A Fast, Block Based, Copy-Move Forgery Detection Approach Using Image Gradient and Modified K-Means

V. Hajihashemi[1] and A. Alavi Gharahbagh[2(✉)]

[1] Software Department, Bam Pardazesh Tehran Co., Tehran, Iran
Hajihashemi.vahid@yahoo.com
[2] Department of Electrical and Computer Engineering, Islamic Azad University, Shahrood Branch, Shahrood, Iran
Dramalavi_gharah@yahoo.com

Abstract. In recent years, due to the fast development of digital images, a rapid growth of research interest in the forgery detection in digital images has been happened. One of the most common techniques in creating forged images is copy-move (region duplication) technique. In this paper, a new method for copy-move forgery detection in digital images is proposed. In this paper a region duplication detection technique which utilizes the image gradient is proposed. In the proposed approach, first the gradient of image is divided into overlapped blocks. Using gradient versus other techniques, decreases processing time in feature extraction step.

A fast pre clustering algorithm is another added step to speedup method by dividing search area into some subset. The unknown parameters of proposed method are determined by implementing different conditions on two standard databases. Finally, the performance of the proposed method is compared with some state of art methods and the acceptable accuracy and lower run time of it, is verified.

Keywords: Image forgery · Copy-move · Image gradient · Fast k means · Forgery detection

1 Introduction

In recent years, due to the fast development of image processing techniques and graphics editors, digital images are easily and masterly forged. Image forgery detection have many applications such as forensic and criminal investigation, insurance industry, security systems, social networks, internet, medical imaging etc. One of the most common techniques in creating forged images is copy-move (region duplication) technique. In this method, one or some parts of an image is selected, copied and pasted onto other regions of the same image as shown in Fig. 1.

It is very important in internet or social networks to verify that the images are genuine, so many researches have been done for copy-move forgery detection (CMFD). Existing CMFD techniques can be classified into two major categories: block-based methods and keypoint based methods.

S.M. Thampi et al. (eds.), *Intelligent Systems Technologies and Applications*,
Advances in Intelligent Systems and Computing 683,
https://doi.org/10.1007/978-3-319-68385-0_25

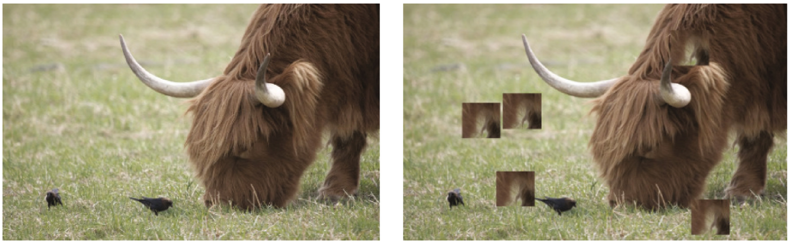

Fig. 1. Original image (right) and forged Image (left)

In block-based CMFD method [1, 2], image is divided into fixed-size overlapping blocks and then similar blocks searched and found. The main differences between these methods are the feature extraction strategy and search area reduction techniques [3]. Because of the large number of blocks, in general, block-based methods are computationally so expensive and the computationally burden of feature extraction algorithm and the preprocessing technique for eliminating some blocks from comparison loop is so important [4].

Keypoints-based CMFD methods is approximately different. In these methods at first step, keypoints are detected in image [5]. After this step, similar regions only based on keypoints, searched and found. Obviously keypoints are much less than total overlapped blocks of an image, so keypoints based method are faster than block based methods. The main challenge in these methods are keypoints detection algorithm.

2 Related Work

The main idea in CMFD algorithms is to search and detect similar blocks or regions in forged images. The speed, robustness and accuracy of this detection are some measures used for evaluating effectiveness of different algorithms. considering the scope of the proposed method, in this section only focused on the work that improve copy move forgery detection techniques. [6] suggested a robust approach which use a combination of human vision information and moment invariants features to represent images. In [6] feature vector is a set of color perception and object representation features. [7] used Speed-Up Robust Features (SURF), Histogram Oriented Gradient (HOG) and Scale Invariant Features Transform(SIFT) for CMFD. [8] using SURF similar to [7], but try to improve keypoints detection by an extra step before extracting SURF features. This step uses a single image super resolution (SISR) algorithm. The [8] method showed accurate forgery detection even when the forgery region size is small. [9] suggested mirror-reflection invariant feature transform (MIFT) as an alternative for SIFT. Flipped images are moving images that is generated by a mirror-reversal of original images. [10] is similar to [7, 8] and used SURF feature. For solving the problem of number of keypoints, [10] improve keypoints detection step by particle swarm optimization (PSO). [11, 12] used Discrete Cosine Transform (DCT) for forgery detection. [11] suggested a dynamic threshold to discard large flat regions of the image. Using this threshold, the number of candidate blocks were decreased and the speed of method was increased. [12] decreased feature vector size in DCT by applying a Zigzag scanning to DCT matrix. In the meantime, in [12] a fast k-means clustering method was utilized to divide blocks to some subsets and reduce the computational burden. In [13, 14] discrete wavelet transform (DWT) was selected for extracting features from image blocks. [13] used stationary wavelet transform and [14] used undecimated dyadic Wavelet Transform. The main difference between [13] and [14] methods are its distance measure that are Euclidean distance in [13] and Canberra distance in [14]. In [15] method a PSO optimization is added to CMFD process to estimate similar regions that are not forged. [16] proposed a key point CMFD method that used Harris corner detection algorithm to find region of interest area (ROI) and eliminate unnoticeable blocks. In [17] a deep learning method is utilized for CMFD. This method did not use any conventional feature extraction method. In deep learning of [17], after patch sampling, by combination of a preprocess including high pass filtering and spatial rich model try to capture artifacts in image. Many other researches has been done in this topic specially in key point based CMFD methods. As a good survey, [18] review some of these works.

All of the aforementioned methods have some limitations. The main challenge in CMFD is computational burden. Specially in high quality images this burden is not acceptable. Key point based methods, try to reduce run time but the accuracy of them is usually lower than block based methods.

In this paper, we propose a system including fast feature extraction process and an extra preprocess step for eliminate some unimportant areas. The other advantage of this step is, its feature participation in final feature vector. Based on this modification the computational cost of total CMFD process is obviously decreased without any accuracy lost. The rest of paper is organized as follows: Sect. 2 explains the related work. Then, the proposed method is clarified in Sect. 3. Comparison with other CMFD algorithms by using a standard dataset is presented in Sect. 4. Finally, conclusion is drawn in Sect. 5.

3 Proposed Method

The details of the proposed algorithm steps are explained in following subsections.

3.1 Image Resizing and Converting to Gray

The proposed algorithm considers the grayscale images. At the first step, if the input image is a color image, it should be transformed to grayscale. In the meantime, the size of the input image directly influences the computation time of CMFD method.

Based on analyzing block size and other conditions, the images larger than 512×512 can be resized to a lower resolution without any decrease in accuracy. For comparing to other methods in similar conditions the resizing step do not apply to images but as an extra step the results of image resizing are given in result section.

3.2 Image Gradient

There are number of ways to extract the unnoticeable pixels from an image. In our CMFD method, an energy value is assigned to each pixel by using gradient energy function that is defined in Eq. (1).

$$e(x,y) = \left|\frac{\partial I}{\partial x}\right| + \left|\frac{\partial I}{\partial y}\right| \tag{1}$$

This value can be easily computed by Sobel masks in both horizontal and vertical directions [19]. The gradient operator is independent of image blocking so the gradient operator can be applied to whole image before blocking. Extracting features from whole image without blocking is so faster than extracting feature vector from overlapping blocks separately. Another important usage of gradient image is discarding unnoticeable blocks in preprocess step. Figure 2 shows original forged image and its gradient.

Fig. 2. CMF image and its gradient

3.3 Image Blocking

In this step, the gradient of input image is divided into overlapping square blocks (b × b pixels). If the image size is M × N with overlapping, we have total (M−b + 1) × (N−b + 1) blocks in the whole image. For a 128 × 128 image and block size 8 × 8, the total number of blocks is 14641 that with increasing image size this value increase so fast.

3.4 Gradient Threshold

One of the main goals of all CMFD methods is speed of execution. In our method for earning speed, we try to decrease number of blocks before feature extraction step. For eliminating some blocks, the gradient energy used as a fast descriptor. All forged areas based on its nature have discontinuous borders. If total energy of a block be smaller than a specified threshold, block does not include any border and can be eliminated from next steps and vice versa. All smooth surfaces of image are discarded using this step. For a sample image about 25% of blocks is eliminated in this step that means the total process of our method is about 25% faster than conventional CMFD block-based method. In Fig. 2 the forged region borders are obviously highlight in the gradient image and the low energy of pixels in the smooth surfaces such as meadow is shown by dark areas.

3.5 Feature Extraction

In this stage, singular value decomposition (SVD) is utilized for feature extraction. In fact, the SVD summarizes some properties of a matrix. In the proposed method SVD is applied to each remained block. If the size of the block is b $\times$ b, there are at most b singular value for the block, so the proposed method does not need feature reduction step similar to DCT [12] or SIFT features. Some of the singular values is so lower than others and in many matrixes the number of singular values is lower than b (the size of matrix). According to the above reasons, in the proposed method only the 5 first values of SVD are selected as feature vector. If the number of SVD values in a block less than 5, for matching all blocks feature vectors, the remained singular values is supposed 0.

The output of SVD method is sorted descending so the proposed method does not need extra step for sorting each feature vector. In the meantime, the first SVD value computed in this step that is the largest, is used in the next step for clustering blocks into similar groups.

3.6 Modified K-Means and Initial Clustering

The most common method for clustering is k-means. The conventional k-means algorithm is slow for large datasets so in the proposed method a modified accelerated k-means [20] is used for clustering. In conventional k-means most distance calculations are redundant and can be eliminated. In addition, if a point is far away from center, it is not necessary to calculate the exact distance between the point and center, if a point is much closer to one center than other centers, calculating distance is not necessary and this point should be assigned to this center. The accelerated K-Means algorithm [20] avoids unnecessary these unnecessary calculations by applying the triangle inequality and keeping track of lower and upper bounds for distances between points and centers. The upper and lower bounds are usually tight for most points and centers so the updated bounds tend to be tight at the start of next iteration. In fact, the main effectiveness of this method is, reducing the number of distance calculations at the start of each iteration. [20] results showed an about 7% to 300% speedup (based on the number of clusters) in this method in comparison to standard k-means.

In our work this k-means is about 10% faster than standard k-means. It should be noted that k-means is not the main time consuming step of the proposed CMFD process, but surely modification of this process can be accelerated total process. Before finding matched blocks in the final stage, a fast high-performance parallel radix sort routine for many core GPUs [21] is used for sorting feature vectors in each class. In the proposed method, k-means divided feature vectors into some clusters and the sort routine can be applied to each cluster via a parallel form, so the selected algorithm is fully better than standard quick sort method. The number of clustered is specified in experimental results section.

3.7 Find Matched Blocks

After sorting feature vectors in each cluster, each feature vector only is compared to next feature vector. This means the proposed method only need M comparison in each cluster that M is the number of blocks belong to cluster. For comparing feature vectors, the simple absolute percentage error with a threshold is used. Firstly, the ratio between all feature vectors elements with the same element in the next block is computed. The mathematical explanation of the matching process is given in Eq. 2.

$$Measure = 100 * abs\left(1 - \frac{Feature\_vector_{i+1}}{Feature\_vector_i}\right) \tag{2}$$

If any of feature vector elements, tend to zero, this element will be discarded from Eq. 2 to avoid error. This measure is applied to each element of feature vectors separately and finally results is compared to threshold.

The threshold value is set to 2 based on analyzing differences between feature vectors in similar regions in noisy and ideal conditions. If this measure in all elements are lower than threshold two blocks are marked as matched or forgery region.

4 Experiment Results

The proposed method was implemented in Matlab 2016a on a computer with CPU core i5-4430, 3.0 GHz with memory 8 GB. The images used in the simulation step, were taken from Christlein et al.'s database [22] and Ng et al.'s database [23].

The properties of images in these two datasets are different. There are 48 tampered images with high resolution in the Christlein et al.'s database. The size of images is about 3000×2300 pixels. The images in Ng et al.'s database are 128×128 pixels gray BMP format image. One of the main parameters in the proposed method is detection time. By choosing these two different datasets, we able to compare detection time of the proposed method with other methods accurately. In order to decrease run time, all images that the sum of height and width of image is more than 1024 are resized to half size of original image, for example a 3000×2300 image is resized to 1500×1150.

The main parameters in the proposed method were set as: Block size = [4…16], Gradient threshold = 4.68, Minimum block distance between detected forged regions = 2 * Block size, measure for finding matched blocks = 2. The block size is changed between 4 to 16 to test this effect on accuracy and run time of the proposed method. Gradient threshold, Minimum block distance and measure for finding matched blocks were chosen based on the best results (maximize accuracy).

Many metrics have been used in researches for quantifying performance of CMFD methods. The accuracy and robustness [13], Precision and recall [12], True Positive and false positive Rate [9] and finally run time are some of them. In this research we used Precision and recall as a two standard metrics for showing performance of the proposed method and run time for showing processing time. precision (positive predictive value) and recall (sensitivity) are defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

where True Positive (TP) indicates count of successfully detected forged images. True Negative (TN) is count of normal images successfully labelled by proposed method. False Positive (FP) is the number of normal images detected as forged. False Negative (FN) is number of forged images labelled as normal.

Test (A) Normal CMF Detection (Rectangle region): In this test a normal tampered image without any noise or compression is used for CMFD. The tampered region has fully rectangular form. Figure 3(a, c) and (b, d) show the input forged images and analyzed images respectively. As shown in Fig. 3(b and d) the proposed method has correctly detected copy-move regions in the input tampered images. In the case of Fig. 3(c and d) that the number of forged regions is more than one, only one region did not detect accurately.

Test (B) Normal CMF Detection (irregular region): similar to test A, normal tampered image without any noise or compression is used for CMFD but the tampered area is irregular. Figure 4(a and b) show the input forged image and analyzed image respectively. As shown in Fig. 4 the proposed method has correctly detected copy-move regions in the input tampered images even in irregular form.

Test (C) The unknown parameters of proposed method are number of clusters in initial clustering and the block size. For analyzing these parameters effect on our method, the proposed method applied to 120 images with similar size and the mean of Precision, recall and processing time are computed. Obviously the lower runtime and higher precision and recall value is desirable. The Table 1 shows results. Based on the results of Table 1 the block size 8 and N of clusters 4 are the best values. In the final step the proposed method is compared with some state of art CMFD methods that had similar conditions. For comparing processing time correctly, all methods were implemented on similar hardware. The results are given in Table 2. The precision and recall are given only with one decimal place based on similar works.

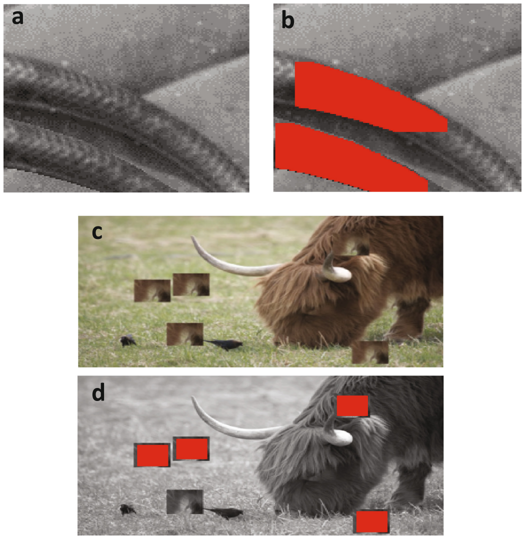

Fig. 3. (a, c) show CMF images and (b, d) the proposed method output

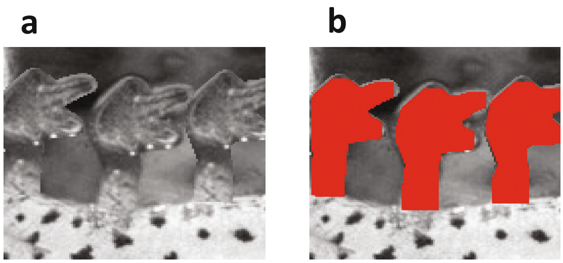

Fig. 4. (a) CMF images and (b) the proposed method output

Table 1. The processing time, precision and recall for different block size and number of clusters.

Block size	N of clusters	Processing time (s)	Precision	Recall
4	4	2.9	87.3	98.1
	8	3.07	88.6	98
	12	3.15	91	98
	16	3.26	94	98
8	4	2.82	99.1	99.4
	8	2.95	98.2	99.2
	12	3.11	97	99.1
	16	3.24	97.3	99
12	4	2.68	96	99.2
	8	2.9	97	99.4
	12	3.22	96	99.1
	16	3.61	94	99

Table 2. The processing time, precision and recall for comparing proposed method to different methods

Ref	Processing time (s)	Precision	Recall
[12]	8.95	99.1	99.1
[14]	10.14	99.1	98.3
[16]	6.64	97.5	99.1
[17]	14.2	95.8	99.1
Proposed	2.82	99.1	98.3

5 Conclusion

In this paper, a new method for CMFD in digital images is proposed. The proposed method decreases processing time in feature extraction step. A fast pre clustering algorithm is another added step to proposed method, to divide search area into some subsets and speedup method. The performance of the proposed method has been analyzed in terms of precision, recall and run time. The unknown parameters of proposed method are determined by implementing different conditions on two standard databases. Finally, the proposed method is compared with some state of art methods and the acceptable accuracy and lower run time of it, is verified. As a future work the proposed method should be modified to identify geometrically or noisy transformed duplicated image regions.

References

1. 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, Taipei, pp. 1053–1056 (2009)
2. Lin, H.J., Wang, C.W., Kao, Y.T.: Fast copy-move forgery detection. WSEAS Trans. Signal Process. **5**(5), 188–197 (2009)
3. Zimba, M., Sun, X.: DWT-PCA(EVD) based copy-move image forgery detection. Int. J. Digit. Content Technol. Appl. **5**, 19–29 (2011)
4. Hu, J., Zhang, H., Gao, Q., Huang, H.: An improved lexicographical sort algorithm of copy-move forgery detection. In: 2011 Second International Conference on Networking and Distributed Computing, Beijing, pp. 23–27 (2011)
5. Li, L., et al.: Detecting copy-move forgery under affine transforms for image forensics. Comput. Electr. Eng. **40**(6), 1951–1962 (2014)
6. Kushol, R., Salekin, M.S., Kabir, M.H., Khan, A.A.: Copy-move forgery detection using color space and moment invariants-based features. In: 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, QLD, pp. 1–6 (2016)
7. Pandey, R.C., Agrawal, R., Singh, S.K., Shukla, K.K.: Passive copy move forgery detection using SURF, HOG and SIFT features. In: Satapathy, S., Biswal, B., Udgata, S., Mandal, J. (eds.) Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014. Advances in Intelligent Systems and Computing, vol 327. Springer, Cham (2014)

8. Al-Hammadi, M.M., Emmanuel, S.: Improving SURF based copy-move forgery detection using super resolution. In: 2016 IEEE International Symposium on Multimedia (ISM), San Jose, CA, pp. 341–344 (2016)
9. Agarwal, V., Mane, V.: Reflective SIFT for improving the detection of copy-move image forgery. In: 2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, 2016, pp. 84–88
10. Wenchang, S., Fei, Z., Bo, Q., Bin, L.: Improving image copy-move forgery detection with particle swarm optimization techniques. China Commun. **13**(1), 139–149 (2016)
11. Moradi-Gharghani, H., Nasri, M.: A new block-based copy-move forgery detection method in digital images. In: 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, pp. 1208–1212 (2016)
12. Fadl, S.M., Semary, N.A.: A proposed accelerated image copy-move forgery detection. In: 2014 IEEE Visual Communications and Image Processing Conference, Valletta, pp. 253–257 (2014)
13. Mahmood, T., Nawaz, T., Mehmood, Z., Khan, Z., Shah, M., Ashraf, R.: Forensic analysis of copy-move forgery in digital images using the stationary wavelets. In: 2016 Sixth International Conference on Innovative Computing Technology (INTECH), Dublin, 2016
14. Dixit, R., Naskar, R.: DyWT based copy-move forgery detection with improved detection accuracy. In: 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, pp. 133–138 (2016)
15. Zhao, F., Shi, W., Qin, B., Liang, B.: A copy-move forgery detection scheme with improved Clone Region Estimation. In: 2016 Third International Conference on Trustworthy Systems and their Applications (TSA), Wuhan, pp. 8–16 (2016)
16. Isaac, M.M., Wilscy, M.: A key point based copy-move forgery detection using HOG features. In: 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, pp. 1–6 (2016)
17. Rao, Y., Ni, J.: A deep learning approach to detection of splicing and copy-move forgeries in images. In: 2016 IEEE International Workshop on Information Forensics and Security (WIFS), Abu Dhabi, pp. 1–6 (2016)
18. Warbhe, A.D., Dharaskar, R.V., Thakare, V.M.: A survey on keypoint based copy-paste forgery detection techniques. Proc. Comput. Sci. **78**, 61–67. ISSN 1877-0509 (2016)
19. Scharr, H.: Optimal second order derivative filter families for transparent motion estimation. In: 2007 15th European Signal Processing Conference, Poznan, pp. 302–306 (2007)
20. Elkan, C.: Using the triangle inequality to accelerate k-means. In: Proceedings of the Twentieth International Conference on Machine Learning (ICML), Washington DC, USA, vol. 3 (2003)
21. Satish, N., Harris, M., Garland, M.: Designing efficient sorting algorithms for manycore GPUs. In: 2009 IEEE International Symposium on Parallel & Distributed Processing, Rome, pp. 1–10 (2009)
22. Christlein, V., Riess, C., Jordan, J., Riess, C., Angelopoulou, E.: An evaluation of popular copy-move forgery detection approaches. IEEE Trans. Inf. Forensics Secur. **7**(6), 1841–1854 (2012)
23. Ng, T.T., Hsu, J., Chang, S.F.: Columbia Image Splicing Detection Evaluation Dataset. http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataSet/AuthSplicedDataSet