BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 08/10 pp347–360
Volume 6, Number 3, September 2023
DOI: 10.26599/BDMA.2022.9020029

Copy-Move Forgery Verification in Images Using Local Feature Extractors and Optimized Classifiers

S. B. G. Tilak Babu* and Ch Srinivasa Rao

Abstract: Passive image forgery detection methods that identify forgeries without prior knowledge have become a key research focus. In copy-move forgery, the assailant intends to hide a portion of an image by pasting other portions of the same image. The detection of such manipulations in images has great demand in legal evidence, forensic investigation, and many other fields. The paper aims to present copy-move forgery detection algorithms with the help of advanced feature descriptors, such as local ternary pattern, local phase quantization, local Gabor binary pattern histogram sequence, Weber local descriptor, and local monotonic pattern, and classifiers such as optimized support vector machine and optimized NBC. The proposed algorithms can classify an image efficiently as either copy-move forged or authenticated, even if the test image is subjected to attacks such as JPEG compression, scaling, rotation, and brightness variation. CoMoFoD, CASIA, and MICC datasets and a combination of CoMoFoD and CASIA datasets images are used to quantify the performance of the proposed algorithms. The proposed algorithms are more efficient than state-of-the-art algorithms even though the suspected image is post-processed.

Key words: copy move forgery detection; image authentication; passive image forgery detection; blind forgery detection

1 Introduction

With the rapid advancement of image processing technologies, modifying a digital image becomes simpler even for an amateur forger because of the availability of easy-to-use photo editing software such as Adobe Photoshop and GIMP. In the last few years, image forgery detection methods that find forgeries without any prior knowledge have been the main research focus.

The detection methods are classified based on the forgery detection procedure that is used. The classification taxonomy is shown in Fig. 1. Forgery detection methods are categorized into active and passive[1,2]. Active methods[3,4] need prior knowledge of the suspected image, whereas passive methods[5–9] can perform detection without any prior knowledge. One of the main subtopics in passive image forgery detection is copy-move forgery detection (CMFD)[10–12]. CMF aims to hide important information, and CMFD detects the copied part pasted in the same image. CMFD has a wide range of applications in forensic investigation and legal evidence. An example of CMF is shown in Fig. 2.

Many CMFD algorithms, such as block matching[13–15] and keypoint matching based[16,17] algorithms, are proposed in the literature, and each has its advantages and disadvantages. Block-based CMFD algorithms are time consuming and fail to identify if the copied and pasted regions are closely present in the suspected image. Keypoint-based CMFD algorithms fail to identify copied and pasted regions in smooth areas. Both algorithms have a bottleneck for the threshold value for assigning similarity among blocks/keypoints and exhibit poor performance in post-processing attacked images. Usually, in block-based algorithms, the suspected image

• S. B. G. Tilak Babu is with the University College of Engineering, JNTU Kakinada, Kakinada 533003, India. E-mail: thilaksayila@gmail.com.
• Ch Srinivasa Rao is with the Department of ECE, JNTUK UCE, Vizianagaram 535003, India. E-mail: chsrao.ece@jntukucev.ac.in.
* To whom correspondence should be addressed.
Manuscript received: 2022-03-02; revised: 2022-06-27; accepted: 2022-07-25

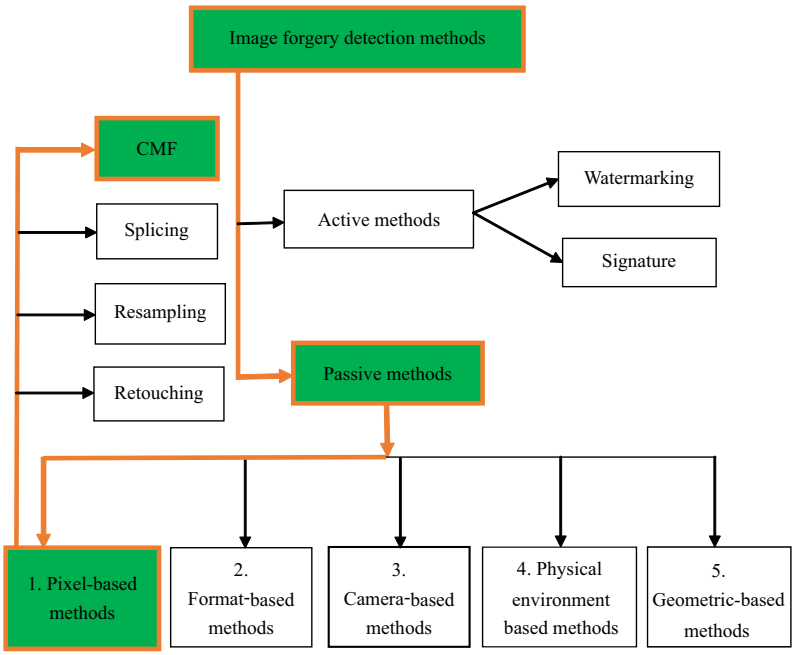

Fig. 1 Various image forgery detection methods.

(a) Authentic image (b) Forged image (c) Forgery marked image

Fig. 2 CMF example.

is first converted into a single-layer image, such as a grayscale image from a color image, and then converted into rectangular or circular overlapping blocks[15, 18, 19]. On each overlapping block, features are extracted using principal component analysis[18], local binary pattern (LBP)[15], singular value decomposition[20], or any other tool. Among the extracted block features, the similarity is verified using distance measure (often Euclidean distance), PatchMatch, and correlation. During the initial identification of similar blocks, a possibility exists that false matches will be obtained. These false matches can be eliminated by using morphological operators[21, 22], random sample consensus, or window method. In keypoint-based CMFD algorithms, keypoints are extracted from the suspected image by using scale-invariant feature transform[23, 24], speeded-up robust feature (SURF)[25, 26], and binary robust invariant scalable keypoints[17]. Among these extracted keypoints, the similarity is verified, and false matches are eliminated using post-processing operators. Apart from these two conventional CMFD algorithms, another widely used technique is the classification of a suspected image as either authentic or forged by using classifiers such as support vector machine (SVM)[27–29]. The proposed CMFD algorithms belong to the classification category, require less computational time, and are more efficient at classifying a suspected picture as either authentic or forged.

In the proposed CMFD algorithm, every feature extractor undergoes three phases: extraction of features from an image, obtaining the histogram of features, and concatenation of obtained feature histograms in different orientations of the image. A brief discussion on each feature extractor, along with necessary mathematical analysis, strengths, and weaknesses, is presented in Section 2. In Section 3, a flowchart for CMFD using local feature extractors (LFEs) and its step-by-step details, hardware and software requirements, datasets, and performance metrics used in experimentation are presented. The test findings are tabulated and given in

Section 4. Finally, the conclusions and implications from tests are presented in Section 5.

2 Feature Extraction

Many texture feature extractors have been used to identify CMF. In the proposed work, new texture feature extractors are used for identifying CMF and then compared with the results in the literature. In 2002, Ojala et al.[30] proposed a rotation and grayscale invariant texture feature extractor for monochrome texture images called LBP. Researchers have since modified LBP based on their requirements for different applications such as local ternary pattern (LTeP)[31,32], local phase quantization (LPQ)[33], local Gabor binary pattern histogram sequence (LGBPHS)[34], local directional pattern (LDP)[35], and local arc pattern[36]. On the basis of their merits, suitability for problems, and empirical analysis, LTeP, LPQ, and LGBPHS are selected in the proposed CMF algorithms to authenticate an image.

The LBP of a pixel at location (x, y) can be calculated by Eq. (1).

$$\mathrm{LBP}_{P,R}(x, y) = \sum_{n=0}^{P-1} S(I_n - I_c)2^n \tag{1}$$

where P and R represent the neighbor pixel number and distance to the neighbor pixel from the center pixel, respectively, I_c represents a center pixel, and I_n represents the neighbor pixel to I_c from distance R.

In Eq. (1), the S function represents the sign function, which results in either 1, if $(I_n - I_c)$ is greater than or equal to zero or 0, if $(I_n - I_c)$ is less than zero. The mathematical representation of the sign function is as follows:

$$S(x) = \begin{cases} 1, & x \geqslant 0; \\ 0, & x < 0 \end{cases} \tag{2}$$

Figure 3 depicts the LBP code generation procedure of the center pixel I_c. It is set as 55. The LBP is rotation invariant, which is why one can consider coding from any neighbor pixel. In this work, all coding considerations are taken from the right side of the center pixel.

In the two important steps of LBP, namely, thresholding and encoding, slight modifications are performed to overcome its drawbacks. LTeP is a slightly modified threshold of LBP and is used for many applications, such as facial recognition and image retrieval because it overcomes noise and false coding. The mathematical formulation of LTeP and LBP is almost the same except in the sign calculation of LBP, as presented in Eq. (3).

$$\mathrm{LTeP}_{P,R}(x, y) = \sum_{n=0}^{N} S(I_n - I_c)\, 2^n,$$
$$S(x) = \begin{cases} 1, & x \geqslant T; \\ 0, & -T < x < T; \\ -1, & x \leqslant -T \end{cases} \tag{3}$$

An example of the LTeP code generation procedure to the center pixel, $I_c = 55$, is depicted in Fig. 4. Similar to LBP, LTeP is rotation invariant. Therefore, one can consider coding from any neighbor pixel. In this work, coding considerations are taken from the right side of the center pixel.

Similar to LBP and LTeP, LMP in a picture region can be calculated as

$$\mathrm{LMP}_{P,R_1,R_2}(x, y) = \sum_{n=0}^{N} [S(I_{n_1} - I_c) \wedge S(I_{n_2} - I_{n_1})]2^n \tag{4}$$

An example of the LMP code generation procedure to the center pixel, $I_c = 83$, is depicted in Fig. 5. A descriptor that recognizes patterns in the same way as the human perception of patterns is the Weber local descriptor (WLD)[37]. The WLD features are concatenated histogram features of differential excitation (relative intensity) and orientation (gradient). The differential excitation can be calculated using Eq. (5) based on the relative intensities between the center and neighbor pixels.

$$\alpha = \arctan\left\{ \sum_{n=0}^{N} \frac{I_n - I_c}{I_c} \right\} \tag{5}$$

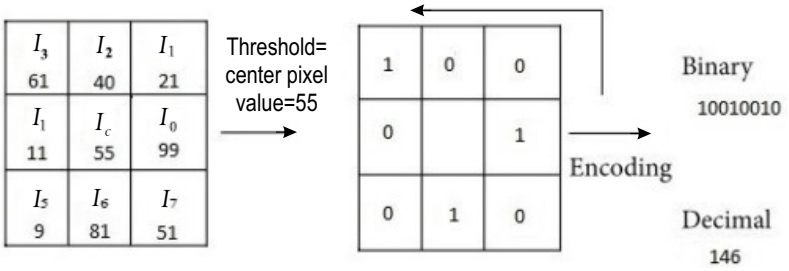

Fig. 3 LBP code generation procedure of the center pixel.

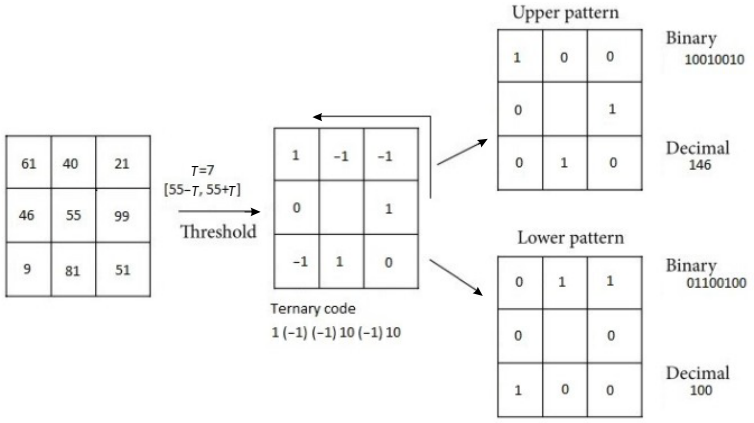

Fig. 4 LTeP code generation procedure of the center pixel.

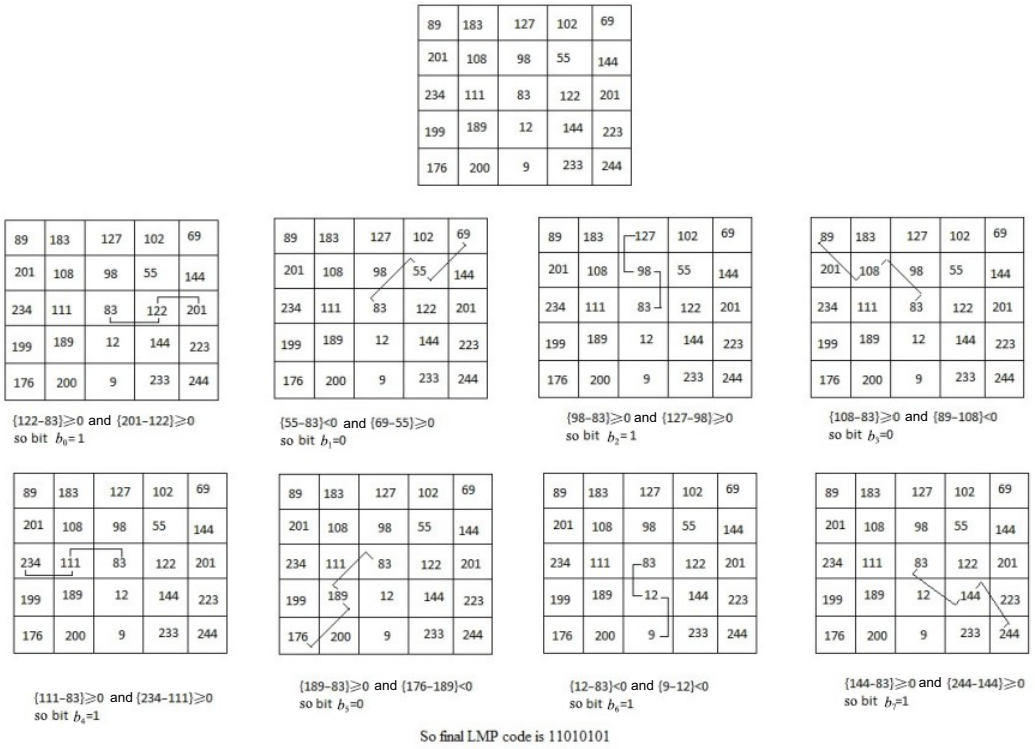

Fig. 5 LMP code generation procedure of the center pixel.

The WLD orientation (ϕ_t) can be calculated using Eq. (6). In Eqs. (2)–(6), the variables I_n and I_c are defined in Eq. (1), and T is the number of parts in angle θ.

$$\phi_t = \frac{2t}{T}\prod \text{ and } t = \text{ mod } \left\{\left\lfloor \frac{\theta'}{2\pi/T} + \frac{1}{2}\right\rfloor, T\right\} \quad (6)$$

$$\theta' = \begin{cases} \theta, & \text{if } A > 0 \text{ and } B > 0; \\ \theta + \pi, & \text{if } A < 0 \text{ and } B > 0; \\ \theta + \pi, & \text{if } A < 0 \text{ and } B < 0; \\ \theta + 2\pi, & \text{if } A > 0 \text{ and } B < 0. \end{cases}$$

$$\theta = \arctan\left(\frac{A}{B}\right) = \arctan\left(\frac{I_7 - I_3}{I_5 - I_1}\right),$$

where A and B are differences between pix values, $A = b_5 - b_1$, $B = b_7 - b_3$.

The gray-level co-occurrence matrix (GLCM) system is a method of expressing second-order statistical texture features. GLCM calculates the exact way that a pixel using intensity i occur—vertically or directly—into some pixel by using strength j. The statistical parameters considered in the proposed CMFD method are given below.

$$Energy = \sum_{i,j=0}^{N-1} (P_y)^2,$$

$$Correlation = \frac{\sum_{i,j=0}^{N-1} \left(P_y\,(i-\mu)\,(j-\mu)\right)}{\sigma^2},$$

$$Entropy = \sum_{i,j=0}^{N-1} \left(\frac{P_y}{1 + (i-j)^2} \right),$$

$$Contrast = \sum_{i,j=0}^{N-1} \left(P_y \, (i-j)^2 \right).$$

GLCM is also used to calculate the remaining features, but the above mentioned are commonly used powerful features[38]. Fourier phase spectrum (FPS) has the great advantage of blur invariance, which is effectively used in LPQ. LPQ preserves the maximum amount of information as FPS coefficients are decor related before the quantization. Finally, quantized coefficients range from 0 to 255 integer values. From empirical results, the final coefficients of the LPQ may vary if the image block is rotated in eight dimensions. This issue needs to be addressed in further extensions of LPQ. LGBPHS has the advantages of the Gabor filter and LBP. In the first step of LGBPHS, the Gabor filter is applied to the input image, which yields Gabor magnitude pictures (GMP), and the local Gabor binary pattern (LGBP) can be obtained from GMP with the help of LBP. Finally, the histogram is calculated for each non-overlapping region of LGBP.

The local feature extractor (LFE) output's histogram $H(m)$ can be calculated by Eq. (7).

$$H\,(m) = \sum_{i=1}^{K} \sum_{j=1}^{L} f\,(L\,(i,j)\,,m) \tag{7}$$

where $L(i,j)$ is the obtained feature value from the image, and m is the bin number.

3 Experiment

This section presents the process of CMFD using LFEs and its step-by-step details, hardware and software requirements, datasets, and performance metrics used in CMFD experiments. The design flow of the experiment is given below.

- A color image is converted into a monochrome image—that is, the given color image is converted into a YCbCr image. Chrominance components are more helpful than luminance components. Thus, chrominance blue (Cb) components are preferred in this experiment.
- These chrominance components are given to steerable pyramid transform to obtain images with various orientations[39]. The multiscale, multi-oriented steerable pyramid is decomposed linearly. Its basis functions are the K-th order directional derivative operators in different dimensions and $K+1$ directions.
- The features of each oriented image are obtained using LFE. Each of these oriented image histogram features is concatenated.
- These features are used to train the classifier or used to test the trained classier.
- The SVM model with sequential minima optimization (SMO) is used for training and validation on binary classification[40,41]. Aside from SVM-SMO, a naïve Bayes classifier with Bayesian optimization is used to test CMFD performance[42,43].

The same flow is shown in Fig. 6. A Dell Inspiron laptop with 8 GB RAM and AMD 7-th generation processor is used for experimentation. The simulations are conducted by using MATLAB 1.0.0.1 (2018a). The CASIA[44], CoMoFoDo[45], and MICC[46] datasets are used to evaluate the proposed CMFD algorithms. A combination of these two datasets is also explored in this work. The standard datasets have limited images with post-processing attacks. Few post-processed images are created using Adobe Photoshop. True positive rate (TPR), true negative rate (TNR), false positive rate (FPR), and false negative rate (FNR)[47] metrics are used to calculate the performance of the proposed CMFD technique.

4 Result and Discussion

The classifiers used in the experimentation are trained with 20 images (10 authentic + 10 counterfeit images) at one instance and then tested with 100 suspected images. Similarly, the classifiers are trained with 40 images (20 authentic + 20 counterfeit images), 50 images (25 authentic + 25 counterfeit images), 80 images (40 authentic + 40 counterfeit images), 100 images (50 authentic + 50 counterfeit images), and tested with 100 images at all individual training instants. As the number of training images increased, the performance improved slightly, but the computation time increased rapidly. Experiments show that the performance for 100 training images is slightly higher than that for 40 training images, but the computational time for 100 training images is much higher than that for 40 training images. In this work, the CMFD algorithms are trained with 40 images (from standard datasets, 20 authentic images and a set of 20 plain CMF images/a set of 20 post-processed CMF images).

To verify the efficacy of the proposed CMFD algorithms, 100 plain CMF images from the CASIA dataset and another 100 plain CMF images from the CoMoFoD dataset are used. Among each 100 test images, 50 are authentic, and another 50 are counterfeit

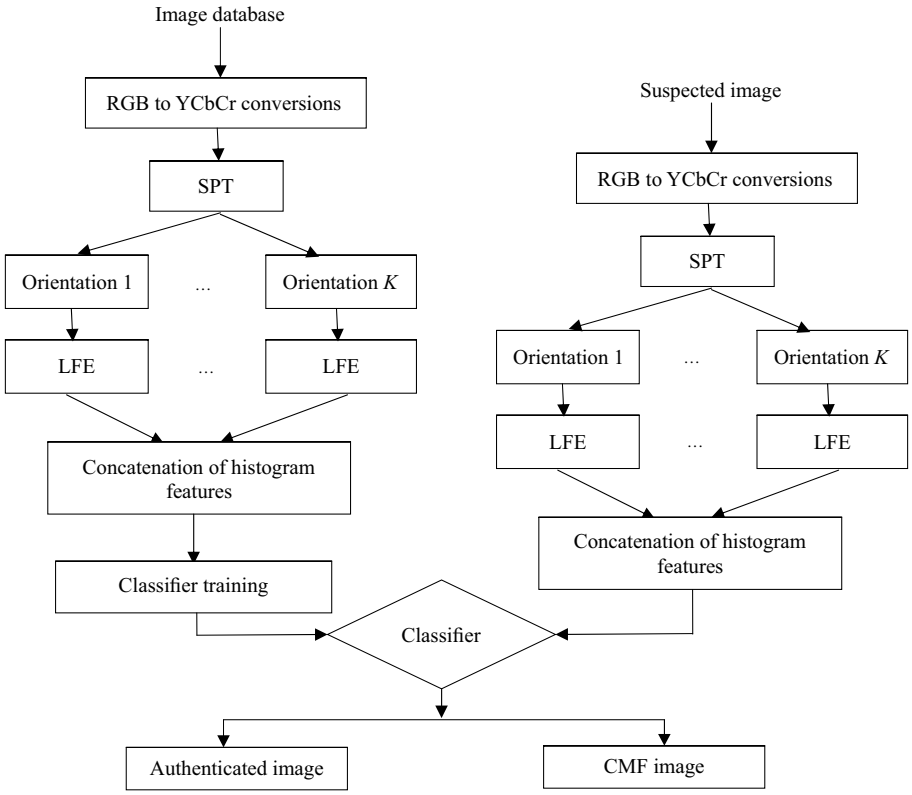

Fig. 6 Flow diagram of the proposed CMFD method.

images. CASIA and CoMoFoD dataset images are tested separately. The proposed CMFD algorithms' performance on a plain CMF image is presented in Table 1. Most images in the CASIA dataset are textured, and CoMoFoD images are not limited to texture, having different variations in images. The performance of the various proposed feature extractors is presented in Table 1, along with the results of some existing methods. The modified SURF and template matching (TM) show better performance than existing methods if the given input image is a plain CMF image. The SURF feature has limitations in identifying keypoints if the image underwent post-processing. Another existing method is considered for comparing the proposed methods, but it is slow and inefficient at forgery detection.

To ensure convincing forgery, various operations are applied to the copied part in the image before it is pasted. These operations include rotation of the copied portion at some angles, scaling and brightness variation of copied portion by a factor, and JPEG compression of the forged image. Along with MICC dataset images, post-processed images are created using authentic MICC images. The performance of the CMFD algorithms on post-processing attack of 3°, 53°, 103°, and 453° angle rotations is presented in Table 2.

One of the post-processing CMF attacks is scaling, which involves scaling the copied part to 95% of its original size or 105% of its original size. The results of scaling the copied portion before being pasted are presented in Table 3. Identifying forgery from a copied part scaled to 105% is often easier than that from a copied part with 95% scaling.

Another post-processing attack is brightness variation. This involves changing the brightness of the copied part to 95% of its original brightness or 105% of its original brightness. The brightness variations of the copied portion before being pasted are presented in Table 4. The LMP feature extractor with OSVM provides better results than the other CMFD algorithms.

JPEG compression attack is explored by subjecting the counterfeit image to various quality factors of JPEG compression. The results of CMFD algorithms on JPEG compressed suspected images are shown in Table 5.

The proposed algorithms are tested under different post-processing attacks with the MICC-F220 dataset. The proposed CMFD algorithms are also evaluated extensively by using a new dataset created with a combination of two standard datasets. A total of four new datasets (mixed dataset) are proposed for an extensive evaluation of the CMFD algorithms. In mixed dataset

Table 1 Performance of CMFD algorithms on plain CMF images.

Histogram method	Classifier	CASIA dataset				CoMoFoD dataset			
		TPR (%)	TNR (%)	FPR (%)	FNR (%)	TPR(%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	98	97	3	2	96	95	5	4
	OSVM	98	96	4	2	97	95	5	3
WLD	ONBC	97	97	3	3	95	96	4	5
	OSVM	97	96	4	3	95	95	5	5
LMP	ONBC	99	98	2	1	98	97	3	2
	OSVM	100	98	2	0	98	98	2	2
LPQ	ONBC	98	99	1	2	96	96	4	4
	OSVM	98	99	1	2	97	96	4	3
GLCM	ONBC	98	99	1	2	95	95	5	5
	OSVM	99	99	1	1	95	96	4	5
LGBPHS	ONBC	99	100	0	1	99	98	2	1
	OSVM	99	100	0	1	100	100	0	0
LTeP	ONBC	100	100	0	0	99	100	0	1
	OSVM	100	99	1	0	99	99	1	1
LBP	SVM	95	96	6	5	95	96	4	5
GLCM	SVM	95	95	5	5	96	95	5	4
Advanced SURF[48]	TM	100	100	0	0	99	100	0	1
PCET[49]	ED-C	98	99	1	2	95	95	5	5

1, the training images (authentic + forged) are from the CASIA dataset, and the testing images (authentic + forged) are from the CoMoFoD dataset. In mixed dataset 2, the training images (authentic + forged) are from the CoMoFoD dataset, and the testing images (authentic + forged) are from the CASIA dataset. The results of the proposed algorithms with mixed datasets 1 and 2 are presented in Table 6.

Apart from mixed datasets 1 and 2, mixed datasets 3 and 4 have more diversity in images. Though the mixed dataset has images from CASIA and CoMoFoD, the training images as well as testing images have a combination of the standard dataset. In the training images of mixed dataset 3, authentic images were obtained from CASIA, and counterfeit images were obtained from CoMoFoD. The testing images of mixed dataset 3 included authentic images from CoMoFoD and counterfeit images from CASIA. Similarly, the training images of mixed dataset 4 included authentic images from CoMoFoD and counterfeit images from CASIA. The testing images of mixed dataset 4 included authentic images from CASIA and counterfeit images from CoMoFoD. The results of the proposed algorithms with mixed datasets 3 and 4 are presented in Table 7.

5 Conclusion

LBP, WLD, and GLCM are computationally simple but sensitive to noise. Furthermore, LBP has the inherent disadvantage of categorizing two different patterns in the same class. Hence, it is not as efficient as other feature extractors. While calculating the LMP, it uses more neighboring pixels with multiple radii than other feature extractors. The LMP feature extractor is effective in post-processing attacks such as scale and brightness variations. However, its performance is poor due to sensitivity to non-monotonic changes in a suspected image. LPQ performs well even when the suspected image has undergone a blurring attack. Experiments on the proposed CMFD algorithms show that LGBPHS and LTeP are the most effective for CMFD among all feature extractors. However, LGBPHS is more computationally expensive than the other feature extractors. The overlapping blocks and iterative similarity matching procedures cause block-based and keypoint-based CMFD algorithms to be more computationally expensive than the proposed CMFD algorithms. The proposed CMFD algorithms also require less memory than the block-based and keypoint-based CMFD algorithms. Despite the promising performance of the proposed CMFD algorithms, the nature of image training and testing plays a vital role. Local descriptors employed in this work can be explored further to implement block-based CMFD and to classify an image as a camera-generated image (authentic) or a computer-generated image (synthetic) in the future.

Table 2 Performance of the proposed CMFD algorithms on post-processing attack of angle rotation.

Histogram method	Classifier	Rotated angle (°)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	3	90	91	9	10
		5	86	85	15	14
		10	81	82	18	19
		45	72	71	29	28
	OSVM	3	90	90	10	10
		5	88	86	14	12
		10	83	81	19	17
		45	75	73	27	25
WLD	ONBC	3	89	92	8	11
		5	86	84	16	14
		10	82	81	19	18
		45	70	69	31	30
	OSVM	3	91	89	11	9
		5	85	86	14	15
		10	82	80	20	18
		45	77	70	30	23
LMP	ONBC	3	92	92	8	18
		5	88	86	14	12
		10	84	85	15	16
		45	76	72	28	24
	OSVM	3	94	94	6	6
		5	89	88	12	11
		10	84	85	15	16
		45	78	78	22	22
LPQ	ONBC	3	93	92	8	7
		5	89	87	13	11
		10	86	86	14	14
		45	78	77	23	22
	OSVM	3	94	94	6	6
		5	89	89	11	11
		10	86	85	15	14
		45	79	78	22	21
GLCM	ONBC	3	94	92	8	6
		5	86	82	18	14
		10	80	80	20	20
		45	75	71	29	25
	OSVM	3	94	92	8	16
		5	85	84	16	15
		10	81	81	19	19
		45	71	72	28	29
LGBPHS	ONBC	3	95	94	6	5
		5	89	88	12	11
		10	85	86	14	15
		45	79	79	21	21
	OSVM	3	94	94	6	6
		5	90	89	11	10
		10	86	86	14	14
		45	81	81	19	19

(to be continued)

Table 2 Performance of the proposed CMFD algorithms on post-processing attack of angle rotation. (continued)

Histogram method	Classifier	Rotated angle (°)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LTeP	ONBC	3	95	95	5	5
		5	89	89	11	11
		10	87	86	14	13
		45	81	81	19	19
	OSVM	3	94	94	6	6
		5	91	89	11	9
		10	85	86	14	15
		45	82	81	19	18
LBP	SVM	3	89	88	12	11
		5	83	81	19	17
		10	78	79	21	22
		45	72	70	30	18
GLCM	SVM	3	93	91	9	7
		5	85	83	17	15
		10	78	79	21	22
		45	73	71	29	27

References

[1] H. Farid, Image forgery detection, *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 16–25, 2009.

[2] M. Kumar and S. Srivastava, Image forgery detection based on physics and pixels: A study, *Australian Journal of Forensic Sciences*, vol. 51, no. 2, pp. 119–134, 2019.

[3] M. Asikuzzaman and M. R. Pickering, An overview of digital video watermarking, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2131–2153, 2018.

[4] K. Jung, A survey of reversible data hiding methods in dual images, *IETE Technical Review*, vol. 33, no. 4, pp. 441–452, 2016.

[5] X. Lin, J. H. Li, S. L. Wang, A. W. C. Liew, F. Cheng, and X. S. Huang, Recent advances in passive digital image security forensics: A brief review, *Engineering*, vol. 4, no. 1, pp. 29–39, 2018.

[6] T. Qazi, K. Hayat, S. U. Khan, S. A. Madani, I. A. Khan, J. Kołodziej, H. Li, W. Lin, K. C. Yow, and C. -Z. Xu, Survey on blind image forgery detection, *IET Image Processing*, vol. 7, no. 7, pp. 660–670, 2013.

[7] M. Kumar and S. Srivastava, Identifying photo forgery using lighting elements, *Indian Journal of Science and Technology*, doi: 10.17485/ijst/2016/v9i48/105748.

[8] M. Kumar, S. Srivastava, and N. Uddin, Forgery detection using multiple light sources for synthetic images, *Australian Journal of Forensic Sciences*, doi: 10.1080/00450618.2017.1356871.

[9] M. Kumar and S. Srivastava, Image authentication by assessing manipulations using illumination, *Multimedia Tools and Applications*, doi: 10.1007/s11042-018-6775-x.

[10] B. Soni, P. K. Das, and D. M. Thounaojam, CMFD: A detailed review of block based and key feature based techniques in image copy-move forgery detection, *IET Image Processing*, vol. 12, no. 2, pp. 167–178, 2018.

[11] S. Teerakanok and T. Uehara, Copy-move forgery detection: A state-of-the-art technical review and analysis, *IEEE Access*, doi: 10.1109/ACCESS.2019.2907316.

[12] S. B. G. T. Babu and C. S. Rao, An optimized technique for copy-move forgery localization using statistical features, *ICT Express*, doi: 10.1016/j.icte.2021.08.016.

[13] D. Cozzolino, D. Gragnaniello, and L. Verdoliva, Image forgery detection based on the fusion of machine learning and block-matching methods, arXiv preprint arXiv: 1311.6934, 2013.

[14] B. Chen, M. Yu, Q. Su, H. J. A. E. Shim, and Y. Shi, Fractional quaternion Zernike moments for robust color image copy-move forgery detection, *IEEE Access*, vol. 6, pp. 56637–56646, 2018.

[15] C. S. Rao and S. B. G. T. Babu, Image authentication using local binary pattern on the low frequency components, *Lecture Notes in Electrical Engineering*, vol. 372, pp. 529–537, 2016.

[16] Y. Li and J. Zhou, Fast and effective image copy-move forgery detection via hierarchical feature point matching, *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1307–1322, 2019.

[17] H. Y. Yang, S. R. Qi, Y. Niu, P. P. Niu, and X. Y. Wang, Copy-move forgery detection based on adaptive keypoints extraction and matching, *Multimedia Tools and Applications*, doi: 10.1007/s11042-019-08169-w.

[18] A. C. Popescu and H. Farid, Exposing digital forgeries by detecting duplicated image regions, https://farid.berkeley.edu/downloads/publications/tr04.pdf, 2022.

[19] K. M. Hosny, H. M. Hamza, and N. A. Lashin, Copy-for-duplication forgery detection in colour images using QPCETMs and sub-image approach, *IET Image Processing*, vol. 13, no. 9, pp. 1437–1446, 2019.

[20] R. Dixit, R. Naskar, and S. Mishra, Blur-invariant copy-move forgery detection technique with improved detection

Table 3 Performance of the proposed CMFD algorithms on post-processing attack of scaling.

Histogram method	Classifier	Scaling (%)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	95	85	87	13	15
		105	88	87	13	12
	OSVM	95	86	87	13	14
		105	89	90	10	11
WLD	ONBC	95	83	85	15	17
		105	86	86	14	14
	OSVM	95	86	87	13	14
		105	88	89	11	12
LMP	ONBC	95	89	90	10	11
		105	92	92	8	8
	OSVM	95	89	89	11	11
		105	94	93	7	6
LPQ	ONBC	95	86	89	11	14
		105	90	89	11	10
	OSVM	95	88	88	12	12
		105	90	91	9	10
GLCM	ONBC	95	82	85	15	18
		105	86	83	17	14
	OSVM	95	86	87	13	14
		105	87	86	14	13
LGBPHS	ONBC	95	87	90	10	13
		105	90	91	9	10
	OSVM	95	87	88	12	13
		105	91	92	8	9
LTeP	ONBC	95	89	89	11	11
		105	91	90	10	9
	OSVM	95	89	89	11	11
		105	93	93	7	7
LBP	SVM	95	87	86	14	13
		105	87	86	14	13
GLCM	SVM	95	86	86	14	14
		105	87	85	15	13

accuracy utilising SWT-SVD, *IET Image Processing*, vol. 11, no. 5, pp. 301–309, 2017.

[21] S. A. Thajeel, A. S. Mahmood, W. R. Humood, and G. Sulong, Detection copy-move forgery in image via quaternion polar harmonic transforms, *KSII Transactions on Internet and Information Systems*, doi: 10.3837/tiis.2019.08.010.

[22] K. B. Meena and V. Tyagi, A copy-move image forgery detection technique based on Gaussian-Hermite moments, *Multimedia Tools and Applications*, vol. 78, no. 23, pp. 33505–33526, 2019.

[23] G. Ramu and S. B. G. T. Babu, Image forgery detection for high resolution images using SIFT and RANSAC algorithm, in *Proc. 2017 2nd International Conference on Communication and Electronics Systems* (*ICCES*), Coimbatore, India, 2017, pp. 850–854.

[24] P. Selvaraj and M. Karuppiah, Enhanced copy-paste forgery detection in digital images using scale-invariant feature transform, *IET Image Processing*, vol. 14, no. 3, pp. 462–471, 2020.

[25] F. M. Al_azrak, Z. F. Elsharkawy, A. S. Elkorany, G. M. E. Banby, M. I. Dessowky, and F. E. A. El-Samie, Copy-move forgery detection based on discrete and SURF transforms, *Wireless Personal Communications*, vol. 110, no. 1, pp. 503–530, 2020.

[26] C. Wang, Z. Zhang, Q. Li, and X. Zhou, An image copy-move forgery detection method based on SURF and PCET, *IEEE Access*, vol. 7, pp. 170032–170047, 2019.

[27] G. Muhammad, M. H. Al-Hammadi, M. Hussain, and G. Bebis, Image forgery detection using steerable pyramid transform and local binary pattern, *Machine Vision and Applications*, vol. 25, no. 4, pp. 985–995, 2014.

[28] S. B. G. T. Babu and C. S. Rao, Texture and steerability based image authentication, in *Proc. 2016 11th International Conference on Industrial and Information Systems* (*ICIIS*), Roorkee, India, 2016, pp. 154–159.

Table 4 Performance of the proposed CMFD algorithms on post-processing attack of brightness variation.

Histogram method	Classifier	Brightness variation (%)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	95	84	85	15	16
		105	89	88	12	11
	OSVM	95	89	88	12	11
		105	91	91	9	9
WLD	ONBC	95	85	85	15	15
		105	89	87	13	11
	OSVM	95	89	89	11	11
		105	90	91	9	10
LMP	ONBC	95	92	91	9	8
		105	94	94	6	6
	OSVM	95	94	93	7	6
		105	96	95	5	4
LPQ	ONBC	95	90	90	10	10
		105	93	92	8	7
	OSVM	95	93	92	8	7
		105	94	93	7	6
GLCM	ONBC	95	84	83	17	16
		105	88	89	11	12
	OSVM	95	89	89	11	11
		105	90	91	9	10
LGBPHS	ONBC	95	88	91	9	12
		105	94	94	6	6
	OSVM	95	94	93	7	6
		105	96	95	5	4
LTeP	ONBC	95	89	91	9	11
		105	92	93	7	8
	OSVM	95	91	93	7	9
		105	95	95	5	5
LBP	SVM	95	82	82	8	18
		105	86	85	15	14
GLCM	SVM	95	81	80	20	19
		105	84	84	16	16

[29] G. Muhammad, M. H. Al-Hammadi, M. Hussain, A. M. Mirza, and G. Bebis, Copy move image forgery detection method using steerable pyramid transform and texture descriptor, in *Proc. IEEE EuroCon 2013*, Zagreb, Croatia, 2013, pp. 1586–1592.

[30] T. Ojala, M. Pietikäinen, and T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[31] T. H. Rassem and B. E. Khoo, Completed local ternary pattern for rotation invariant texture classification, *The Scientific World Journal*, doi: 10.1155/2014/373254.

[32] X. Tan and B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[33] Y. Xiao, Z. Cao, L. Wang, and T. Li, Local phase quantization plus: A principled method for embedding local phase quantization into Fisher vector for blurred image recognition, *Information Sciences*, vol. 420, pp. 77–95, 2017.

[34] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition, in *Proc. Tenth IEEE International Conference on Computer Vision volume 1*, Beijing, China, 2005, pp. 786–791.

[35] A. R. Rivera, J. R. Castillo, and O. Chae, Local directional texture pattern image descriptor, *Pattern Recognition Letters*, vol. 51, pp. 94–100, 2015.

[36] M. S. Islam and S. Auwatanamo, Facial expression recognition using local arc pattern, *Trends in Applied Sciences Research*, vol. 9, no. 2, pp. 113–120, 2014.

[37] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikäinen, X. Chen, and W. Gao, WLD: A robust local image descriptor, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1705–1720, 2010.

Table 5 Performance of the proposed CMFD algorithms on post-processing attack of JPEG compression.

Histogram method	Classifier	JPEG compression quality factor (%)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	90	92	90	10	8
		80	88	88	12	12
		70	84	86	14	16
		60	80	80	20	20
	OSVM	90	94	92	8	6
		80	90	90	10	10
		70	86	88	12	14
		60	82	80	20	18
WLD	ONBC	90	88	90	10	12
		80	86	86	14	14
		70	84	84	16	16
		60	80	78	22	20
	OSVM	90	92	92	8	8
		80	90	90	10	10
		70	88	88	12	12
		60	82	82	18	18
LMP	ONBC	90	94	94	6	6
		80	90	90	10	10
		70	86	84	16	14
		60	80	82	12	20
	OSVM	90	94	94	6	6
		80	91	90	10	9
		70	90	88	12	10
		60	82	80	20	18
LPQ	ONBC	90	98	98	2	2
		80	98	96	4	2
		70	90	92	8	10
		60	85	86	14	15
	OSVM	90	98	98	2	2
		80	98	98	2	2
		70	94	94	6	6
		60	88	86	14	12
GLCM	ONBC	90	90	90	10	10
		80	86	88	12	14
		70	84	86	14	16
		60	78	78	22	12
	OSVM	90	92	92	8	8
		80	90	90	10	10
		70	88	88	12	12
		60	82	82	18	18
LGBPHS	ONBC	90	96	94	6	4
		80	94	94	6	6
		70	90	90	10	10
		60	82	84	16	18
	OSVM	90	96	94	6	4
		80	92	94	6	8
		70	90	92	8	10
		60	84	86	14	16

(to be continued)

Table 5 Performance of the proposed CMFD algorithms on post-processing attack of JPEG compression. (continued)

Histogram method	Classifier	JPEG compression quality factor (%)	Performance metric			
			TPR (%)	TNR (%)	FPR (%)	FNR (%)
LTeP	ONBC	90	95	94	6	5
		80	94	92	8	6
		70	88	86	14	12
		60	82	82	18	18
	OSVM	90	94	94	6	6
		80	92	92	8	8
		70	90	91	9	10
		60	84	84	16	16
LBP	SVM	90	88	88	12	12
		80	86	88	12	14
		70	84	86	14	16
		60	78	78	22	22
GLCM	SVM	90	88	88	12	12
		80	86	86	14	14
		70	84	82	18	16
		60	78	76	24	22

Table 6 Performance of CMFD algorithms with mixed datasets 1 and 2.

Histogram method	Classifier	Mixed dataset 1				Mixed dataset 2			
		TPR (%)	TNR (%)	FPR (%)	FNR (%)	TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	80	82	18	20	78	80	20	22
	OSVM	84	84	16	16	80	82	18	20
WLD	ONBC	80	80	20	20	80	78	22	20
	OSVM	84	82	18	16	82	82	18	18
LMP	ONBC	82	82	18	18	80	82	18	20
	OSVM	86	86	14	14	84	84	16	16
LPQ	ONBC	86	84	16	14	86	86	14	14
	OSVM	88	86	14	12	88	86	14	12
GLCM	ONBC	80	80	20	20	78	80	20	22
	OSVM	82	82	18	18	80	82	18	20
LGBPHS	ONBC	90	88	12	10	90	90	10	10
	OSVM	92	90	10	8	92	92	8	8
LTeP	ONBC	88	88	12	12	88	90	10	12
	OSVM	90	88	12	10	90	88	12	10
LBP	SVM	78	76	24	22	76	76	14	24
GLCM	SVM	76	74	26	24	74	74	16	26

[38] G. Suresh and C. S. Rao, Copy move forgery detection using GLCM based statistical features, *International Journal on Cybernetics & Informatics*, vol. 5, no. 4, pp. 165–171, 2016.

[39] E. P. Simoncelli and W. T. Freeman, The steerable pyramid: A flexible architecture for multi-scale derivative computation, in *Proc. IEEE International Conference on Image Processing*, Washington, DC, USA, 1995, pp. 444–447.

[40] H. Zhu, L. Yu, Y. Zhang, L. Cheng, Z. Zhu, J. Song, J. Zhang, B. Luo, and K. Yang, Optimized support vector machine assisted BOTDA for temperature extraction with accuracy enhancement, *IEEE Photonics Journal*, vol. 12, no. 1, pp. 1–14, 2020.

[41] A. Messac, *Optimization in Practice with MATLAB®*. Cambridge, UK: Cambridge University Press, 2015.

[42] L. Kuncheva and Z. Hoare, Error-dependency relationships for the Naïve Bayes classifier with binary features, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 735–740, 2008.

[43] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York, NY, USA: Springer, 2014.

[44] CASIA: Image tampering detection evaluation database, http://forensics.idealtest.org, 2010.

[45] CoMoFoD - Image database for copy-move forgery detection, http://www.vcl.fer.hr/comofod/, 2019.

[46] Image and Communication Laboratory, MICC_copy-move

Table 7 Performance of CMFD algorithms with mixed datasets 3 and 4.

Histogram method	Classifier	Mixed dataset 3				Mixed dataset 4			
		TPR (%)	TNR (%)	FPR (%)	FNR (%)	TPR (%)	TNR (%)	FPR (%)	FNR (%)
LBP	ONBC	76	76	24	24	74	74	26	26
	OSVM	82	80	20	18	82	82	18	18
WLD	ONBC	76	78	22	24	76	74	26	24
	OSVM	78	78	22	22	76	76	24	24
LMP	ONBC	80	84	16	20	80	82	18	20
	OSVM	84	82	18	16	82	80	20	18
LPQ	ONBC	78	80	20	22	78	78	22	22
	OSVM	82	82	18	18	80	82	18	20
GLCM	ONBC	76	76	24	24	74	76	24	26
	OSVM	78	78	22	22	74	76	24	26
LGBPHS	ONBC	82	84	16	18	82	82	18	18
	OSVM	86	86	14	14	86	84	16	14
LTeP	ONBC	84	86	14	16	86	86	14	14
	OSVM	86	88	12	14	86	88	12	14
LBP	SVM	74	74	26	26	72	72	28	28
GLCM	SVM	74	72	28	26	74	72	28	26

forgery detection and localization, http://lci.micc.unifi.it/labd/2015/01/copy-move-forgery-detection-and-localization/, 2019.

[47] O. M. Al-Qershi and B. E. Khoo, Evaluation of copy-move forgery detection: Datasets and evaluation metrics, *Multimedia Tools and Applications*, vol. 77, no. 24, pp. 31807–31833, 2018.

[48] A. Rani, A. Jain, and M. Kumar, Identification of copy-move and splicing based forgeries using advanced SURF and revised template matching, *Multimedia Tools and Applications*, vol. 80, no. 16, pp. 23877–23898, 2021.

[49] K. M. Hosny, H. M. Hamza, and N. A. Lashin, Copy-move forgery detection of duplicated objects using accurate PCET moments and morphological operators, *Imaging Science Journal*, doi: 10.1080/13682199.2018.1461345.

Ch Srinivasa Rao is working as a professor in the Department of ECE, JNTUK UCE, Vizianagaram, Andhra Pradesh, India. He is currently deputed as a principal to JNTUK, University College of Engineering, Narasaraopet, AP, India. He received the PhD degree in digital image processing area from the University College of Engineering, JNTUK, Kakinada, AP, India. He has 28 years of teaching and research experience. He published 65 research papers in reputed international journals and conferences. He is a fellow of IETE and a member of CSI. His research interests include content-based image and video retrieval, medical image processing, video watermarking, and image forensics. He also serves as a reviewer for many reputed international journals.

S. B. G. Tilak Babu is currently working as an assistant professor of the Department of ECE at Aditya Engineering College, Surampalem, AP, India. He is pursuing the PhD degree in digital image processing area at the University College of Engineering, JNTU Kakinada, Kakinada, AP, India. He has four years of teaching and research experience. He published 20 research papers in reputed international journals and conferences. His research interests include digital image processing, soft computing techniques, and image forensics.