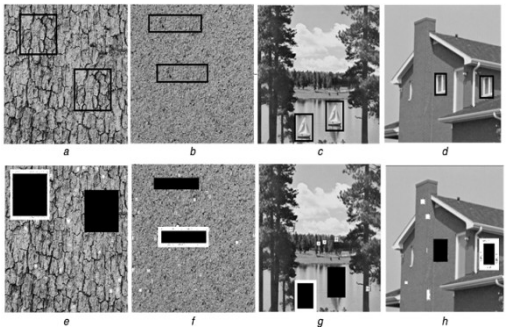

FIG : 12 Manually forged test images with blurring and corresponding forgery detection results using the proposed SWT-SVD method (a)–(d) Manually induced forgeries, (e)–(h) Duplicate regions detected

V. Experimental Results and Discussion

The proposed methods have been implemented in MATLAB using the MATLAB Image Processing Toolbox. Our experiments have been carried out on 12 standard 256 × 256 test images, including texture images and natural images, shown in Fig. 10. In order to avoid experimental bias, all results presented in this paper are taken as the average over multiple test images of Fig. 10. The performance results of the proposed method, for randomly selected texture and natural test images (from Fig. 10) have been shown in Figs. 11 and 12, for non-blurred and blurred forgeries, respectively.

5.1 Performance Evaluation

For quantitative evaluation of the performance efficiency of the proposed method, we have used DA, defined as

$$DA = \frac{Number\ of\ correctly\ detected\ copy - movied\ pixels}{Number\ of\ pixels\ actually\ copy - moved} \times 100\% \quad (5)$$

In order to evaluate the performance of the proposed method without blurring, we have varied the unit detection block size, well as the forgery size, and measured the corresponding DA results. These results have been presented in Table 2, and are the average results for all our test images shown in Fig. 10. The DA results of the proposed method when blurring is involved in the forgery have been shown in Table 3 for varying block and forgery sizes. Here, the block sizes for both HH and LL subbands are equal. The results presented in Table 3 represent the average DA over all our test images. Here the evaluation has been done by combining the results obtained by considering both LL (similarities) and HH (noise) subbands.

5.2 Comparison with State-of-the-Art

In this section, we compare the proposed method with state-of-theart in terms of DA. We have varied the forgery size from 10 to 40% in steps of 10%. The results (averaged over our entire test set) have been presented in Table 4, which prove that the proposed technique is considerably more efficient as compared with the state-of-theart. We also present the results in form of two-dimensional plots in Fig. 13.

5.3 Improved False Positives

In this section, we present our experimental results pertaining to usefulness of the 8-connected neighbourhood check with respect to reduction of the FPR. These results have been presented in Tables 5 and 6, for forged images without blurring and with blurring, respectively. FPR is defined as the total number of authentic image pixels, falsely detected to be forged, as

$$FPR = \frac{Number\ of\ pixels\ falsely\ dtetcted\ to\ be\ copy - moved}{Number\ of\ pixels\ actually\ copy - moved} \times 100 \quad (6)$$

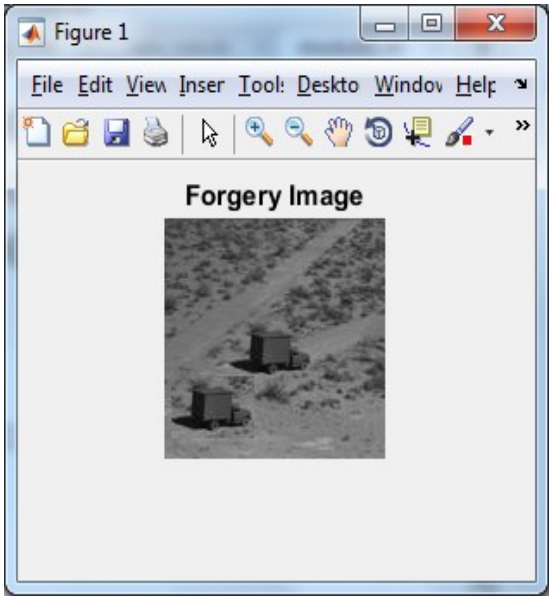

Fig13.Forgery Image

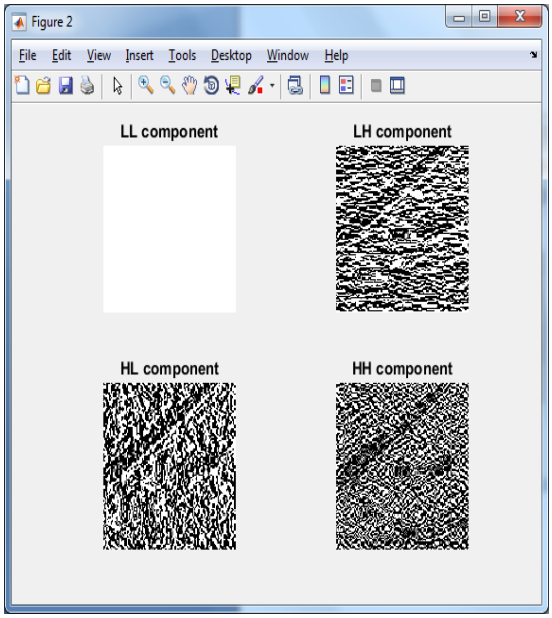

Fig14.Applying DWT (a) LL Component (b) LH Component (c) HL Component (d) HH Component

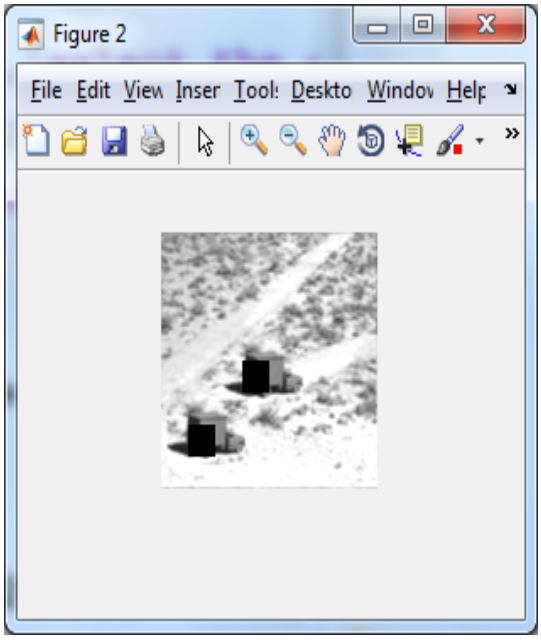

Fig15.Forgery part detected by black box

VI. CONCLUSION

In this paper, we have proposed a block-based copy-move forgery detection method for digital images, based on SWT with SVD that is robust to blurring. We introduced the concept of automatic threshold fitting to minimise manual effort and computation time. We carried out colour-based segmentation to achieve blur invariance and 8-connected neighbourhood checking to optimise the number of false positives. The proposed method is evaluated for two types of forgeries: copy-move forgery (i) without blurring, (ii) with blurring. Our experimental results prove that the proposed method provides higher forgery DA as compared with the state-ofthe-art. Future research in this direction would include investigation of other forms of image region transformations, such as rotation, rescale and reflection, in copy-move forgery.

REFERENCES

[1] Christlein, V., Riess, C., Jordan, J., et al.: 'An evaluation of popular copymove forgery detection approaches', IEEE Trans. Inf. Forensics Sec., 2012, 6, pp. 1841–1854

[2] Fridrich, A.J., Soukal, B.D., Luk, A.J.: 'Detection of copy-move forgery in digital images'. Proc. of Digital Forensic Research Workshop, 2003

[3] Huang, Y., Lu, W., Sun, W., et al.: 'Improved DCT-based detection of copymove forgery in images', Forensic Sci. Int., 2011, 206, (1), pp. 178–184

[4] Farid, A.P., Popescu, A.C.: 'Exposing digital forgeries by detecting duplicated image region'. Technical Report, Hanover, Department of Computer Science, Dartmouth College, USA, 2004 [5] Kang, X., Wei, S.: 'Identifying tampered regions using singular value decomposition in digital image forensics'. Int. Conf. on Computer Science and Software Engineering, 2009, vol. 3, pp. 926–930

[6] Zhang, J., Feng, Z., Su, Y.: 'A new approach for detecting copy-move forgery in digital images'. 11th IEEE Singapore Int. Conf. on Communication Systems, 2008, pp. 362–366

[7] Yang, J., Ran, P., Tan, J.: 'Digital image forgery forensics by using undecimated dyadic wavelet transform and zernike moments', J. Comput. Inf. Syst., 2013, 9, (16), pp. 6399–6408

[8] Bayram, S., Sencar, H.T., Memon, T.N.: 'An efficient and robust method for detecting copy-move forgery'. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, 2009, pp. 1053–1056

[9] Muhammad, G., Hussain, M., Bebisi, G.: 'Passive copy-move image forgery detection using undecimated dyadic wavelet transform', Digit. Invest., 2012, 9, (1), pp. 49–57

[10] Li, G., Wu, Q., Tu, D., et al.: 'A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD'. IEEE Int. Conf. on Multimedia and Expo, 2007, pp. 1750–1753

[11] Huang, H., Guo, W., Zhang, Y.: 'Detection of copy-move forgery in digital images using SIFT algorithm', Comput. Intell. Ind. Appl., 2008, 2, pp. 272– 276

[12] Bo, X., Junwen, W., Guangjie, L., et al.: 'Image copy-move forgery detection based on SURF'. Int. Conf. on Multimedia Information Networking and Security, 2010, pp. 889–892

[13] Ilea, D., Whelan, P.: 'The stationary wavelet transform and some statistical applications'. Lecture Notes in Statistics, New York, 1995

[14] Klema, V.C., Laub, A.J.: 'The singular value decomposition: Its computation and some applications', IEEE Trans. Autom. Control, 1980, 25, (2), pp. 164– 176

[15] Ilea, D., Whelan, P.: 'Color image segmentation using K-means clustering algorithm'. 10th Int. Machine Vision and Image Processing, 2006