Research Article

Detection of Free-Form Copy-Move Forgery on Digital Images

Emre Gürbüz 1, Guzin Ulutas 2, and Mustafa Ulutas 2

1Tokat Technical Sciences Vocational School, Tokat Gaziosmanpaşa University, Tokat, Turkey
2Department of Computer Engineering, Karadeniz Technical University, Trabzon, Turkey

Correspondence should be addressed to Emre Gürbüz; emregurbuz@msn.com

Received 3 April 2019; Accepted 30 July 2019; Published 22 September 2019

1. Introduction

In the technology age that we live in, both the increment in the digital image manipulation software applications and the ease of using these have increased the probability of malevolent changes on digital images, which are also called digital image forgeries. Digital images, which are considered as the concrete proofs of facts, have a persuasive effect on people, and they are used as evidences in courtrooms. As a consequence, an obligation has arisen for authenticating the originality of digital images.

One of the most widely used forgery techniques is copy-move forgery. In 2003, Fridrich et al. proposed a block-based technique for detection of copy-move forgery [1]. This was the first copy-move forgery detection technique in the literature, and its working principle was covering a region on a forged image by another region on the same image. The mentioned technique presumes that the suspicious image consists of overlapping square blocks with sizes of 8 × 8. From each of these blocks, the technique extracts feature vectors with sizes of 1 × 64 by using discrete cosine transform (DCT), sorts these vectors lexicographically, and decides whether the blocks with adjacent feature vectors are similar or not. Herein, similarity between two blocks means that these blocks are copy of each other. In 2004, Popescu and Farid proposed another block-based technique for detecting copy-move forgery [2]. Their technique employs principal component analysis (PCA) for feature extraction, and in this way, it reduces the sizes of the feature vectors and decreases the complexity of the authentication process. However, the main disadvantage of the previous studies [1, 2] is that these techniques are unguarded against blurring and noise addition operations which are intentionally applied to the forged images in order to remove the fingerprints of the forgery operations. In 2007, Mahdian and Saic used blur invariants during the feature extraction process in order to render their technique robust against noise addition and blurring operations [3]. However, since their study presents only visual results, we could not make any numerical comparison between their study and ours. In 2011, in order to develop a technique that is robust against noise addition, blurring and JPEG compression operations,
Huang et al. proposed acquiring feature vectors by applying DCT to the blocks and using the previously determined parts of these vectors [4]. In 2012, Cao et al. proposed obtaining the frequency components of the image blocks by applying DCT to these blocks, separating these frequency components into four regions and using the mean values of each of these regions in order to construct the feature vectors with sizes of $1 \times 4$ [5]. Their study shows that the mentioned feature vectors are robust against noise addition and blurring operations. However, while both these studies [4, 5] present numerical results for the experiments performed on square-form selection copy-move forgeries, they present only visual results for the experiments performed on free-form selection copy-move forgeries so that we could not make any numerical comparison between these studies and our study. In 2011, Zimba and Xingming proposed a block-based technique which uses a combination of discrete wavelet transform (DWT) and principal component analysis-eigenvalue decomposition (PCA-EVD) [6]. The key point of this technique is reducing the sizes of the suspicious image and the feature vectors. This approach increases the speed of the forgery detection algorithm. Their test images include only square-form copy-move forgeries which are quite easy to be detected. However, their experimental results show that their technique is not stable against JPEG compression operations with decreasing JPEG quality factors. In 2014, Sharma proposed a block-based technique which utilizes DCT and singular value decomposition (SVD) [7]. Similar to the technique in [6], the main goal of this technique is reducing the sizes of the feature vectors and increasing the speed of the algorithm. Almost all of their experimental results belong to the test images which are exposed to square-form copy-move forgery, and these results show that their technique is not stable against JPEG compression operations with decreasing JPEG quality factors. In the study, there is only one instance of free-form copy-move forgery detection with varying JPEG quality factors, which does not give enough clues about the stability of their algorithm. In 2017, Wang et al. proposed a block-based technique that employs local binary pattern (LBP) and SVD for extracting features [8]. Their experiments include both rectangular-form and free-form copy-move forgeries, but since there is no numerical results belonging to these experiments, we could not evaluate their algorithm in terms of success rates and stability. In 2017, Ansari and Ghreira proposed a block-based method in which ring projection transform (RPT) and modified fast discrete Haar wavelet transform (MFHWT) are used [9]. Their method aims at reducing the computational work by using MFHWT, and it utilizes RPT for extracting features. However, since the effects of JPEG compression with varying quality factors are not examined in their study, it is not possible to evaluate the stability of their algorithm. Besides, since they calculate their success rates according to the numbers of correctly and incorrectly detected images instead of pixel-based metrics, we could not make any numerical comparison between their study and ours. In 2017, Hayat and Qazi proposed a block-based technique which uses DWT and DCT together for feature reduction [10]. Their technique deals with free-form copy-move forgery. However, as in [9], since the effects of JPEG compression with varying quality factors are not covered in their study, we were unable to evaluate the stability of their algorithm. In 2018, Alkawaz et al. proposed a block-based technique that uses DCT for extracting features [11]. However, as in [9, 10], we could not evaluate the stability of their algorithm since their study does not cover the effects of JPEG compression with varying quality factors. In our previous work, which is about detection of copy-rotate-move forgery, we used a simple block-matching approach in order to mark the matched region pairs and realized that this approach is inadequate and needs to be improved especially if the forged regions are in free-form [12]. Although the proposed technique is also block-based, unlike the other block-based techniques, it utilizes block-matching mechanism only for marking the forged region pairs partially, and then fully detects the forged areas with a novel approach which is described in the subsequent sections.

When a copy-move forgery is implemented on a digital image, the shape of the copied and pasted region is quite important for the forgery detection algorithm. If the forged region is selected in free-form, block-based forgery detection techniques have difficulty in detecting the forgery because the rectangular blocks may not match up with the outer sides of the forged regions. The success of the proposed technique has been tested against free-form copy-move forgery.

In this study, a novel technique has been developed for detecting free-form copy-move forgery, and all of the test images used for testing the success rates of the proposed technique include free-form copy-move forgery. The proposed technique is block-based, and it uses intensity coherence vector (ICV) for feature extraction. Unlike the classical block-based techniques, the proposed technique determines the forged regions by overlapping and differentiating two clones of the suspicious image, instead of marking the matched blocks. Details of the proposed technique will be given in the subsequent sections. By means of this novel technique, the success rate of the forgery detection operation remains stable even as the distortion strength of the postprocessing operations increases. We need to state that during our studies in the field of detecting free-form copy-move forgery, our success criteria are not only acquiring high average success rates but also achieving stable detection performance against various postprocessing operations which aim at making the detection of forgery harder.

2. Materials and Methods

In this study, a novel technique for detection of free-form copy-move forgeries is proposed. In a free-form copy-move forgery operation, on an original digital image (e.g., in Figure 1(a)), a determined area with an irregular shape is copied and pasted onto another area on the same image (e.g., in Figure 1(c)) according to a copy-move mask (e.g., in Figure 1(b)). The proposed technique divides the suspicious image into overlapping square blocks and uses ICV to extract features from these blocks. After extracting
features for each of the overlapping blocks, the feature vectors are lexicographically sorted, and according to the mechanism explained below, the blocks belonging to the similar regions are matched. In order to obtain higher success rates, instead of marking the areas belonging to the matched block pairs, a new approach is applied. According to this approach, the centers of the matched block pairs are computed and these are regarded as the reference points belonging to both the copied and the pasted regions. Then, by using these reference points,
which match the forged areas, two clones of the suspicious image are overlapped and differentiated, and the intersected areas with the lowest differentials are regarded as the forged region pairs. The details belonging to the aforesaid mechanism will be given under the following subsections.

2.1. Intensity Coherence Vector. Intensity coherence vector can be defined as the vector which consists of the numbers of coherent and incoherent pixels that are determined according to the intensities (luminances) of the pixels belonging to the image [13]. For a pixel to be categorized as coherent, it is obligatory to be a member of a large and adjacent pixel group with the same or similar intensities, otherwise the pixel is categorized as incoherent. Coherence of an intensity is the total number of pixels with that intensity or similar intensities which are members of large regions that consist of adjacent pixels. Coherent regions can be regarded as the signature of a block. Calculation of ICV of a block requires blurring as preprocessing. For each of the pixels constructing a block, the process of blurring includes using the intensity values of each of the adjacent pixels in order to compute the new value of the pixel. The next step contains discretizing the luminance space so that $n$ distinct intensities appear in the block. Following this step, each pixel of the block is categorized as coherent or incoherent, by the algorithm, according to the categorization rule described above. In order to get the fingerprint of the block, the connected components are computed in the next step. For any pixel pair $(p_1, p_2)$ having the same discretized intensity, if there is a connection between $p_1$ and $p_2$ via the pixels that also have the same discretized intensity, then they are regarded as connected components. The computation of the connected components is made for each of the discretized intensity segments. So as to determine whether a pixel is coherent or not, the algorithm uses a threshold value. If the connected component of that pixel has total number of elements greater than the mentioned threshold value, then the pixel is marked as coherent; if not so, then it is marked as incoherent. For each discretized intensity segment in a block, there can exist both coherent and incoherent pixels. Assuming that there are totally $s$ discretized intensity segments for a block, if there are $c_k$ coherent pixels and $i_k$ incoherent pixels in the $k^{th}$ discretized intensity segment, then the feature vector extracted from this block will be as given the following equation:

\[ \left\langle (c_1, i_1), (c_2, i_2), \ldots, (c_k, i_k), \ldots, (c_s, i_s) \right\rangle. \quad (1) \]

ICV can be regarded as a variation of color coherence vector (CCV), in which all of the color channels are used instead of using only intensity. There are some recent studies in which CCV is utilized in imaging applications [14–16].

2.2. Matching of the Similar Regions. Provided that $B$ is an odd number, the proposed technique firstly converts the suspicious image to grayscale and then divides the image into overlapping square blocks with size of $B \times B$. For each of these blocks, an intensity coherence vector is obtained, and these vectors are considered as the features of the blocks. The extracted features constitute the rows of the feature matrix (FM). Then, FM matrix is lexicographically sorted so that the feature vectors of the similar blocks exist in the close rows.

After the process of lexicographically sorting, in order to decide which blocks are similar, a group of elimination operations are needed together with block-matching operation. Five basic parameters required for these elimination operations are maximum lexigraphical distance ($LD_{\text{MAX}}$), minimum Euclidean distance ($ED_{\text{MIN}}$), nonuniformity threshold ($T_N$), maximum difference between blocks ($DB_{\text{MAX}}$), and maximum distance from mean ($DM_{\text{MAX}}$). The duties of these parameters in matching and elimination operations are explained below, and for each experiment, the values of these parameters are to be chosen by the implementer.

(i) $LD_{\text{MAX}}$: each row in the sorted feature matrix (FM) represents a single block. Considering that the feature vectors of similar blocks will exist in close positions on FM matrix, in order to prevent the dissimilar blocks from being considered as similar, $LD_{\text{MAX}}$ value is utilized. If the feature vector of a block has a distance greater than $LD_{\text{MAX}}$ value to the feature vector of another block on FM matrix, then the mentioned blocks are considered as dissimilar.

(ii) $ED_{\text{MIN}}$: this parameter is used for preventing the algorithm from being misled by the regions which consist of uniform colors (such as sky, or the surface of an object). The principle accepts that two similar blocks which are close to each other with respect to Euclidean distance are dissimilar. If the Euclidean distance between the coordinates ($x$ and $y$ coordinates belonging to the top-left corner) of two blocks is less than $ED_{\text{MIN}}$ value, then these blocks are considered as dissimilar.

(iii) $T_N$: in a similar way with the usage of $ED_{\text{MIN}}$ parameter, this parameter is also used for preventing the algorithm from being misled by the regions which consist of uniform colors (such as sky, or the surface of an object). However, $T_N$ parameter is used without regarding any relations (Euclidean distance, lexicographical distance, etc.) between two blocks. For each block, a nonuniformity value is computed. For this, firstly, each block is vectorized (converted to a row vector from the matrix form). Then, for each of these vectors, a determined element is subtracted from all the remaining elements and the absolute values of the results acquired after these subtraction operations are computed. When all of these absolute values are summed, the acquired value will be the sum of the absolute values of the subtraction operations between the mentioned element and the remaining elements of the vector. In a similar manner, sums of absolute values are computed for the remaining elements of the vector, and, by averaging these sums of absolute values, a nonuniformity value is computed for each block.
Finally, all of the blocks nonuniformity values of which are less than $T_N$ are eliminated because it is considered that the blocks having uniform colors cannot give enough clues for the algorithm to be able to recognize and distinguish the objects. The eliminated blocks are never taken into consideration during the block-matching processes.

(iv) $DB_{\text{MAX}}$: this is another parameter which is used for deciding whether two blocks are similar or not. When any two blocks are superposed and the sum of the absolute values of the subtraction of each block element is computed, if this sum is greater than $DB_{\text{MAX}}$ value, these two blocks are regarded as dissimilar.

(v) $DM_{\text{MAX}}$: this parameter is used for eliminating the incorrect matchups. At the last step of the block-matching process, the centers of gravity of both the source blocks and the destination blocks are computed separately. All the source blocks which are at least $DM_{\text{MAX}}$ far away from the center of gravity of the source blocks are eliminated together with their pairs in the destination blocks. In a similar way, all of the destination blocks which are at least $DM_{\text{MAX}}$ far away from the center of gravity of the destination blocks are eliminated together with their pairs in the source blocks.

Each block features of which exist on the feature matrix FM is compared with all of the blocks except itself. The block pairs, which do not conflict with any of the elimination rules belonging to the first four parameters mentioned above, are considered as similar. Then, a second elimination operation is processed according to the fifth parameter mentioned above. After these operations, the remaining block pairs are considered as similar. In classical block-based copy-move forgery detection techniques, the values of such parameters directly affect the success rates because these techniques determine the forged regions according to the block-matching operations by using these parameters. However, our technique needs these parameters only for determining the relative positions of the forged regions, and it determines the forged regions in an original way explained in the following subsection.

The technique which is given in detail above assumes that a region on the forged image is copied and then pasted on a different location on the same image. Assuming that the coordinates of the source blocks are kept in a matrix named $M_{\text{SBI}}$ and the coordinates of the destination blocks are kept in a matrix named $M_{\text{DBI}}$, it is expected that all of the blocks coordinates of which are kept in the $M_{\text{SBI}}$ matrix exist close to each other but far away from the blocks coordinates of which are kept in $M_{\text{DBI}}$ while all of the blocks coordinates of which are kept in the $M_{\text{DBI}}$ matrix exist close to each other but far away from the blocks coordinates of which are kept in $M_{\text{SBI}}$. However, the results of the conducted experiments showed that while some of the block coordinates belonging to the source region fell into the $M_{\text{SBI}}$ matrix, some of them fell into the $M_{\text{DBI}}$ matrix, and in a similar manner, while some of the block coordinates belonging to the destination region fell into the $M_{\text{DBI}}$ matrix, some of them fell into the $M_{\text{SBI}}$ matrix. Although this situation does not cause any problem in the block-matching process since the indice values of the matched blocks are stored in the parallel indices of $M_{\text{SBI}}$ and $M_{\text{DBI}}$ matrices, it may cause problems during the subsequent stages in which the copied and pasted areas are accurately detected. For this reason, $M_{\text{SBI}}$ and $M_{\text{DBI}}$ matrices are exposed to a polarization process (based on their center of gravity) so that it is ensured that $M_{\text{SBI}}$ and $M_{\text{DBI}}$ matrices store only the coordinates of the blocks belonging to source and destination regions, respectively. Figure 2 describes the whole process in a simplified way.

2.3. Determination of the Similar Regions. Because of understandability concerns, we prefer step-by-step explanations besides a flow chart. Determination of the similar regions is performed according to the following steps:

(i) An empty data matrix ($D_M$), elements of which are to be determined on the oncoming steps of the algorithm, is defined.

(ii) By adding $(B−1)/2$ to mean of all of the $x$ and $y$ coordinates stored in $M_{\text{SBI}}$ matrix, $S_xavg$ and $S_yavg$ values are acquired, respectively. At the same time, by adding $(B−1)/2$ to mean of all of the $x$ and $y$ coordinates stored in $M_{\text{DBI}}$ matrix, $D_xavg$ and $D_yavg$ values are acquired, respectively. While $(S_xavg, S_yavg)$ gives the center of gravity of the source blocks, $(D_xavg, D_yavg)$ gives the center of gravity of the destination blocks.

(iii) A copy of the suspicious image is created.

(iv) Firstly, a convergence point is determined by putting the copy of the suspicious image onto the original suspicious image so that $S_xavg$ and $S_yavg$ coordinates in the original image will coincide with the $D_xavg$ and $D_yavg$ coordinates on the copy image. Then, a movement distance ($D_{\text{MOV}}$) parameter, which accepts only positive integer values, is defined. Next, taking the mentioned convergence point as the reference point, the copy image is moved step by step within the region between $D_{\text{MOV}}$ pixels up and $D_{\text{MOV}}$ pixels down and between $D_{\text{MOV}}$ pixels left and $D_{\text{MOV}}$ pixels right (see Figure 3). In our experiments, we have chosen the value of $D_{\text{MOV}}$ parameter to be 10 since it gave the best results in terms of success rates and computation speed, and this means that the mentioned moving operation consists of $21 \times 21 = 441$ steps.

(v) In each step of the moving operations mentioned in the previous item, the absolute value of the difference between the circle with diameter $B$ which is centered on the $(D_xavg, D_yavg)$ coordinate of the copy image and the circle which corresponds to the projection of the mentioned circle on the original image is computed (this absolute value is
denoted by $\Delta$). In addition, the differences between the coordinates $Dx_{avg}$, $Dy_{avg}$ on the image being moved (copy image) and the coordinates $Sx_{avg}$, $Sy_{avg}$ on the fixed (original) image ($DIFFx_{avg}$, $DIFFy_{avg}$) are computed by using the statements $DIFFx_{avg} = Sx_{avg} - Dx_{avg}$ and $DIFFy_{avg} = Sy_{avg} - Dy_{avg}$.

(vi) During each computation in the previous item, the vector which consists of the values $\Delta$, $DIFFx_{avg}$, and $DIFFy_{avg}$ is inserted into the next empty row of $D_M$ matrix.

(vii) After $D_M$ matrix is lexicographically sorted, $DIFFx_{avg}$ and $DIFFy_{avg}$ values which exist in the first row of this matrix will give the relative position of the image being moved with respect to the fixed image. It must be noted that the mentioned relative position is acquired in the movement in which the minimum difference between the suspicious regions of the copy and original images is obtained.

(viii) The absolute value of the differences between the intersected regions of the two images, one of which is placed onto the other by using $DIFFx_{avg}$ and $DIFFy_{avg}$ values, is computed (in this stage, it is expected that two objects one of which is the copy of the other are overlapped and the absolute value of the difference between the regions belonging to these two objects is equal to or close to 0.), and thus, a differential image is acquired. In order to remove noise, a two-dimensional median filter is applied to this differential image, and then, on the differential image, the regions which consist of adjacent pixels having values less than a threshold value ($\tau_F$) are detected.

(ix) After the detection process mentioned in the previous item, depending on the structure of the image, there may be numerous detected regions with miscellaneous dimensions on the differential image. Between these regions, only the one which intersects with $(Sx_{avg}, Sy_{avg})$ coordinate of the original image is
selected, and the remaining ones are eliminated. This selected region corresponds to the source region on the original image, and it corresponds to the destination region on the copy image.

(x) After region detection, a grayscale result mask \(M_{\text{RES}}\) which has the same dimensions with the suspicious (original) image is generated and all of the indices of this mask are equalized to 0. Since the selected region corresponds to the source region on the original image, this region is directly marked on the result mask by assigning 255 to the indices of \(M_{\text{RES}}\) mask which directly correspond to the selected region. Since the selected region corresponds to the destination region on the copy image, the relative position of the selected region on the copy image is calculated and all of the indices of \(M_{\text{RES}}\) mask which correspond to the region that exists on the mentioned relative position are also equalized to 255. After these operations, on the \(M_{\text{RES}}\) mask, while the areas which correspond to the copy-move regions are colored with white, the remaining of the mask is colored with black.

(xi) The acquired \(M_{\text{RES}}\) mask is the end result of the developed forgery detection technique, and it is used for matching the regions, one of which is the copy of the other, on the suspicious image that is subject to the forgery detection operation.

The output image of the algorithm is created by coloring the pixels of the suspicious image which correspond to the white pixels of the \(M_{\text{RES}}\) mask with a specific color (e.g., in Figure 1(d)). Figure 4 describes the whole process in a simplified way.

3. Results and Discussion

The proposed technique has been tested on 224 experiments all of which include free-form copy-move forgery. In real life, copy-move forgery is applied on both the images on the internet and the images which are originally acquired by the forgers (digital camera images). In order to test our technique on real-life situations, we have constructed a dataset and named it Free-Form Copy-Move Forgery (FFCMF) dataset [17]. This dataset includes 160 forged images which are exposed to different attacks. While 128 of these images were generated by using original images acquired from internet, the remaining 32 of them were generated by using an original image from our camera. Besides, so as to compare the results of the proposed technique with the results of the other studies [18–20], we also tested our technique on 64 free-form copy-move forgery images which originate from CoMoFoD dataset [21]. During the test operations, in order to measure the strength of the forgery detection algorithm against the postprocessing operations which aim at making the detection of the forgery harder, the test images were exposed to JPEG compression, Gaussian filtering, and different combinations of these two operations. By way of these operations, 32 distinct experimental images can be generated from a free-form copy-move forgery image.

Since different metrics have been used in the recent studies in the field of our study for measuring performance, it has become compulsory to evaluate the performance of our algorithm in terms of various metrics. Because the copy-move forgery detection algorithms that we are interested in cannot determine which of the matched copy-move region pairs is the original one and which is the copy of the other, we do not have to make a distinction between the copied regions and the pasted regions. So that, while TP denotes the total number of pixels which belong to the forged region pairs and marked as forged by the algorithm (true positive), TN denotes the total number of pixels which do not belong to the forged region pairs and marked as not forged by the algorithm (true negative), FP denotes the total number of pixels which do not belong to the forged region pairs but marked as forged by the algorithm (false positive), and FN denotes the total number of pixels which belong to the forged region pairs but marked as not forged by the algorithm (false negative), the metrics verification rate \(V_r\), accuracy rate \(A_r\), error rate \(E_r\), false-positive rate \(FP_r\), and false-detection rate \(FD_r\) can be stated as in equations (2)–(6), respectively.

\[
V_r = \frac{TP}{TP + FN} \quad (2)
\]

\[
A_r = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)
\]

\[
E_r = \frac{FP + FN}{TP + FN} \quad (4)
\]

\[
FP_r = \frac{FP}{FP + TN} \quad (5)
\]

\[
FD_r = \frac{FP}{TP + FP} \quad (6)
\]

The results of the test operations for measuring the success of the proposed technique are given in Figures 5–10 together with the results belonging to some recent studies which also use CoMoFoD dataset [18, 19]. Since the proposed technique aims at detecting only free-form copy-move forgeries, it has been compared only with the other techniques which were developed for detecting free-form copy-move forgeries. Figure 1 presents three examples of the original, mask, forgery, and result images which became subject to the free-form copy-move forgery detection operations carried out by the proposed technique.

Thirunavukkarasu et al. used discrete stationary wavelet transform (DSWT) for copy-move forgery detection. Their technique is based on dividing the suspicious image into overlapping blocks, extracting features from these blocks and then matching the similar blocks by using these features. In their study, which was tested on CoMoFoD dataset, they used accuracy rate in equation (3) and false-positive rate in equation (5) in order to measure the performance of their
Parallelly stored coordinates of source and destination blocks

Computing the center of gravity of source and destination blocks

Creating a copy of the suspicious image

Determining a convergence point so that the center of gravity of the source blocks on the original image will coincide with the center of gravity of the destination blocks on the copy image

Superposing the images according to the convergence point

Moving the copy image on the original image within a determined margin

Applying a two-dimensional median filter to the differential image

Computing the absolute value of difference of the whole intersection area of the superposed images and acquiring a differential image

Selecting the region which intersects with the center of gravity of the source blocks

Determining the moving step with minimum absolute value of difference and superposing the original and copy images as they used to be in this step

Accepting the selected region as the source region on the original image and as the destination region on the copy image

Generating the result mask

Creating the output image

Output image

Figure 4: The process of determination of the similar regions.

Comparison of accuracy rates

Accuracy rate (%)

Postprocessing type

BMP
JPEG (q = 95)
JPEG (q = 90)
JPEG (q = 85)
JPEG (q = 80)
JPEG (q = 75)
JPEG (q = 70)
JPEG (q = 65)
JPEG (q = 60)
JPEG (q = 55)
JPEG (q = 50)

FFCMF
CoMoFoD

Figure 5: Accuracy and verification rates for the copy-move forgery detection experiments which include no postprocessing and JPEG compression. (a) Accuracy rates. (b) Verification rates.
algorithm. In their study, although there are robustness tests against blurring, brightness change, and color reduction, their technique was not tested against JPEG compression [18]. Figures 7(a) and 8(c) show accuracy and false-positive rates of their and our experiments together, respectively. In these figures, their experimental results exist only for the experiments which include only Gaussian filtering as the postprocessing operation.

Lee used Gabor magnitude for detecting copy-move forgery. The author’s technique is also based on dividing the suspicious image into overlapping blocks and matching the similar blocks by using the features extracted from these blocks. During the feature extraction process, the author used histogram of orientated Gabor magnitude (HOGM) descriptors. The technique was tested against JPEG compression with various quality factors, on CoMoFoD dataset. The author used correct detection ratio, that is the same of the verification rate in equation (2), and false-detection ratio in equation (6) for testing their technique [19]. Figures 5(b) and 6(b) show verification and false-detection rates of their and our experiments together, respectively.

Zhou et al. developed a block-based technique and they used color information and its histograms while extracting features from the blocks. They used accuracy ratio, which is
Figure 7: Accuracy and verification rates for the copy-move forgery detection experiments which include only Gaussian filtering and Gaussian filtering followed by JPEG compression. (a) Accuracy rates. (b) Verification rates.

Figure 8: Continued.
the same of the verification rate in equation (2), in order to test their technique. They tested their technique against the free-form copy-move forgeries which include no post-processing, and they state that they found their accuracy ratio higher than 0.95 [20]. Since there is no more information about their success rate on CoMoFoD dataset, we could not make a graphical comparison between their study and ours.
In order to test our technique on the unforged images which contain repeated objects or patterns, we have constructed a dataset and named it Original Images Containing Identical Objects (OICIO) dataset [22]. OICIO dataset consists of 160 unforged images that contain repeated objects or patterns. There are 5 main images in this dataset, and the remaining ones are variations of these main images which were produced by exposing the main images to JPEG compression operations with different quality factors, Gaussian filtering operations and combination of these two operations. We acquired the mentioned main images by using our digital camera. Figure 11 shows three example images from OICIO dataset. During our experiments, our technique did not detect any forgery on 156 of these 160 images while it claimed that 4 of them were forged. The ratio of correct results to total number of test images is 97.50%. This shows that the proposed technique is also robust against deceptive unforged images which contain repeated objects or patterns.

4. Conclusions

First of all, since the other block-based free-form copy-move forgery detection techniques focus on marking the areas...
belonging to the similar blocks but we focus on directly
detecting the whole suspicious area, we need to state that we
have developed a novel and original mathematical approach
for detection of free-form copy-move forgery.

When the test results of the proposed technique is
compared with the test results of the recent studies in the
field of free-form copy-move forgery detection, it can be
obviously seen that the proposed technique has greater
performance and better stability. We present our experi-
mental results on FFCMF and CoMoFoD datasets sepa-
ately. Thirunavukkarasu et al. [18], Lee [19], and Zhou et al.
[20] made various experiments by using the copy-move
forgery images in the CoMoFoD dataset. During the
mentioned experiments, while Thirunavukkarasu et al.
tested their technique against free-form copy-move forgeries
followed by Gaussian blurring attack, Lee made experiments
on the free-form copy-move forgeries which were exposed to
different degrees of JPEG compression attacks, and Zhou
et al. tested their technique against simple free-form copy-
move forgery with no postprocessing. Figures 5–10 indicate
not only the comparison of the proposed technique with the
ones proposed in [18, 19], but also the stability of the
proposed technique against the postprocessing operations
with varying distortion strengths. The reason why the
performance of the proposed technique was tested on two
distinct datasets is both comparing the proposed technique
with the other techniques and proving the stability of the
proposed technique in a more effective way.

Since we have significantly changed the region matching
mechanism of the classical block-based free-form copy-
move forgery detection techniques, our expectation is get-
ning not only high success rates, but also stable performance.
For this reason, we have prepared and made our experiments
so that these experiments clearly show whether our tech-
nique keeps its stability while the postprocessing operations
distort the forged image in increasing strengths. Our ex-
perimental results, which are illustrated in Figures 5–10,
show that unlike the classical block-based free-form copy-
move forgery detection methods, our technique has a stable
performance against varying JPEG quality factors and re-
gardless of the JPEG compression operations with different
quality factors, Gaussian filtering operations and combi-
nation of these two operations, our verification rates and
accuracy rates are quite high and our error rates, false-
positive rates and false-detection rates are satisfactory. We
need to underline that some rises and falls belonging to the
detection results of our technique can be clearly seen because
we did not select the range of the vertical axis between 0%
and 100%. At this point, the most important proof of the
stability is having no sharp and steady rises and falls es-
pecially while the distortion strengths of the postprocessing
operations increase.

We can clearly state that, in addition to having high
success rates, another main contribution of our proposed
technique to the literature is its distortion strength-invariant
performance against postprocessing attacks, which aim at
making the detection of the forgeries harder. Our aim in
making our experiments on the forged images with different
variants of the postprocessing operations although none of
the other studies did so is proving the stability of our free-
form copy-move forgery detection technique. In other
words, our study is not only based on comparing the success
and failure rates with those of the other studies but also
proving its stability on different variants of experiments. We
also need to state that, when we tried using the classical
block-matching technique as in the other studies [18–20], we
could not acquire any satisfactory results as claimed in these
studies.

When Figures 7–10 are examined and evaluated to-
gether, it can be stated that, for the forgery images which are
exposed to both Gaussian filtering and JPEG compression
operations, the order of these two operations affects the
forgery detection success rates; however, the proposed
technique keeps its stability in each condition.

Additionally, we need to state that the experiments
which were performed by using the test images in the OICIO
data set show that our technique is able to distinguish the
copy-move forgery images from the unforged images which
include repeated objects or patterns.

As the future work, since our forgery detection algorithm
detects the forged areas but it cannot determine which of the
matched copy-move region pairs is the original one and

![Figure 11: Three example images from OICIO dataset.](image-url)
which is the copy of the other, we are planning to design a new technique that solves this issue. Besides, it can be a good idea to construct a mechanism for determining the values of $L_{\text{MAX}}$, $E_{\text{MIN}}$, $T_{N}$, $D_{\text{MAX}}$, and $D_{\text{MAX}}$ parameters automatically.

Data Availability

The image data used to support the findings of this study have been deposited in the FFCMF repository (http://emregurbuz.tc/research/imagedatasets/ffcmf/ffcmf.html) and the OICIO repository (http://emregurbuz.tc/research/imagedatasets/oicio/oicio.html).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References


