A Survey on Passive Image Copy-Move Forgery Detection

Zhi Zhang*, Chengyou Wang*, and Xiao Zhou*

Abstract
With the rapid development of the science and technology, it has been becoming more and more convenient to obtain abundant information via the diverse multimedia medium. However, the contents of the multimedia are easily altered with different editing software, and the authenticity and the integrity of multimedia content are under threat. Forensics technology is developed to solve this problem. We focus on reviewing the blind image forensics technologies for copy-move forgery in this survey. Copy-move forgery is one of the most common manners to manipulate images that usually obscure the objects by flat regions or append the objects within the same image. In this paper, two classical models of copy-move forgery are reviewed, and two frameworks of copy-move forgery detection (CMFD) methods are summarized. Then, massive CMFD methods are mainly divided into two types to retrospect the development process of CMFD technologies, including block-based and keypoint-based. Besides, the performance evaluation criterions and the datasets created for evaluating the performance of CMFD methods are also collected in this review. At last, future research directions and conclusions are given to provide beneficial advice for researchers in this field.

Keywords
Copy-Move Forgery Detection (CMFD), Image Forensics, Image Tamper Detection, Passive Forgery Detection

1. Introduction
Nowadays, the digital image is still one of the most significant carriers to help people obtain large amounts of information. It is said that an image is worth innumerable words, which describes the fact that an image contains tremendous information. However, with the increasing use of modern sophisticated image editing software, such as Adobe Photoshop and GNU image manipulation program (GIMP), digital images are easily manipulated and altered without leaving any clues, and the credibility of image content cannot be identified even by trained observers. The tampered images altered maliciously by promulgator may pose the severe social problems, especially in medical diagnosis, court sentence, patent infringements, and insurance claims. One of the most famous events of the forged images is Iranian missile test in July 2008 [1] that is published on the front-pages of a few major news websites, including The New York Times, The Chicago Tribune, The Los Angeles Times, etc. The tampered photo was obtained from the web site of Iran’s Sepah News, as shown in Fig. 1. A genuine Iranian missile photo is exhibited in Fig. 1(a); while Fig. 1(b) and (c) show the published version of Iranian missile situation, in which the third missile from the left was digitally appended to the original
photo to cover up the fact that it did not fire. A day later, the Associated Press news agency published the original photo (Fig. 1(a)) that further proved that published picture was synthetic. The similar event triggered by the counterfeit image is occurring every day. To validate the creditability of digital image content, image forensics technology is urgent to develop to avoid a huge loss of social benefits.

![Fig. 1. Actual event of Iranian missile: (a) genuine Iranian missile photo, (b) forged Iranian missile photo published on BBC NEWS, and (c) forged Iranian missile photo with marked region-duplication.](image)

Since the advent of synthetic images, many researchers have been devoting to the field of image forensics aim at different image forgery manners, such as copy-move [2], splicing [3], resampling [4], filtering [5], and double JPEG (joint photographic experts group) compression [6]. The classification of images forensics technologies is given in [7], as shown in Fig. 2. Image forensics technologies are divided into two categories: active forensics and passive forensics. Active forensics verifies the integrity of auxiliary information to decide whether the image has been tampered with, for instance, digital signature [8] and digital watermarking [9]. However, this type of technology requires special software or hardware to insert the authentication information into images or extract authentication information from images before being distributed. Passive forensics verifies the authenticity of the image by analyzing its contents and structure. In this survey, we focus on blind image forensics technologies for copy-move. Copy-move is one of the most common manners to alter the images by manipulators, which is usually used to obscure the objects by flat regions or append the objects within a same image. Copy-move forgery detection (CMFD) technologies are mainly divided into two classes, block-based methods and keypoint-based methods, which will be discussed in detail in the later subsequent sections.

![Fig. 2. Image forensics technologies classification.](image)
The rest of this survey is organized as follows. Section 2 starts with a brief review of two models of copy-move forgery, and then two frameworks of CMFD methods are described. Block-based CMFD technologies are presented in Section 3, and Section 4 shows keypoint-based CMFD technologies. In Section 5, datasets consisting of forged images with copy-move and performance evaluation criterions are collected to evaluate the performance of CMFD technologies. Section 6 gives finally the future directions of CMFD and conclusion.

2. Models of Copy-Move Forgery and Frameworks of CMFD

In this section, two models of copy-move forgery are reviewed. Two frameworks accordant with the diverse CMFD technologies are also presented, and most of CMFD schemes are based on these models.

2.1 Models of Copy-Move Forgery

In [10], after analyzing 100 natural images, the authors found that it is impossible that a single image has two similar areas larger than 0.85% of the image area. The goal is looking for two similar large areas in suspicious image, as shown in Fig. 3(a). They made a deduction as below:

Given an image $I$, the forged image $I'$, must subordinate to: $\exists$ areas $D_1$ and $D_2$ are subsets of $D$ and a shift vector $d = (dx, dy)$ , (they assumed that $|D_1| = |D_2| > |D| * 0.85\%$ and $|d| > L$ ), $I'(x, y) = I(x, y)$ if $(x, y) \notin D_2$ and $I'(x, y) = I(x - dx, y - dy)$ if $(x, y) \in D_2$ , where $D_1$ is the source region and $D_2$ is the pasted region, $D_2 = D_1 + d$. Nevertheless, Luo’s model cannot describe the forgery way that a copied region is pasted to two or more places, and the copied region is rotated before being pasted. Plain copy-move forgery is shown in Fig. 3(a).

![Fig. 3. Two models of copy-move forgery: (a) Luo’s model and (b) Liu’s model.](image)

To remedy the defect of Luo’s model, Liu et al. [11] presented a more comprehensive copy-move forgery model, as shown in Fig. 3(b). They assumed that the shift vector threshold is $V_T = [V_{\text{dx}}, V_{\text{dy}}]$ , and copied region threshold (the ratio of the copy area and the whole image area) is $A_t$ . An image $I$ is forged to $I'$ via copy-move manipulation, if

1) The copied region $C_i, i \in \{1,2,\cdots,n\}$ is single-connected and has no hole inside, and its area is greater than $A_t a(I)$ , where $a(I)$ denotes the area of $I$.

2) Supposing the pasted region of copied region $C_i$ is $M_i$ , there might be many duplicated region
pairs \( \{ C_i \| M_i \| M_i, \ldots, C_n \| M_n \} \), \( C_i, M_i \in I' \), which satisfy \( C_i \neq C_j, \forall i \neq j, i, j \in \{1, 2, \ldots, n\} \) and \( C_i \cap C_j = \emptyset \). For any pair \( C_i \| M_i \), defining the origin of the reference system as the center of rotation, the copy-move forgery can be considered as translation after rotation, described by

\[
\begin{align*}
\forall (x, y) \in C_i, f(x, y) &= f'(x', y'), \\
x' &= x \cos \theta - y \sin \theta + \Delta_x, \\
y' &= x \sin \theta + y \cos \theta + \Delta_y, \\
\sqrt{\Delta_x^2 + \Delta_y^2} &\geq |V|, \\
A(C_i) &> A_t \cdot a(I),
\end{align*}
\]

where \( f \) denotes the pixel value at the position \((x, y)\); \( \Delta_x \) and \( \Delta_y \) is the shift distance along \( x \) and \( y \) axis, respectively; and \( \theta \) is the rotation angle. However, the pasted region may be altered by other operations, like scaling that changes one-to-one mapping. Therefore, a more generalized copy-move model needs to be proposed. A good CMFD scheme can detect the duplicated regions even if the pasted region is distorted by blurring, rotation, noise contamination, scaling, or JPEG compression.

2.2 Frameworks of CMFD

In this subsection, two frameworks of CMFD methods are presented, and most of CMFD schemes adhere to the two frameworks.

1) Pre-processing: The suspicious image is processed by a series of operations. For example, adopt Wiener filter [12] or dyadic wavelet transform (DyWT) [13] for de-noise; convert RGB (red, green, blue) color space into grayscale space [14,15], YCrCb color space [16], HSV (hue, saturation, and value) space [17], or color local binary pattern (LBP) space [18]; perform discrete wavelet transform (DWT) [19] or Gaussian pyramid decomposition [11] to obtain the dimension reduction presentation of the image.

2) Feature extraction: Image segmentation is used for feature extraction in block-based CMFD methods. The image is divided into overlapping square image blocks [20], non-overlapping square image blocks [21], or overlapping circle image blocks [22]. Besides, simple linear iterative clustering (SLIC) method [23] is also used for image segmentation [24]. A great deal of features are extracted for CMFD, such as discrete cosine transform (DCT) [14], Fourier-Mellin transform (FMT) [25], 2D-Fourier transform [26], polar harmonic transform (PHT) [27], singular value decomposition (SVD) [28], LBP [22], Zernike moment [29], Hu moment [11], scale-invariant feature transform (SIFT) [15], speeded up robust features (SURF) [30], Harris corner features [31], DAISY [32], etc.

3) Feature matching: Feature matching is the procedure of finding similar feature vectors. To narrow the range of similar feature vectors, sorting algorithm causes similar features to be adjacent, such as lexicographic sorting [19] and radix sorting [33]. Besides, k-d tree [18] or locality-sensitive hashing (LSH) [34] speeds up the process of finding similar feature vectors a lot. In addition, there are many methods to evaluate the similarity between the feature vectors, such as Euclidean distance and Manhattan distance, which are corresponding with Eq. (2) and Eq. (3), respectively.
\[ d_{\text{Euclidean}} = \sqrt{\sum_{i=1}^{n} (v_1(i) - v_2(i))^2} \],

\[ d_{\text{Manhattan}} = \sum_{i=1}^{n} |v_1(i) - v_2(i)|, \]

where \( v_1 \) and \( v_2 \) are \( n \)-dimensional feature vectors.

4) Localization and post-processing: If the regions determined by the process of feature matching are shown in the map, there will be many isolated points, and morphologic operations [35], filtering [33] or random sample consensus (RANSAC) algorithm [36] is usually used to refine detected regions.

Another model is based on machine learning, as shown in Fig. 4(b). The classifier is trained by trained image set with labels and the trained classifier will determine whether the test image has been tampered or not, like support vector machine (SVM) classifier [16]. However, CMFD methods based machine learning only decide whether the test image has been forged or not. It is still a challenging problem for this type of methods to locate tampered region.

![Fig. 4. Two frameworks of CMFD methods: (a) framework of block-based CMFD and keypoint-based CMFD and (b) framework of CMFD based on machine learning. Adapted from G. K. Birajdar and V. H. Mankar. Digital image forgery detection using passive techniques: a survey. Digital Investigation 2013;10(3):226-245, with the permission of Elsevier [37].](image)

3. Block-Based CMFD Methods

Diverse CMFD methods are briefly described in this section, including DCT-based, wavelet transform-based, PHT-based, LBP-based, Zernike-based, SVM-based, etc. A table is given to compare the performance of CMFD methods from various aspects at the end of this section.

3.1 DCT Based Algorithms

Fridrich et al. [14] first proposed the DCT-based method for CMFD. The image is divided into fixed-size overlapping image blocks at raster-scan, and DCT is performed on each block. The quantization feature vector is obtained by performing zigzag scanning on the quantized DCT coefficient matrix. The feature matrix is lexicographically ordered and Euclidean distance is used for similarity judgment.
However, this method has high computation cost. To reduce computational complexity, Huang et al. [38] truncated the feature vector by using a constant to reduce the dimensionality of the feature and presented a scheme to judge similarity between feature vectors. Mahmood et al. [39] used Gaussian radial basis function (RBF) and kernel principal component analysis (KPCA) to reduce the dimensionality of the feature vector that ameliorated the efficiency in the feature matching process. Cao et al. [40] divided the inscribed circle of a square image block into four non-overlapping parts and extracted mean of each part coefficients as the feature to detect duplicated regions. Fadl and Semary [41] divided feature vectors into several groups by using fast k-means algorithms and searched duplication-region in each group. In [42], the authors divided the image into smooth and complex by using edge detection information, and if the image is smooth (complex), it is divided into big (small) blocks. DCT coefficients were set as 0 or 1 according to the rules in [43] for CMFD. Alkawaz et al. [44] studied the effects of different block size on the performance of CMFD method. In [45], package clustering algorithm is used to divide the DCT feature vectors and coordinates into different packages, and then find similar feature vectors in each package. Zhao and Guo [46] presented a scheme combining DCT and SVD for CMFD. Doyoddorj and Rhee [47] used quantized DCT coefficients obtained by performing DCT in Radon space of each image block and detected copy-move regions. Ustubioglu et al. [48] proposed a CMFD algorithm based on LBP and DCT.

3.2 Wavelet Transform Based Algorithms

In this subsection, different wavelet transform algorithms are collected. The image is decomposed into four parts by DWT, including approximate component sub-band LL, horizontal detail sub-band LH, vertical detail sub-band HL, and diagonal detail sub-band HH. In [19], LL is divided into overlapping image blocks and singular value vector is the feature vector obtained by performing SVD on each block. Kashyap and Joshi [49] extracted blur moment invariants from each block and performed PCA on blur moment invariants matrix to reduce the dimensionality. DyWT [50] is shift invariant and captures the structure in a better way than DWT. In [51,52], authors presented a CMFD scheme by using the fact that copied region and pasted region should exhibit similarity between them in LL, while copied region and pasted region in HH should exhibit high dissimilarity between them in noise pattern. A similar approach in [53] utilized singular value vectors obtained from overlapping blocks in LL and HH to detect duplicated regions. However, LL and HH are obtained by stationary wavelet transform (SWT) [54].

3.3 Other Transforms Based Algorithms

In [55], the authors offered a CMFD scheme that used image block feature mapped to log-polar coordinates and phase correlation to search duplicated region. Bravo-Solorio and Nandi [56] computed 1D descriptor invariant to rotation and reflection by summing a log-map along the log-radius axis to make the localization of duplication-region more precise. In [57], they proved the effectiveness of their scheme by various experiments and comparisons with other CMFD methods. However, interpolation from the Cartesian coordinates to a log-polar gridding reduces precision and results in considerable errors in low image resolution or small block-size. By using rotation and scale invariant features, Wu et al. [58] proposed a CMFD scheme to detect region-duplication in a forged image by using log-polar fast
Fourier transform (LPFFT), while Park et al. [59] presented a scheme to detect region-duplication by utilizing the feature extracted from the up-sampled log-polar Fourier (ULPF) descriptor. By introducing an adaptive phase correlation method in the log-polar coordinate system and utilizing the information extracted from the band limitation, Yuan et al. [60] presented a robust CMFD method which can handle with large scaling operation.

Yap et al. [61] proposed PHTs-based method against rotation, which included polar sine transform (PST), polar cosine transform (PCT), and polar complex exponential transform (PCET). PCET has relatively better performance than the other two transforms. In [27,62], the authors divided the image into overlapping circular blocks, and PST was performed on each circular block to extract feature. After filtering and morphological processing, the duplicated regions were found. Li [34] extracted PST coefficients of each block as the feature and searched duplicated region by approximate nearest searching and LSH to achieve CMFD. Ganty and Kousalya [63] realized spectral-hashing-based PCT image CMFD algorithm. In [64], the authors proposed a PCET-based CMFD scheme in which LSH was used for identifying the potential similar image blocks. Bi et al. [65] extracted color texture descriptor and invariant moment descriptor calculated from the PCET moments to solve the problem of searching duplicated regions. Wo et al. [66] presented a CMFD method based on multi-radius PCET that can detect the pasted region with large-scale scaling and rotation. In [67], the authors proposed an efficient discrete Radon polar complex exponential transform (DRPCET)-based scheme for extracting the scaling and rotational invariant features for CMFD. It is worth mentioning that they introduced an auxiliary circular template to construct invariant feature, as shown in Fig. 5. Zhong et al. [68] extracted the discrete radial harmonic Fourier moments (DRHFMs) from each circular block with the help of the circular template (Fig. 5).

![Fig. 5. A circular template in Cartesian space. Adapted from J. Zhong et al. A new block-based method for copy move forgery detection under image geometric transforms. Multimedia Tools and Applications 2017;76(13):14887-14903, with the permission of Springer [68].](image)

In [25], the authors proposed FMT-based CMFD scheme in which counting bloom filters, rather than lexicographic sorting, were used to save computational time. Li and Yu [69] improved Bayram’s scheme in which distance vectors are clustered by a vector erosion filter that is robust to rotation and scaling.
After introducing analytical Fourier-Mellin transform (AFMT), Zhong and Gan [70] proposed discrete analytical Fourier-Mellin transform (DAFMT-based CMFD scheme. However, as pointed in [67], the defect of AFMT is too complicated, especially the construction of its invariant moment.

Ketenci and Ulutas [26] applied 2D-Fourier transform on each overlapping square block to extract feature for achieving CMFD algorithm. After performing the Fourier transform of the polar expansion on the overlapping windows pair and implementing an adaptive band limitation to construct a correlation matrix, Shao et al. [71] offered a CMFD scheme by estimating the rotation angle of the forged region and using search algorithm to locate the duplicated regions. In [72], the authors utilized four features extracted from Fourier transform coefficients of each circular block to achieve CMFD algorithm. After extracting electromagnetism-like (EMag) mechanism descriptor from each non-overlapping block, Dadkhah et al. [73] applied discrete Fourier transform (DFT) to EMag features to achieve CMFD. By using city block filter, horizontal filter, vertical filter, and frequency filter, Huang et al. [74] offered a threshold-free CMFD scheme by combining the features including fast Fourier transform (FFT), SVD, and PCA.

3.4 LBP and Moment Invariant Based Algorithms

Li et al. [22] divided the image into overlapping circular blocks and extracted features using rotation invariant uniform LBP to detect duplicated regions. Davarzani et al. [75] presented an efficient scheme for CMFD using multi-resolution LBP (MLBP), in which k-d tree is used to save time and RANSAC is used to remove the possible false matches. In [76], the object is detected by normalized cut segmentation, and then, with the help of Hessian method, local interest points are localized. Duplication-region is found by using center-symmetric LBP (CSLBP). Yang et al. [77] also used uniform LBP to detect duplicated region. What is different with [22] is that the authors used a shift-vector counter instead of the block matching. Trajic et al. [78] combined cellular automata (CA) and LBP to extract feature vectors for CMFD. In [79], the authors proposed a CMFD method using binary gradient contours (BGC), and they proved the performance of their schemes is superior to many LBP-based methods.

Ryu et al. [29] presented a CMFD algorithm by using the magnitude of Zernike moment invariant against rotation. In [80], the authors combined LSH and RANSAC to improve the accuracy and efficiency of CMFD scheme based on Zernike moment. Al-Qershi and Khoo [81] adopted a grouping method [82] for block matching to improve detection accuracy. Thuong et al. [83] extracted foreground of the image by morphological technology and performed the wavelet transform to the foreground to extract approximate component. Zernike moment is used for CMFD in [83]. Mahmoud and Abu-Alrukab [84] proposed pseudo-Zernike moment (PZM)-based scheme for CMFD and improved Zernike moment based method.

Mahdian and Saic [85] proposed a CMFD method based on blur invariant moment constructed by applying the algorithm [86]. Du et al. [87] combined the 1D moment, the 2D moment, and the Markov feature to present a CMFD algorithm based on multiple features, where 1D moment is the feature of 1D histogram and 2D moment is the feature of 2D histogram in the horizontal and vertical direction. Imamoglu et al. [88] extracted Krawtchouk moment to detect the duplicated region in the forged image. Liu et al. [11] extracted Hu moment for CMFD and Kushol et al. [89] combined Hu moment and Lab color space-based feature for CMFD.
3.5 Other Algorithms

Popescu and Farid [90] utilized PCA to reduce the dimensionality of block features and proposed a scheme to detect duplicated regions in forged images. Kakar and Sudha [91] extracted the features by using the MPEG-7 image signature tools and presented a novel technology for CMFD. Malviya and Ladhake [92] employed auto color correlogram (ACC), a feature used in image retrieval, to obtain feature vector and detected duplicated regions successfully. In [93], the authors presented a CMFD method against scaling operation. Vladimirovich and Valerievich [94] proposed a plain CMFD algorithm using structure pattern and 2D Rabin-Karp rolling hash, which achieves zero false negative error and fast execution speed for the images with high resolution. On the basis of the method in [94], Kuznetsov and Myasnikov [95] presented a CMFD scheme by using a hash value calculation in a sliding window mode. Kashyap et al. [96] combined SVD and cuckoo search algorithm that can automatically generate suitable parameter value for each image.

PatchMatch [97] is a fast approximate nearest-neighbor search algorithm for block matching [98-100]. In [98], the authors modified the basic PatchMatch algorithm and proved its efficiency by using CMFD based on Zernike moment and CMFD based on RGB value, respectively. They presented two detectors in [99] that can detect forged regions in the spliced image and copy-move image, respectively. By utilizing the invariant features and a suitably modified version of PatchMatch, Cozzolino et al. [100] achieved a CMFD scheme that has a good robustness to various types of geometrical distortion.

In [16], the authors extracted multi-resolution Weber law descriptors (WLD) as the feature and trained a model by SVM in which the accuracy of their method can reach up to 91%. After training models by SVM with MLBP and multi-resolution WLD, respectively, Hussain et al. [101] found that multi-resolution WLD performs better than multi-resolution LBP in detecting splicing and copy-move forgeries. In [102], steerable pyramid transform (SPM) is performed on the chrominance channels Cr and Cb to obtain some multi-scale and multi-oriented sub-bands. The feature vector is produced by concatenating the histograms from each sub-band and SVM uses the feature vectors to classify images into forged or authentic. Rao and Ni [103] presented a CMFD scheme based on the deep learning technology including SVM and convolutional neural network (CNN). A 10-layer CNN is used to automatically learn hierarchical representations from the RGB images. Dense features extracted from the test image are obtained by using the pre-trained CNN, and a feature fusion technology is designed to obtain the discriminative features for SVM classification.

State-of-the-art block-based CMFD algorithms and some of the classical schemes are described in Table 1. Table 1 describes the methods from several aspects including pre-processing, feature extraction, method for searching similar blocks, post-processing, performance, and dataset.

In Table 1, several aspects need to be explained. The collected data are the basis of the corresponding literature, and the detail information can be found in literature. In ‘Feature extraction’ column, GLCM is the abbreviation of gray-level co-occurrence matrix; CLD is the abbreviation of color layout descriptors; CHT is the abbreviation of circular harmonic transforms; and feature1 is three averages of the red, blue, and green color of the pixels and entropy. In ‘Performance’, single/multiple means the method can detect the number of the forged region. AWGN means additive white Gaussian noise; [max_value,min_value] means the range of relevant processing with maximum value and minimum value; and min_value:step:max_value means the value ranges from maximum value and minimum value with the step. The parameters in ‘Performance’ need to be distinguished by readers to read.
relevant literature because of the difference of researchers’ comprehension. In ‘Dataset’, datasets for CMFD will be listed in Section 5.2, and basic datasets used by researchers to create their own datasets for CMFD are also listed in this survey, such as UCID [104], National Geographic [105], ImageNet [106], Kodak [107], DOCR [108], PIMPRCG [109], USC-SIPI [110], KSU [51], and Caltech-256 [111].

Table 1. Block-based CMFD methods comparison

<table>
<thead>
<tr>
<th>Ref</th>
<th>Year</th>
<th>Pre-processing</th>
<th>Feature extraction</th>
<th>Method for searching similar blocks</th>
<th>Post-processing</th>
<th>Performance</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liu et al.</td>
<td>2011</td>
<td>Gaussian pyramid decomposition; Circular block</td>
<td>Hu moment</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Multiple; AWGN (15:10:15); JPEG compression (45:20:85); Rotation (-20, 12, 90); Gaussian blurring (5, 1:1:3); Flipping (horizontal)</td>
<td>Internet</td>
</tr>
<tr>
<td>Zhu et al.</td>
<td>2017</td>
<td>Convert RGB into color LBP; Square block</td>
<td>GLCM</td>
<td>K-d tree; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Single; Gaussian noise (40,80); Gaussian blurring ((3,1), (5,2)); JPEG compression (60, 90)</td>
<td>GRIP</td>
</tr>
<tr>
<td>Ustubioglu et al.</td>
<td>2016</td>
<td>Square block</td>
<td>Color moments ; CLD</td>
<td>Clustering; Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Multiple; AWGN ((5, 0.5:5.3:0), (3, 0.5:5.1:1.0); Gaussian blurring (15:5:35); JPEG compression (40:10:90)</td>
<td>CoMoFD</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2013</td>
<td>Convert RGB into gray; Low-pass filtering; Circular block</td>
<td>LBP</td>
<td>Sorting; Euclidean distance</td>
<td>Filtering; Morphologic operations</td>
<td>Single; Rotation (90:90:270); Flipping (horizontal, vertical); JPEG compression (50:10:90); AWGN (15:5:35); Gaussian blurring (5, 1:1:5)</td>
<td>UCID; Internet</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2012</td>
<td>Convert RGB into gray; Circular block</td>
<td>PHT</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Single; Rotation (15°, 30°, 90°); AWGN ((15, 40); JPEG compression (&gt;50); Mixture operations</td>
<td>Internet</td>
</tr>
<tr>
<td>Kang and Wei</td>
<td>2018</td>
<td>Square block</td>
<td>SVD</td>
<td>Sorting; Euclidean distance</td>
<td>Single; JPEG compression (50:10:100); AWGN (25:55)</td>
<td>Internet</td>
<td></td>
</tr>
<tr>
<td>Ryu et al.</td>
<td>2010</td>
<td>Square block</td>
<td>Zernike moment</td>
<td>Sorting; Euclidean distance</td>
<td>Single; Rotation (0°:10°:90°); JPEG compression (40:10:100); AWGN (0.001:0.002:0.009); Gaussian blurring (0.5:0.5:3.0); Mixture operations</td>
<td>Internet; National Geographic</td>
<td></td>
</tr>
<tr>
<td>Li [34]</td>
<td>2013</td>
<td>Circular block</td>
<td>PCT</td>
<td>LSH</td>
<td>Morphologic operations</td>
<td>Single; Gaussian blurring (0.5:0.5:3.0); Gaussian noise (0.001:0.002:0.009); JPEG compression (20:10:90); Rotation (10°:10°:90°)</td>
<td>ImageNet; Kodak; OCR</td>
</tr>
<tr>
<td>Huang et al.</td>
<td>2011</td>
<td>Convert RGB into gray; Square block</td>
<td>DCT</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Single; Gaussian blurring ((3, 0.5:5.1:0), (3, 0.5:5.1:0)); AWGN (1, 2, 4); JPEG compression (50, 70:10:90)</td>
<td>DVMM</td>
</tr>
<tr>
<td>Mahmood et al.</td>
<td>2016</td>
<td>Convert RGB into gray; Square block</td>
<td>KPCA; DCT</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Multiple; JPEG compression (70:5:90); AWGN (20:5:40); Gaussian blurring (5, 0.5:5:3.0)</td>
<td>DVMM; Internet</td>
</tr>
<tr>
<td>Cao et al.</td>
<td>2012</td>
<td>Convert RGB into gray; Circular block</td>
<td>DCT</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Multiple; AWGN (10:5:35); Gaussian blurring (5, 0.5:5:3.0)</td>
<td>DVMM; Kodak; Internet</td>
</tr>
<tr>
<td>Wang et al.</td>
<td>2017</td>
<td>Convert RGB into gray; Square block</td>
<td>DCT</td>
<td>Package clustering; Euclidean distance</td>
<td>-</td>
<td>Multiple; AWGN (10:10:50); Gaussian blurring (2, 0.5:1:0:2.5); Mixture operations</td>
<td>MICC-F; DVMM; PIMPRCG</td>
</tr>
</tbody>
</table>
Table 1. (Continued)

<table>
<thead>
<tr>
<th>Ref</th>
<th>Year</th>
<th>Pre-processing</th>
<th>Feature extraction</th>
<th>Method for searching similar blocks</th>
<th>Post-processing</th>
<th>Performance</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhao and Guo [46]</td>
<td>2013</td>
<td>Convert RGB into gray; Square block</td>
<td>DCT; SVD</td>
<td>Sorting; Euclidean distance</td>
<td>Morphologic operations</td>
<td>Multiple; Gaussian blurring ((5, 0.5, 0.5, 1.0); AWGN (20:545); JPEG compression (65:595); Mixture operations</td>
<td>USC-SIPI; Kodak; Internet</td>
</tr>
<tr>
<td>Doyoddoji and Rhee [47]</td>
<td>2014</td>
<td>Square block</td>
<td>Radon transform; DCT</td>
<td>Sorting; Euclidean distance</td>
<td>Filtering</td>
<td>Multiple; AWGN (5:5:40); Gaussian blurring (5, 1.1:7); Mixture operations (JPEG compression+Rotation, JPEG compression+Scaling, Rotation+AWGN, Rotation+Scaling, etc.)</td>
<td>FAU</td>
</tr>
<tr>
<td>Muhammad et al. [51]</td>
<td>2012</td>
<td>DyWT; Square block</td>
<td>LL; HH</td>
<td>Sorting; Euclidean distance</td>
<td>-</td>
<td>Multiple; JPEG compression (60, 80, 90); Rotation (&lt;20°)</td>
<td>CASIA TIDE v1.0; KSU</td>
</tr>
<tr>
<td>Dixit et al. [53]</td>
<td>2017</td>
<td>Convert RGB into gray; SWT; Square block</td>
<td>LL; HH; SVD</td>
<td>Sorting; Euclidean distance</td>
<td>-</td>
<td>Single; Blurring</td>
<td>Internet</td>
</tr>
<tr>
<td>Bravo-Solorio and Nandi [57]</td>
<td>2011</td>
<td>Square block</td>
<td>Feature1</td>
<td>Sorting; Bravo</td>
<td>-</td>
<td>Multiple; JPEG compression (60:10:100); Rotation; Scaling (0.95:0.01:1.5); Flipping (horizontal); Mixture operations</td>
<td>Caltech-256</td>
</tr>
<tr>
<td>Li et al. [62]</td>
<td>2014</td>
<td>Convert RGB into gray; Gaussian low-pass filtering; Circular block</td>
<td>PST</td>
<td>Sorting; Euclidean distance</td>
<td>Filtering; Morphologic operations</td>
<td>Multiple; Scaling (0.1:0.2:1.1); Rotation (15°, 30°, 90°); Flipping (horizontal, vertical); AWGN (20:10:40); JPEG compression (40:20:80); Gaussian blurring (3,12:5); Mixture operations</td>
<td>Internet</td>
</tr>
<tr>
<td>Granty and Kousalya [63]</td>
<td>2016</td>
<td>Circular block</td>
<td>PCT; Spectral hashing; PCA</td>
<td>Sorting; Hamming distance; Euclidean distance</td>
<td>Filtering; Morphologic operations</td>
<td>Single; Blurring (0.5:0.5:3.0); Noise (1:2:9); JPEG compression (20:30:90); Rotation (20°:10°:90°)</td>
<td>FAU</td>
</tr>
<tr>
<td>Emam et al. [64]</td>
<td>2016</td>
<td>Circular block</td>
<td>PCET</td>
<td>ISH</td>
<td>Morphologic operations</td>
<td>Multiple; JPEG compression (0.20:80); Gaussian noise (0.002:0.002:0.006); Rotation (9°, 2°, 10°, 60°, 180°); Scaling (0.93:0.04:1.9)</td>
<td>FAU</td>
</tr>
<tr>
<td>Wo et al. [66]</td>
<td>2016</td>
<td>Circular block</td>
<td>Multi-radius PCET</td>
<td>Sorting</td>
<td>Morphologic operations</td>
<td>Multiple; JPEG compression (20:20:100); Gaussian blurring (3, 0.5), (5, 0.5); Gaussian noise (0, 0.0004:0.0002:0.001); Rotation (0°:30°:360°); Scaling (0.5:0.1:2.0)</td>
<td>FAU; Kodak</td>
</tr>
<tr>
<td>Zhong et al. [67]</td>
<td>2017</td>
<td>Convert RGB into gray; Circular block</td>
<td>DRPCET</td>
<td>Sorting; Person correlation coefficient</td>
<td>-</td>
<td>Single; Rotation (0°:20°:180°); Gaussian noise; Scaling; Mixture operations (Rotation+Scaling, Rotation+Gaussian noise)</td>
<td>MICC-F2000; Internet</td>
</tr>
<tr>
<td>Zhong et al. [68]</td>
<td>2017</td>
<td>Convert RGB into gray; Gaussian filtering; Circular block</td>
<td>DRHFM</td>
<td>Sorting; Euclidean distance</td>
<td>-</td>
<td>Multiple; Rotation; Gaussian noise; Scaling; JPEG compression; Mixture operations (Rotation+Scaling, Rotation+JPEG compression, etc.)</td>
<td>MICC</td>
</tr>
<tr>
<td>Zhong and Gan [70]</td>
<td>2016</td>
<td>Convert RGB into gray; Circular block</td>
<td>DAFMT</td>
<td>Sorting; Spearman rank correlation coefficient</td>
<td>-</td>
<td>Single; Rotation (18°, 180°); Scaling (0°:1.4:1.4); Gaussian noise; Mixture operations (Rotation+Scaling+Noise adding, etc.)</td>
<td>Internet</td>
</tr>
</tbody>
</table>
Table 1. (Continued)

<table>
<thead>
<tr>
<th>Ref</th>
<th>Year</th>
<th>Pre-processing</th>
<th>Feature extraction</th>
<th>Method for searching similar blocks</th>
<th>Post-processing</th>
<th>Performance</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huang et al. [74]</td>
<td>2017</td>
<td>Convert RGB into gray; Square block;</td>
<td>FFT; SVD; PCA</td>
<td>Matchers</td>
<td>City filtering; Horizontal filtering; Vertical filtering; Frequency filtering</td>
<td>Multiple; JPEG compression (20, 50, 80, 90); Gaussian blurring (3:4:7, 1); Gaussian noise (0, 5:5:20)</td>
<td>Internet; CASIA TIDE v1.0</td>
</tr>
<tr>
<td>Davarzani et al. [75]</td>
<td>2013</td>
<td>Convert RGB into gray; Circular block; Low-pass filtering;</td>
<td>MLBP</td>
<td>Sorting; K-d tree</td>
<td>RANSAC</td>
<td>Multiple; Rotation; Scaling; JPEG compression; Gaussian blurring; Gaussian noise; Mixture operations</td>
<td>Internet; PIMPR CG</td>
</tr>
<tr>
<td>Kuznetsov and Myasnikov v [79]</td>
<td>2016</td>
<td>Square block</td>
<td>BGC; LBP</td>
<td>K-d tree</td>
<td>Filtering</td>
<td>Single; Contrast enhancement; Gaussian noise; JPEG compression (40:10:90)</td>
<td>Internet</td>
</tr>
<tr>
<td>Ryu et al. [80]</td>
<td>2013</td>
<td>Square block</td>
<td>Zernike moment</td>
<td>LSH</td>
<td>RANSAC</td>
<td>Multiple; Rotation(0°:10°:90°); JPEG compression (40:20:100); AWGN (2:2:8); Linear filtering (0.5:0:5:2.5); Scaling</td>
<td>FAU</td>
</tr>
<tr>
<td>Mahmoud and Abu-Alrukab [84]</td>
<td>2016</td>
<td>Square block</td>
<td>Zernike moment; PZM</td>
<td>Sorting; Euclidean distance; Physical distance</td>
<td>-</td>
<td>Multiple; Color reduction; Additive noise; Contrast adjustment; Blurring; Brightness change; Rotation; Scaling</td>
<td>CoMoFoD</td>
</tr>
<tr>
<td>Malviya and Ladhake [92]</td>
<td>2016</td>
<td>Square block</td>
<td>ACC</td>
<td>Manhattan distance</td>
<td>-</td>
<td>Multiple</td>
<td>CoMoFoD</td>
</tr>
<tr>
<td>Cozzolino et al. [100]</td>
<td>2015</td>
<td>Circular block</td>
<td>CHT</td>
<td>PatchMatch</td>
<td>Linear filtering</td>
<td>Multiple; JPEG compression (20:10:100); Scaling; Rotation; Noise</td>
<td>GRIP; FAU</td>
</tr>
</tbody>
</table>

4. Keypoint-Based CMFD Methods

Typical keypoint-based CMFD methods selected from many keypoint-based methods are presented in this section, such as SIFT, dense scale-invariant feature transform (DSIFT), affine-scale-invariant feature transform (ASIFT), SURF, Harris corner feature, DAISY, mirror reflection invariant feature transform (MIFT) [112], multi-support region order-based gradient histogram (MROGH) [113,114].

In [115], the authors extracted SIFT descriptor as the feature, and best-bin-first (BBF) search method is used to match the similar feature. Pan and Lyu [15] estimated the geometric transform between matched SIFT keypoints and found the duplicated regions. Amerini et al. [116-118] proposed SIFT-based CMFD methods. In [116], maximum likelihood estimation (MLE) of the homograph and RANSAC algorithm are used for geometric transformation estimation. Amerini et al. [117] proposed a generalized 2NN test for multiple duplicated regions localization and agglomerative hierarchical clustering is used to identify the possible cloned region. In [118], they introduced J-Linkage algorithm to improve their works. Jin and Wan [119] used non-maximum value suppression and optimized J-Linkage to ameliorate the performance of SIFT-based CMFD methods. In [120], the authors proposed a SIFT-based CMFD scheme by using the SIFT keypoints extracted from actual part obtained by
performing DyWT to the image. Different with the manner of converting the color image into the gray image in pre-processing, Gong and Guo [121] extracted the color gradient from the suspicious image and took the gradient as the only input for SIFT extraction. In [122], the authors converted the color image into HSV. To solve value setting, Zhao and his colleagues [123,124] proposed a CMFD method based on SIFT with particle swarm optimization (PSO). ASIFT [125] and DSIFT [126] are also used in CMFD, respectively. In [127], the authors used expectation maximization (EM) algorithm to estimate the transform matrix. Warif et al. [128] presented a CMFD method that combined SIFT-based CMFD scheme with symmetry-based matching.

Shivakumar and Baboo [129] proposed a CMFD scheme based on SURF and k-d tree was used for feature matching. Mishra et al. [130] combined SURF and hierarchical agglomerative clustering (HAC) and presented a CMFD method. After multi-scale analysis and voting processes, Silva et al. [131] presented a CMFD scheme by using SURF as the feature. By combining adaptive minimal-maximal suppression (AMMS) and SURF, Yang et al. [132] presented a CMFD method to solve the problem of insufficient keypoints in the uniform area. SLIC was used for image segmentation, and SURF was used as the feature to find the duplicated region in [24].

Chen et al. [133] proposed a CMFD scheme based on Harris corner points and step sector statistics, in which BBF algorithm was used to find duplicated region. By combining Harris corner points and LBP, Zhao and Zhao [134] presented a scheme to detect region duplication in images. Wang et al. [135] used the statistical features of the Harris corner keypoints neighborhoods as forensics feature, and a new feature matching method was used for the improvement of detected accuracy. Combining the angular radial partitioning and Harris keypoints, Uliyan et al. [136] presented a CMFD scheme.

In recent several years, many CMFD schemes based on hybrid keypoints have been proposed and implemented, such as SIFT, SURF, and Harris corner [137], SURF and SIFT [138], SURF and binary robust invariant scalable keypoints (BRISK) [139], Harris corner points and BRISK [140], SURF, SIFT, and histogram oriented gradient (HOG) [141,142], MROGH and Harris corner points [143], and KAZE and SIFT [144].

State-of-the-art keypoint-based CMFD algorithms and some of the classical schemes are described in Table 2. Table 2 describes the methods from the below aspects: feature, performance, and dataset. In ‘Performance’, the second item is the visualization form. In ‘Dataset’, SATA-130 is included in FAU.

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Year</th>
<th>Feature</th>
<th>Performance</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jaberi et al.[112]</td>
<td>2014</td>
<td>MIFT</td>
<td>Single; Closed region; Scaling; Rotation; Deformation; Mixture operations (Scaling+Blurring, Scaling + Rotation + Blurring, Rotation + Blurring, etc.)</td>
<td>CASIA TIDE v2.0</td>
</tr>
<tr>
<td>Yu et al. [113]</td>
<td>2016</td>
<td>MROGH</td>
<td>Multiple; Closed region; JPEG compression (20:10:100); Rotation (2°:2°:10°, 20°, 60°, 180°); Scaling ([0.5, 2.0]); AWGN (0.02:0.02:0.10); Mixture operation (Rotation + Scaling)</td>
<td>FAU; MICC-F2000</td>
</tr>
<tr>
<td>Amerini et al.[117]</td>
<td>2011</td>
<td>SIFT</td>
<td>Multiple; Lines; Rotation; Scaling; Mixture operation (Rotation + Scaling)</td>
<td>MICC-F2000; MICC-F220</td>
</tr>
<tr>
<td>Amerini et al.[118]</td>
<td>2013</td>
<td>SIFT</td>
<td>Multiple; Lines; Rotation; Scaling; Mixture operation (Rotation + Scaling)</td>
<td>MICC-F2000; SATA-130; MICC-F600</td>
</tr>
<tr>
<td>Karsh et al. [125]</td>
<td>2016</td>
<td>ASIFT</td>
<td>Single; Points and lines</td>
<td>CoMoFoD</td>
</tr>
</tbody>
</table>
Table 2. (Continued)

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Year</th>
<th>Feature</th>
<th>Performance</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Li et al. [127]</td>
<td>2015</td>
<td>DSIFT</td>
<td>Multiple; Closed region; Noise (20:20:100); JPEG compression (20:10:100); Rotation (2°:2°:10°); Scaling (0.91:0.02:1.09)</td>
<td>FAU; MICC-F600</td>
</tr>
<tr>
<td>Mishra et al.</td>
<td>2013</td>
<td>SURF</td>
<td>Single; Points and lines; JPEG compression (20:20:80); AWGN(20:10:50); Gaussian blurring ((5, 0.5:0.5:1.0), (7.0:5.0:5.1:0.0)); Gamma correction (1.2:0.2:1.8); Scaling; Rotation</td>
<td>MICC-F220</td>
</tr>
<tr>
<td>Chen et al.</td>
<td>2013</td>
<td>Harris corner points</td>
<td>Single; Circulars and lines; Rotation; Scaling (0.8, 0.9, 1.1, 1.2, 1.3); Mixture operation (Rotation+Scaling); Flipping (horizontal); JPEG compression (50:10:90); AWGN (20:5:40)</td>
<td>FAU; Kodak; CASIA TIDE v2.0</td>
</tr>
<tr>
<td>Zhao and Zhao</td>
<td>2013</td>
<td>Harris corner points; LBP</td>
<td>Multiple; Lines; Rotation; Gaussian blurring; Flipping (horizontal, vertical); JPEG compression; AWGN; Mixture operations (Rotation+Gaussian blurring, Flipping+Gaussian blurring, etc.)</td>
<td>FAU; Kodak; Internet; CASIA TIDE v2.0</td>
</tr>
<tr>
<td>Ardizzone et al.</td>
<td>2015</td>
<td>SIFT; SURF; Harris corner points</td>
<td>Single; Closed region; Rotation; Scaling</td>
<td>CVIP</td>
</tr>
<tr>
<td>Pandey et al.</td>
<td>2015</td>
<td>SURF; SIFT</td>
<td>Single; -; Rotation; Scaling; Mixture operation (Rotation+Scaling)</td>
<td>MICC-F220</td>
</tr>
<tr>
<td>Kumar et al.</td>
<td>2015</td>
<td>SURF; BRISK</td>
<td>Multiple; Points; JPEG compression (40:10:100); Gaussian noise (0:0.02:0.10)</td>
<td>FAU; CoMoFoD</td>
</tr>
<tr>
<td>Isaac and Wilscy</td>
<td>2015</td>
<td>Harris corner points; BRISK</td>
<td>Single; Lines; Noise adding; Brightness change; Color reduction; Blurring</td>
<td>CoMoFoD; MICC-F220</td>
</tr>
<tr>
<td>Prasad and Ramkumar</td>
<td>2016</td>
<td>SIFT; HOG; SURF</td>
<td>Single; Lines</td>
<td>MICC-F220</td>
</tr>
<tr>
<td>Yang et al.</td>
<td>2017</td>
<td>KAZE; SIFT</td>
<td>Multiple; Closed region; Rotation (2°:2°:10°); Scaling (0.91:0.02:1.09); Gaussian noise (0, 0.02:0.02:0.10); JPEG compression (20:10:100)</td>
<td>FAU</td>
</tr>
</tbody>
</table>

5. Performance Evaluation Criterions and Datasets

5.1 Performance Evaluation Criterions

The performance of CMFD methods is usually evaluated from two aspects: the image level and the pixel level. The most frequently used performance evaluations are Precision $p$, Recall $r$, and $F_1$ score [145], which are shown in Eqs. (4)–(6), respectively.

$$p = \frac{T_p}{T_p + F_p}, \quad (4)$$

$$r = \frac{T_p}{T_p + F_N}, \quad (5)$$

$$F_1 = 2 \cdot \frac{p \cdot r}{p + r}, \quad (6)$$
where $T_p$ denotes the number of doctored images correctly detected as doctored images; $F_p$ denotes the number of authentic images erroneously detected as doctored images; and $F_N$ denotes the number of doctored images falsely detected as authentic images, at image level. At pixel level, $T_p$ denotes the number of correctly detected as doctored pixels; $F_p$ denotes the number of falsely detected as doctored pixels; and $F_N$ denotes the number of falsely detected as authentic pixels. The larger the $p$, $r$, and $F_1$ are, the higher the accuracy of the CMFD scheme is.

Zhao and Guo [46] presented another evaluation criterion at pixel level, the detection accuracy rate $R_{DA}$ and the false positive rate $R_{FP}$, which are shown in Eqs. (7) and (8), respectively.

$$R_{DA} = \frac{|\psi_C \cap \psi_{DC} | + |\psi_p \cap \psi_{DP} |}{|\psi_C | + |\psi_p |},$$  

$$R_{FP} = \frac{|\psi_{DC} - \psi_C | + |\psi_{DP} - \psi_p |}{|\psi_{DC} | + |\psi_{DP} |},$$

where $| |$ denotes the area of the copied region or pasted region, $\cap$ denotes the intersection of two regions, $-$ denotes the difference between two regions, $\psi_C$ denotes the pixels of the copied region, $\psi_p$ denotes the pixels of the pasted region, $\psi_{DC}$ denotes the pixels of detected copied region, and $\psi_{DP}$ denotes the pixels of detected pasted region. The closer $R_{FP}$ is to 0 and $R_{DA}$ is to 1, the higher the accuracy of the CMFD method is.

### 5.2 Datasets

Diverse datasets for CMFD are listed in this sub-section. A good dataset for CMFD should have the original images, the forged images, the distorted forged images, and their corresponding ground truth maps, as shown in Fig. 6, which are from the CoMoFoD dataset [146]. Some commonly used datasets for the evaluation of CMFD methods are collected in Table 3, and their corresponding links are shown in References.

Besides these datasets mentioned above, many methods created their own datasets by using images from the Internet and other datasets that are also collected in this survey [104-111,151].

![Fig. 6](image_url)  
**Fig. 6.** Example of the CoMoFoD dataset: (a) original image, (b) forged image, (c) forged image with image blurring, and (d) ground truth map.
Table 3. Dataset comparison

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Content</th>
<th>Content detail</th>
<th>Ground truth map</th>
<th>Format (size)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRIP [100]</td>
<td>80 tampered color images 80 authentic color images</td>
<td>80 single plain tampered images</td>
<td>Yes</td>
<td>PNG (1024×768; 768×1024)</td>
</tr>
<tr>
<td>CVIP [137]</td>
<td>1,060 tampered color images 50 authentic color images</td>
<td>680 single tampered images (rotation) 380 single tampered images (scaling)</td>
<td>Yes</td>
<td>BMP 1000×700 or 700×1000</td>
</tr>
<tr>
<td>MICC-F8multi [147]</td>
<td>8 tampered color images</td>
<td>8 multiple tampered images</td>
<td>No</td>
<td>JPG (2048×1536, 800×532, 947×683)</td>
</tr>
<tr>
<td>MICC-F220 [147]</td>
<td>110 tampered color images 110 authentic color images</td>
<td>110 single tampered images (rotation, scaling)</td>
<td>No</td>
<td>JPG (from 722×480 to 800×600)</td>
</tr>
<tr>
<td>MICC-F2000 [147]</td>
<td>700 tampered color images 1,300 authentic color images</td>
<td>700 single tampered images (rotation, scaling)</td>
<td>No</td>
<td>JPG (2048×1536)</td>
</tr>
<tr>
<td>MICC-F600 [147]</td>
<td>152 tampered color images 448 authentic color images</td>
<td>38 single plain tampered images; 38 multiple plain tampered images; 38 images in which the copied region is rotated by 30°; 38 images in which the copied region is rotated by 30° and scaled by 120%</td>
<td>Yes</td>
<td>PNG, JPG (from 722×480 to 800×600)</td>
</tr>
<tr>
<td>FAU [145]</td>
<td>48 color image sets (1632×1224) 48 color image sets (3039×2014)</td>
<td>Single or multiple tampered images (translation, rotation, scaling, distortion, combination)</td>
<td>Yes</td>
<td>PNG, JPG (1632×1224; 3039×2014)</td>
</tr>
<tr>
<td>CoMoFoD [146]</td>
<td>200 color image sets (512×512) 60 color image sets (3000×2000)</td>
<td>Single or multiple tampered images (translation, rotation, scaling, distortion, combination)</td>
<td>Yes</td>
<td>PNG, JPG (512×512; 3000×2000)</td>
</tr>
<tr>
<td>CASIA TIDE v1.0 [148]</td>
<td>921 tampered color images 800 authentic images</td>
<td>480 tampered images within same images 451 tampered images from different images (rotation, deformation, resize)</td>
<td>No</td>
<td>JPG (384×256)</td>
</tr>
<tr>
<td>CASIA TIDE v2.0 [148]</td>
<td>5,123 tampered color images 7,491 authentic color images</td>
<td>5,123 tampered color images (rotation, deformation, resize)</td>
<td>No</td>
<td>TIF, JPG (from 240×160 to 900×600)</td>
</tr>
<tr>
<td>COVERAGE [149]</td>
<td>100 tampered color images 100 authentic color images</td>
<td>100 single tampered color images (translation, scaling, rotation, free-form, illumination, combination)</td>
<td>Yes</td>
<td>TIF (400×486)</td>
</tr>
<tr>
<td>DVMM [150]</td>
<td>912 tampered gray images 933 authentic gray images</td>
<td>180 single plain tampered images</td>
<td>No</td>
<td>BMP (128×128)</td>
</tr>
</tbody>
</table>

6. Future Direction and Conclusion

6.1 Future Direction

On the basis of the existing problems in current research status, several future directions for CMFD research are provided in this subsection based on the existing problems.

- *Benchmark dataset*. A dataset is indispensable to evaluate the performance of CMFD method. Dataset for CMFD evaluation should include original images and corresponding forged images
with different resolution, diverse forged images with regions (smooth or texture) which have different size in various geometric transformation (rotation, scaling, etc.), the forged region saved individually as images, the distorted images with post-processing methods (JPEG compression, AWGN, noise contamination, blurring, etc.). Besides, the corresponding ground truth maps and post-processing methods with open-source code (MATLAB, OpenCV) also should be included in dataset.

- **Effectiveness and robustness.** CMFD methods should be effective to detect the forged regions in distorted doctored images as mentioned in benchmark dataset. It is worth exploring efficient local invariant feature and descriptors extraction, high-speed method of feature matching, and accurate localization method.

- **Deep learning.** It is relatively few CMFD methods based on deep learning. The application of deep learning is only used in the classification of authentic images and forged images, and it is hard to determinate the accurate forged regions. It is also a difficult problem that these CMFD methods based on deep learning are hard repeatable and used for comparison because of the difference of the training set and testing set or the complex experiments. The researchers study this topic by using deep learning technologies in the future, such as deep Boltzmann machines [152] and CNN [153].

### 6.2 Conclusion

Passive forensics technology of digital image is one of the rapidly growing fields of research. Our brief review of image CMFD technologies indicates that the research is still in the phase of vigorous development and has a huge potential for the future research and development applications. Two classical models of copy-move forgery and two frameworks of CMFD technologies are presented at first. Then, block-based and keypoint-based CMFD methods are reviewed from different aspects, respectively, including the classical CMFD technologies and the state-of-the-art algorithms for CMFD in recent several years. The performance evaluation criterions and frequently used datasets for evaluating the performance of the CMFD schemes are collected. The future directions of this topic are given at last. With the help of the advanced technologies, some CMFD schemes with high performance are expected to become standard tools in the future. We also hope that this survey will provide related information to scientists, researchers, and relevant research communities in this field. The investigation on image forensics is still a continual, sustainable process and it will continue to explore forensics technologies with high accuracy and robustness.

### Acknowledgement

This work was supported by the National Natural Science Foundation of China (No. 61702303, No. 61201371); the Natural Science Foundation of Shandong Province, China (No. ZR2017MF020, No. ZR2015PF004); and the Research Award Fund for Outstanding Young and Middle-Aged Scientists of Shandong Province, China (No. BS2013DX022). The authors thank Surong Zhang, Xiuhong Wei, and Chi Wang for their kind help and valuable suggestions in revising this paper.
References


A Survey on Passive Image Copy-Move Forgery Detection


[150] T. T. Ng and S. F. Chang, “A data set of authentic and spliced image blocks,” Columbia University, New York, NY, A D V E N T T e c h n i c a l R e p o r t # 2 0 3 - 2 0 0 4 - 3 , 2 0 0 4 .


Zhi Zhang  https://orcid.org/0000-0002-1476-6790
He was born in Shandong province, China, in 1992. He received his B.E. degree in electronic information engineering from Shandong University of Science and Technology, China, in 2016. He is currently pursuing his M.E. degree in information and communication engineering at Shandong University, China. His current research interests include image watermarking, forgery detection, and computer vision.

Chengyou Wang  https://orcid.org/0000-0002-0901-2492
He was born in Shandong province, China, in 1979. He received his M.E. and Ph.D. degrees in signal and information processing from Tianjin University, China, in 2007 and 2010, respectively. He is currently an associate professor and supervisor of postgraduate students at Shandong University, Weihai, China. His current research interests include image/video processing and analysis, computer vision, and wireless communication technology.

Xiao Zhou  https://orcid.org/0000-0002-1331-7379
She was born in Shandong province, China, in 1982. She received her M.E. degree in information and communication engineering from Inha University, Korea, in 2005; and her Ph.D. degree in information and communication engineering from Tsinghua University, China, in 2013. She is currently a lecturer and supervisor of postgraduate students at Shandong University, Weihai, China. Her current research interests include wireless communication technology, digital image processing, and computer vision.