A Forensic Technique to Detect Copy-Move Forgery Based on Image Statistics

Ayush Nirwal(✉), Raghav Khandelwal(✉), Smit Patel(✉),
and Priyanka Singh(✉)

Dhirubhai Ambani Institute of Information and Communication Technology,
Gandhinagar, India
{201701091,201701015,201701071,priyanka_singh}@daiict.ac.in

Abstract. The proliferation of easy multimedia editing tools has ruined the trust in what we see. Forensic techniques are proposed to detect forgeries unnoticeable by naked human eyes. In this paper, we focus on a specific copy-move forgery attack that happens to alter portions within an image. It may be aimed to hide any sensitive information contained in a particular image portion or misguide the facts. Here, we propose to exploit the image's statistical properties, specifically, mean and variance, to detect the forged portions. A block-wise comparison is made based on these properties to localize the forged region called a prediction mask. Post-processing methods have been proposed to reduce false positives and improve the accuracy(F-score) of the prediction mask. This decrease in FPR in the final result is comes from post processing method of overlaying multiple masks with different values of threshold and block_size of the sliding window.

Keywords: Copy move forgery · Statistical properties · Mean

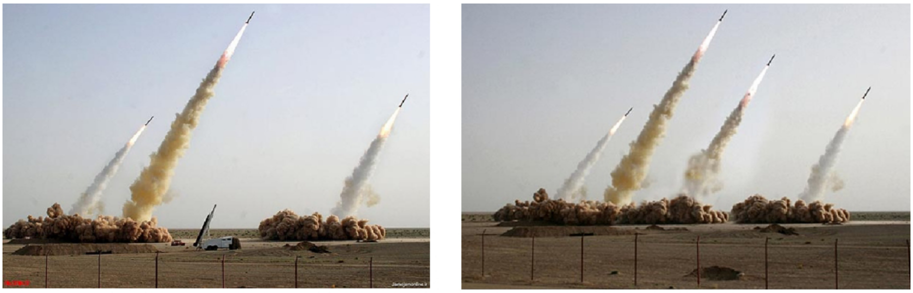
1 Introduction

The readily accessible, easy-to-use, and potent digital image editing tools such as Photoshop have made it easy to manipulate and tamper with digital images without leaving any visible clues. As a result, there is a massive rise in digitally produced forgeries in mass media and on the Internet. This pattern suggests vulnerabilities issues and reduces the integrity of digital images. Developing techniques for checking the validity and authenticity of digital images has become very necessary, mainly since the images displayed are evidence in a courtroom, as news reports, as a financial document. In this context, image tamper identification has become one of the critical objectives of image forensics.

We focus here on a particular form of image manipulation where a part of the image is usually copied and pasted on to another section, typically to cover unwanted parts of the image, named as copy-move forgery. An example of copy-move forgery is shown in Fig. 1, where image (a) is the original image and shows

L. Batina et al. (Eds.): SPACE 2020, LNCS 12586, pp. 86–98, 2020.
https://doi.org/10.1007/978-3-030-66626-2_5

three missiles, whereas image (b) is the forged image in which one missile is copy pasted at a different location on the image to show that there were four missiles launched instead of 3. From this example, it becomes clear that it is quite possible that forgeries may not leave any perceptual clues of tampering. Thus, it becomes quite challenging to identify such cases and ensure that the integrity of the image is intact. They may be crucial to applications at times.



(a) Original Image

(b) Forged Image

Fig. 1. An example of copy-move forgery

To detect copy-move forgeries, many schemes have already been proposed in the literature. Some schemes propose solutions that are too computation intensive while others lack at accurate region localization for the forged portions and result in high false positive rate (FPR). FPR values for various copy move forgery detection (CMFD) schemes has been enlisted in Table 1.

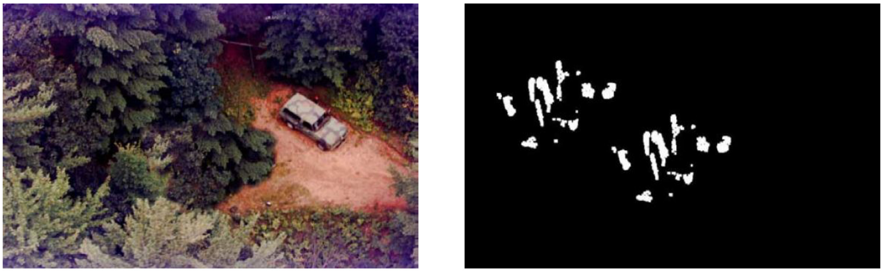
Table 1. Approximate FPR of various CMFD techniques

Algorithms	False positive rate
PCA [15]	9.04
DCT [2]	11.33
IDCT [14]	9.81
DyWT [9]	12.91
DWT [17]	10.11
DyWT_zernike [16]	8.08
SVD [11]	7.87
Dixit et al. [3]	2.55
Proposed SCMFD	0.051

In this paper, we propose a copy-move forgery detection algorithm for images. The baseline idea is to utilize the statistical image properties, specifically, mean and variance to detect the duplicate regions. The image is partitioned into blocks and comparison based on the block properties is done to categorize it as tampered or authentic region. We achieve a detection accuracy (F-score) of 97.05% with FPR as low as 0.051%. Rest of the paper is organized as follows: The related work is discussed in Sect. 2. Detailed steps of the proposed Statistical Copy Move Forgery Detection (SCMFD) algorithm are presented in Sect. 3. In Sect. 4, we present the results and the analysis and Sect. 5 concludes the work along with the future directions.

2 Related Work

The area of copy-move forgery is well researched and many methods have already been proposed to detect copy-move forgery. One of the most straightforward and obvious technique is comparing each pixel of an image with other pixels to detect manipulation [2]. Though the idea seems pretty simple, but it has a lot of computational complexity. The computation would be of order $O(P^2)$, where P is the total number of pixels in the image. The number of computations can be reduced by lexicographically sorting the pixels according to their values and only comparing the values in the near vicinity to find copy-move pixels [1]. However, this method has its shortfalls even after optimizing the computations. This method can be tricked by slightly changing the values of the pixel or rotating it during copy-move forgery. And perpetrators often change this value by color corrections and smoothing. This often results in disconnected pixels being detected as shown in Fig. 2. This method is also not robust against JPEG compression [2].



(a) Copy-move forgery on a image

(b) Disconnected pixels in simple block based detection

Fig. 2. Shortfalls of a simple block based copy-move forgery detection technique [2]

A simple block based approach to detect copy-move forgery would be to compare the mean and standard deviation of blocks [3]. However, this approach alone is not resilient to images where background looks similar or have similar pixels properties. This background creates a high number of false positives, which increases the false positive rate (FPR) and decreases the accuracy. Another standard method for copy-move forgery detection is auto-correlation. Most of the 'information' in an image is stored in the low-frequency range, so we cannot directly apply auto-correlation on the image; otherwise, we will have spikes on the edges [2]. We first pass it through a high pass filter, which will remove all the high frequency from the image. Then we compute the auto-correlation of the image to detect copy-move forgery. This method is not computationally intensive, but it has a hard time detecting copy-move forgeries, which are relatively smaller to the size of the image [2].

A popular method for copy-move detection is using Discrete Cosine Transform (DCT) [1]. The image is divided into a number of consecutive blocks usually 8 × 8, then DCT is applied on that block, and low-frequency data is extracted using zig-zag traversal. Then this block is sorted lexicographically to find similar blocks within a user-defined threshold [1]. Another similar method for copy-move forgery detection is using Principal component analysis (PCA) instead of DCT [13]. PCA is used to represent higher dimension data into lower dimensions, and in this case, PCA will extract data from the blocks and then compare that said data to find similar blocks that are copy-moved. DCT is a better approach in comparison to PCA to find copy-move forgery [1].

Similarly, many more approaches have been proposed in the field of copy-move forgery detection. In [4], the authors proposed a sorted neighborhood technique based on a discrete wavelet transform (DWT). Then Singular Value Decomposition (SVD) is applied on the image's low frequency information, this method is robust against JPEG compression. In [5], the authors proposed an approach based on Fourier-Mellin Transform (FMT) along with bloom filters for CMFD. This approach is also resilient to post processing techniques like Gaussian Noise and blur. In [6], the authors proposed an approach based on DCT and singular value decomposition (SVD). Although the approach is not robust against rotation but it gives good results in case of noise, blurring, and compression.

In [7], the authors proposed an approach based circular block extraction and Local Binary Patterns (LBP). This approach is robust against compression, rotation, blurring and noise. In [8], the authors used an approach based on circular blocks and Polar Harmonic Transform (PHT). In [9], the authors proposed a technique based on Dyadic Wavelet Transform (DyWT). In [10], an approach based on Histogram of Oriented Gradients (HOG) is used to detect the copy-move forge regions. High false positive rate (FPR) was the bottleneck for most of the approaches which the proposed approach is successful at drastically decreasing as tabulated in Table 1.

3 The Proposed Statistical Copy Move Forgery Detection (SCMFD) Approach

In this section, we present the proposed Statistical Copy Move Forgery Detection (SCMFD) approach. It aims to accurately localize the forged portions within an image exploiting it's statistical properties. An overview of the proposed SCMFD approach is shown in Fig. 3. Some optimizations are proposed to improve the overall accuracy of the final prediction mask.

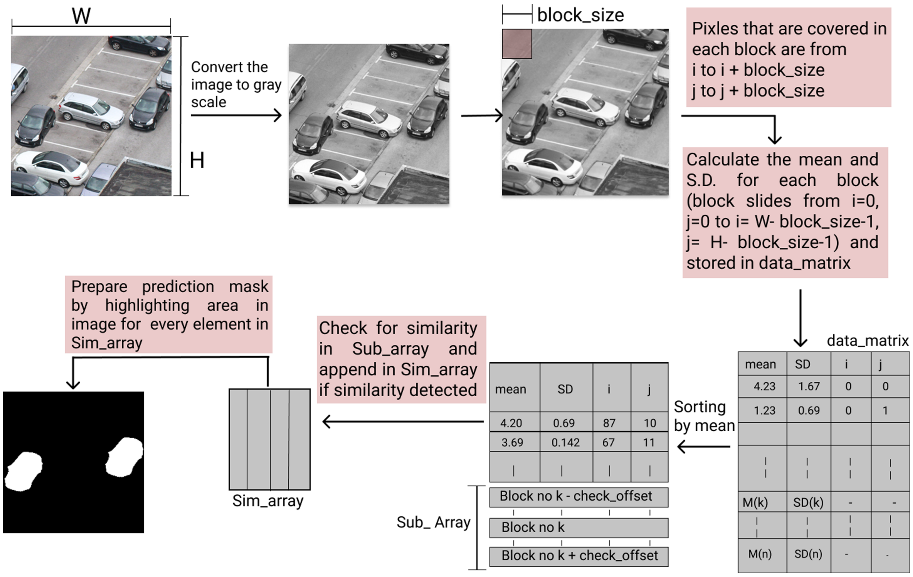

W
H
Convert the image to gray scale
block_size
Pixles that are covered in each block are from i to i + block_size j to j + block_size
Calculate the mean and S.D. for each block (block slides from i=0, j=0 to i= W- block_size-1, j= H- block_size-1) and stored in data_matrix
data_matrix
mean SD i j
4.23 1.67 0 0
1.23 0.69 0 1
M(k) SD(k) - -
M(n) SD(n) - -
Sorting by mean
mean SD i j
4.20 0.69 87 10
3.69 0.142 67 11
Block no k - check_offset
Block no k
Block no k + check_offset
Sub_ Array
Check for similarity in Sub_array and append in Sim_array if similarity detected
Sim_array
Prepare prediction mask by highlighting area in image for every element in Sim_array

Fig. 3. An overview of proposed SCMFD approach

3.1 SCMFD

Here, we present a region duplication detection method for images. It utilizes the statistical image features viz., mean, and variance to detect forgery on a block-basis. The image is first converted to a gray-scale (if input image is a color image) to reduce computational cost and further analysis for forgery detection is done with just one channel. A filter of a particular block size is slid across the image and then mean and standard deviation (SD) is calculated for all the pixel in it. Each of these values are appended to a matrix $data\_array$ along with the coordinates of the top left pixel of that image block. Thus, this matrix has four columns as mean, SD, x and y co-ordinates respectively.The matrix is sorted row-wise based on the mean values to arrange in order (either descending or ascending). Traverse the resulting matrix row-wise and for every element in sorted matrix $sorted\_array$, generate a Sub_array with $check\_offset$ as number of neighbors on both of its side. Compute absolute mean difference, absolute SD difference and euclidean distance between the element and its neighbors (elements of Sub_array).

Algorithm 1. SCMFD

INPUT: Image I of size $W \times H$, Mean_Ths, SD_Ths, Dist_Ths, SD_block,block_size

OUTPUT: Prediction Mask Bit Matrix PM of size $W \times H$

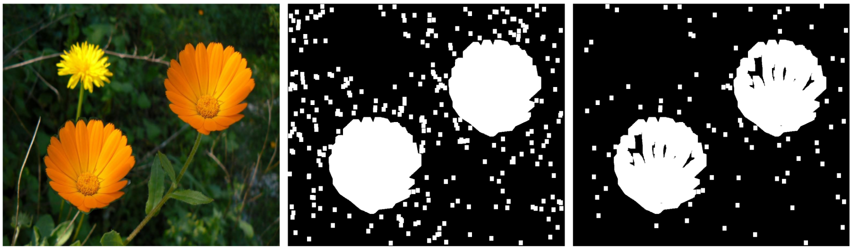
Define:
1. Mean_Ths is threshold for maximum threshold for absolute minimum difference between two image blocks to be identified as similar
2. SD_Ths is threshold for maximum threshold for absolute Standard deviation difference between two image blocks to be identified as similar
3. Dist_Ths is threshold for minimum threshold for euclidean distance between two image blocks to be identified as similar
4. SD_block is threshold for minimum threshold for image block to be considered for detection
5. block_size is block size for image blocks
6. PM is the output prediction mask of size $W \times H$. It is initialized with zeros

```
procedure SCMFD(I, Mean_Ths, SD_Ths, Dist_Ths, SD_block, block_size)
    Convert image to gray-scale
    data_matrix = Empty array
    for $i = 0$; i $<$ $H - block\_size$; $i = i + 1$ do
        data_array = Empty array
        for $j = 0$; $j$¡$W - block\_size$; $j = j + 1$ do
            block = I [i:i+block_size][j:j+block_size]
            data_array[0] = mean(block)
            data_array[1] = SD(block)
            data_array[2] = i
            data_array[3] = j
            append data_array to data_matrix
        end for
    end for
    Sort the data_matrix row-wise according to the mean value(key at j=0)
    Traverse in the resulting matrix row-wise
    Sim_array = Empty array
    for $i = 0$; i $<$ len(sorted_array); $i = i + 1$ do
        Sub_array = sorted_array[max(0,i-check_offset) : min(len(sorted_array), i+check_offset)]
        Calculate abs(mean difference), abs(SD difference) and distance between the element and its neighbors (elements of sub_array)
        if abs(mean difference) < Mean_Ths and SD of element > SD_block and, abs(SD difference) < SD_Ths and euclidean distance > Dist_Ths then
            append element to Sim_array
        end if
    end for
    for each element in Sim_array do
        PM(i:i+block_size,j:j+block_size) = 1
    end for
    return PM
end procedure
```

If absolute mean difference $< Mean\_Ths$, and absolute SD difference $< SD\_Ths$, and euclidean distance $> Dist\_Ths$, SD of element $> SD\_block$ then append element into a new array called Sim_array. The prediction mask PM is then created by highlighting blocks identified as similar.

3.2 Optimizations in SCMFD

We can reduce the false positives by comparing both mean value and SD of the blocks with a specified threshold. If the difference in mean value between two blocks is below this threshold $Mean\_Ths$ and if the standard deviation of these blocks is the same, then these blocks will be identified as similar by SCMFD algorithm. Image blocks that are physically closer in the image may also lead to false positives. Therefore we would only consider those block pairs where the sum of the number of similar blocks identified at a particular distance meets the minimum user-defined threshold $Dist\_Ths$. This way, we discard anomalies that are often next to each other or are often a part of the same object and not the corresponding copy-moved object. We also use another user-defined threshold SD_block to ensure that block pairs whose distance from each other is below a threshold are discarded as shown in Fig. 4.



(a) Image with copy move forgery

(b) SCMFD with SD_block = 2

(c) SCMFD with SD_block = 4

Fig. 4. PM using threshold SD_block on SD of individual image blocks

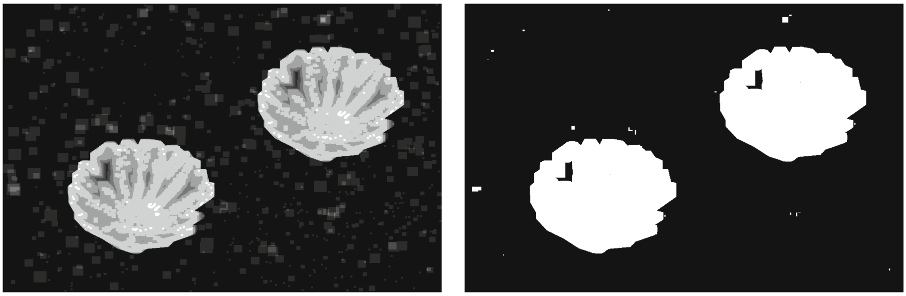
The proposed SCMFD algorithm does provide accuracy of 76% with FPR of 1.38% that increases with increase in $block\_size$ and decrease in SD_block ($Mean\_Ths$, SD_Ths and $Dist\_Ths$ kept as constant). It seems to be far less when compared to other statistical approaches for CMFD, but the big advantage of SCMFD is that it runs very fast as feature vector's length is just 2 containing only the mean value and SD value. This makes similarity check a lot easier among the blocks. Therefore, no PCA is needed. Also opposed to other statistical approaches, increasing block size increases performance (note that if the block size is large, there is a slight decrease in number of total blocks which slightly increases performance). To further increase the accuracy of the PM, we prepare the final PM by overlaying $PM's$ iteratively from the previous pass of the

SCMFD algorithm varying the block sizes *block_size* and *SD_block* for every pass.

$$PM_{mul} = \frac{\sum_0^N PM_i}{N} \tag{1}$$

where, PM_i represents the prediction mask at the i^{th} iteration and N as the total number of passes.

The final PM will have values ranging from 0 to 1. A simple filter is applied on this mask to change every value greater than a threshold to 1 and 0 otherwise (we used 0.4 in our study) as shown in Fig. 5. The resulting prediction mask gives us 97% accuracy on an average with 0.05% FPR.



(a) Multiple passes of SCMFD overlayed

(b) Applying filter

Fig. 5. Overlayed prediction mask

4 Experimental Results and Analysis

To evaluate the performance of the proposed approach, we have used Copy-Move Forgery Dataset [12]. Exhaustive set of experiments were conducted using the images from this standard dataset and overall F-score and FPR presented in the results.

4.1 Dataset

The Dataset [12] is made of medium sized images (almost all 1000 × 700 or 700 × 1000) and it is subdivided into several datasets (D0, D1, D2). We have used the subdivided dataset - D0 for our experiments which contains uncompressed images and their translated copies.

4.2 Results

The base algorithm SCMFD provides accuracy up to 76% with a FPR of 1.38%. The F-score and FPR are shown in Table 2 and Table 3 respectively, which shows

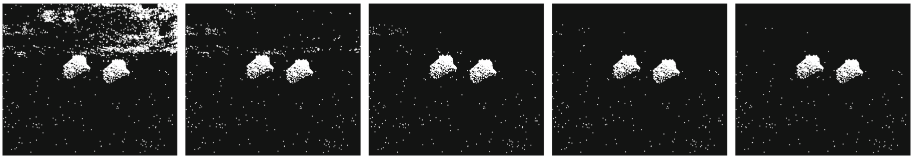
Table 2. Average values for F-score for different block sizes and respective *SD_block*

	SD_block				
block_size	0	1	2	3	4
5	21.01	26.93	35.70	41.83	45.81
10	34.67	40.26	52.97	60.30	64.84
15	40.55	45.61	57.87	65.47	70.64
20	44.67	49.17	60.45	68.17	73.39
25	47.62	51.62	62.28	71.06	76.56
30	50.82	54.87	64.43	73.31	76.60

Table 3. Average values for FPR for different block sizes and respective *SD_block*

	SD_block				
block_size	0	1	2	3	4
5	19.35	10.72	4.65	2.56	1.68
10	16.04	11.52	4.09	2.21	1.51
15	15.68	12.06	4.52	2.38	1.73
20	14.74	11.63	4.77	2.60	1.72
25	12.99	12.03	5.48	2.52	1.44
30	10.40	8.74	5.01	2.25	1.37

the increase in accuracy and decrease in FPR as we increase the *block_size* and *SD_block*. By increasing the *SD_block*, we see that the areas having repeated pattern or solid colors are reduced in the PM as shown in Fig. 6 moving from (a) to (e). And as we increase the *block_size*, we see that the false positives are reduced greatly that can be seen if we move from Fig: 6, 7, 8, 9, 10 and 11. The accuracy obtained by SCMFD are further optimize.

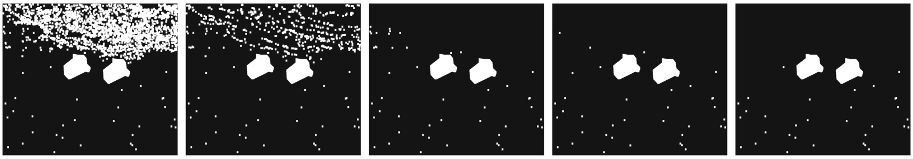


(a) SD_block = 0
(b) SD_block = 1
(c) SD_blocks = 2
(d) SD_block = 3
(e) SD_block = 4

Fig. 6. Prediction mask for block_size = 5 with different *SD_block*

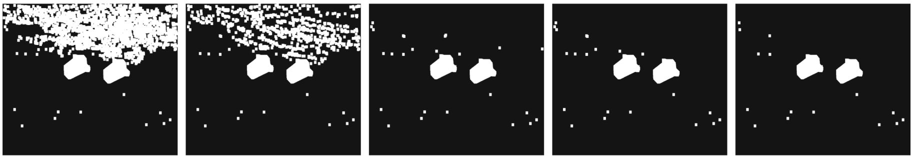
The results are further improved by doing multiple passes and overlaying different prediction mask from different passes of SCMFD with different block sizes and *SD_block*.

Table 4 shows how the accuracy is increased by overlapping prediction mask of different block-sizes and taking different standard deviations into consideration. By doing so, we get 5 different overlaying prediction masks which are shown in Fig. 12(a) prediction mask obtained by overlaying prediction mask of all block-sizes and all *SD_block* values. Figure 12(b) prediction mask obtained by overlaying prediction mask of all block-sizes with *SD_block* values = 1, 2, 3, 4. Figure 12(c) prediction mask obtained by overlaying prediction mask of all block-sizes with *SD_block* values = 2, 3, 4. Figure 12(d) prediction mask obtained by overlaying prediction mask of all block-sizes with *SD_block* values = 3, 4. Figure 12(e) prediction mask obtained by overlaying prediction mask of



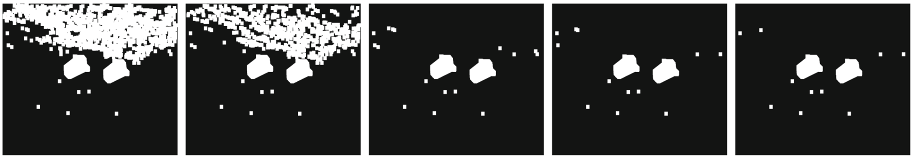
(a) $SD\_block = 0$ (b) $SD\_block = 1$ (c) $SD\_block = 2$ (d) $SD\_block = 3$ (e) $SD\_block = 4$

Fig. 7. Prediction mask for block_size = 10 with different *SD_block*



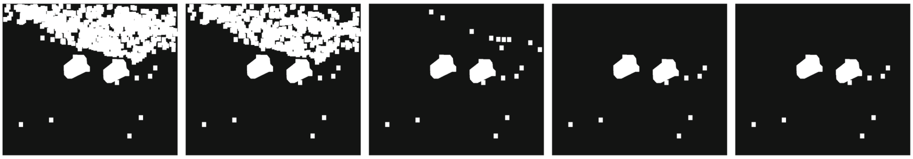
(a) $SD\_block = 0$ (b) $SD\_block = 1$ (c) $SD\_block = 2$ (d) $SD\_block = 3$ (e) $SD\_block = 4$

Fig. 8. Prediction mask for block_size = 15 with different *SD_block*



(a) $SD\_block = 0$ (b) $SD\_block = 1$ (c) $SD\_block = 2$ (d) $SD\_block = 3$ (e) $SD\_block = 4$

Fig. 9. Prediction mask for block_size = 20 with different *SD_block*



(a) $SD\_block = 0$ (b) $SD\_block = 1$ (c) $SD\_block = 2$ (d) $SD\_block = 3$ (e) $SD\_block = 4$

Fig. 10. Prediction mask for block_size = 25 with different *SD_block*

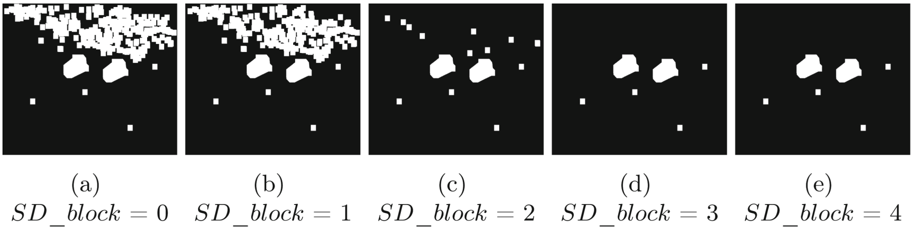


Fig. 11. Prediction mask for block_size = 30 with different *SD_block*

all block-sizes with *SD_block* values = 4. From these prediction masks, we can see that the FPR is decreased and the F-score is increased.

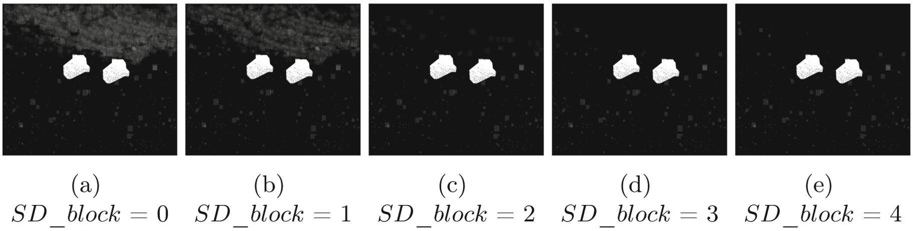


Fig. 12. Prediction mask obtained after doing multiple passes of SCMFD

Table 4. F-score and FPR after optimization

SD_block	F-score	FPR
[0,1,2,3,4]	86.14	1.9
[1,2,3,4]	88.892	1.023
[2,3,4]	96.487	0.114
[3,4]	96.644	0.095
[4]	97.047	0.051

4.3 Comparative Performance with State-of-the-art Approaches

The comparative performance of the proposed approach with the other state-of-the-art schemes in tabulated in the Tables 5 and Table 6 based on F-score and FPR respectively. High F-score of approximately 98.38% in case of DyWT_zernike [16] and Dixit et al. [3] is achieved which is marginally higher

Table 5. Comparative performance based on F-score

Schemes	F-score [%]
PCA [15]	96.78
DCT [2]	97.24
IDCT [14]	97.84
DyWT [9]	98.02
DWT [17]	98.09
DyWT_zernike [16]	98.38
SVD [11]	97.62
Dixit et al. [3]	98.38
Proposed SCMFD	97.05

Table 6. Comparative performance based on FPR

Schemes	FPR
PCA [15]	9.04
DCT [2]	11.33
IDCT [14]	9.81
DyWT [9]	12.91
DWT [17]	10.11
DyWT_zernike [16]	8.08
SVD [11]	7.87
Dixit et al. [3]	2.55
Proposed SCMFD	0.051

than that achieved by the proposed approach. However, these methods have approximate FPR of 8.08% and 2.55% which is much higher when compared to the proposed approach that attains FPR as low as 0.051%.

5 Conclusion

In this paper, we propose SCMFD approach that aims to accurately localize the forged portions within an image exploiting it's statistical properties. It utilizes the statistical image features viz., mean, and variance to detect forgery on a block-basis. The accuracy is further improved by preparing new prediction mask by overlaying different prediction mask from different passes of SCMFD. As our future work, we would like to make the proposed method robust against various attacks of noise addition, scaling, jpeg compression, etc.

References

1. Gupta, A., Saxena, N., Vasistha, S.K.: Detecting copy move forgery using DCT. Int. J. Sci. Res. Publ. **3**(5), 1 (2013)
2. Fridrich, A.J., Soukal, B.D., Lukáš, A.J.: Detection of copy-move forgery in digital images. In: Proceedings of Digital Forensic Research Workshop (2003)
3. Roy, A., Dixit, R., Naskar, R., Chakraborty, R.S.: Copy-move forgery detection exploiting statistical image features. Digital Image Forensics. SCI, vol. 755, pp. 57–64. Springer, Singapore (2020). https://doi.org/10.1007/978-981-10-7644-2_4
4. Li, G., Wu, Q., Tu, D., Sun, S.: A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD. In: 2007 IEEE International Conference on Multimedia and Expo, pp. 1750–1753. IEEE, July 2007

5. Bayram, S., Sencar, H.T., Memon, N.: An efficient and robust method for detecting copy-move forgery. In: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1053–1056. IEEE, April 2009
6. Zhao, J., Guo, J.: Passive forensics for copy-move image forgery using a method based on DCT and SVD. Forensic Sci. Int. **233**(1–3), 158–166 (2013)
7. Li, L., Li, S., Zhu, H., Chu, S.C., Roddick, J.F., Pan, J.S.: An efficient scheme for detecting copy-move forged images by local binary patterns. J. Inf. Hiding Multimed. Signal Process. **4**(1), 46–56 (2013)
8. Li, L., Li, S., Zhu, H., Wu, X.: Detecting copy-move forgery under affine transforms for image forensics. Comput. Electr. Eng. **40**(6), 1951–1962 (2014)
9. Muhammad, G., Hussain, M., Bebis, G.: Passive copy move image forgery detection using undecimated dyadic wavelet transform. Digit. Invest. **9**(1), 49–57 (2012)
10. Lee, J.C., Chang, C.P., Chen, W.K.: Detection of copy-move image forgery using histogram of orientated gradients. Inf. Sci. **321**, 250–262 (2015)
11. Kang, X., Wei, S.: Identifying tampered regions using singular value decomposition in digital image forensics. In: 2008 International Conference on Computer Science And Software Engineering, vol. 3, pp. 926–930. IEEE, December 2008
12. Diid.unipa.it. n.d. Download — CVIP Group. http://www.diid.unipa.it/cvip/?page_id=48#CMFD. Accessed 1 Aug 2020
13. Sunil, K., Jagan, D., Shaktidev, M.: DCT-PCA based method for copy-move forgery detection. In: Satapathy, S., Avadhani, P., Udgata, S., Lakshminarayana, S. (eds.) ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol II. Advances in Intelligent Systems and Computing, vol. 249, pp. 577–583. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-03095-1_62
14. Huang, Y., Lu, W., Sun, W., Long, D.: Improved DCT-based detection of copy-move forgery in images. Forensic Sci. Int. **206**(1–3), 178–184 (2011)
15. Popescu, A.C., Farid, H.: Exposing digital forgeries by detecting duplicated image regions. Department of Computer Science, rtmouth College, Technical report TR2004-515, pp. 1–11 (2004)
16. Yang, J., Ran, P., Xiao, D., Tan, J.: Digital image forgery forensics by using undecimated dyadic wavelet transform and Zernike moments. J. Comput. Inf. Syst. **9**(16), 6399–6408 (2013)
17. Zhang, J., Feng, Z., Su, Y.: A new approach for detecting copy-move forgery in digital images. In: 2008 11th IEEE Singapore International Conference on Communication Systems, pp. 362–366. IEEE, November 2008