Received: May 7, 2023. Revised: June 2, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A New Matching Strategy for SIFT Based Copy-Move Forgery Detection

Ibrahim A. Zedan[1]* Mona M. Soliman[1,2] Khaled M. Elsayed[1] Hoda M. Onsi[1]

[1]*Information Technology Department, Faculty of Computers and Artificial Intelligence, Cairo University, Giza-12611, Egypt*
[2] *Member of Scientific Research Group in Egypt, Egypt*
* Corresponding author's Email: i.zedan@fci-cu.edu.eg

Abstract: SIFT-based techniques have achieved satisfying performance in detecting copy-move forgery (CMF). Typically, these techniques find the matched regions of an image and re-examine them using a variety of methods to determine whether CMF has occurred or not. However, these techniques have some shortcomings related to how they handle false matches, which usually occur due to image continuity or self-similarity. First, a spatial distance threshold or segmentation-based methods are commonly utilized to handle image continuity. Second, several external methods along with manually created thresholds are utilized to handle image self-similarity. In this paper, we propose a new matching strategy that is resistant to false matches while reducing reliance on external methods and thus avoiding several thresholds. We model the keypoint as a whole region rather than a single point and employ the intersection over union measure to deal with image continuity. To reduce false matches caused by image self-similarity, we combine the cross-matching test with a modified distance ratio test. This combination takes into account the ability to detect multiple cloning. Moreover, we utilize a support vector machine to learn the threshold(s) needed to decide if CMF has occurred or not. The proposed methodology is evaluated over three challenging datasets: MICC-F600, Coverage, and MICC-F220. On MICC-F600 dataset, our proposed methodology outperforms other state-of-art techniques and achieves high precision of 99.38%, recall of 97.5%, and 98.42% of F1 score. Additionally, the comparative evaluation using Coverage, and MICC-F220 datasets proved the effectiveness of the proposed methodology to handle a variety of attacks.

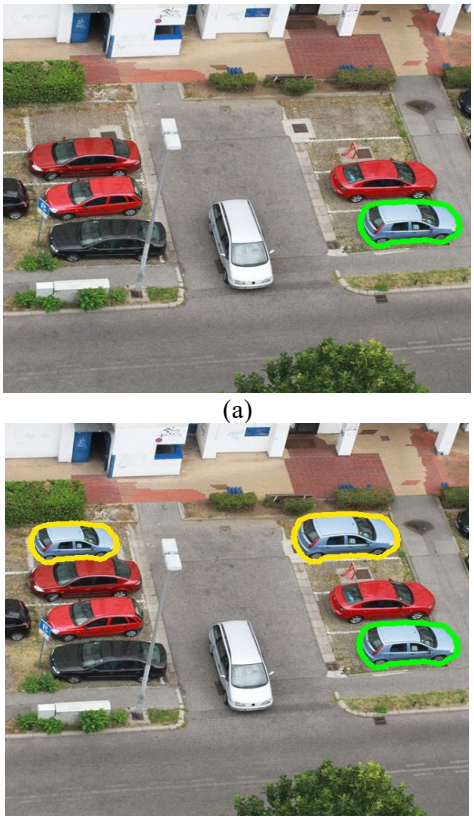
Keywords: Image forensics, Copy-move forgery detection, SIFT, Intersection over union, Distance ratio test, Cross-matching, Support vector machine.

1. Introduction

Images represent a major source of information especially in the forensic evidence field [1]. But, it has become easy for even non-professionals to forge images [2]. Forged images can cause many troubles to individuals and misdirect the public opinion [1]. There are several ways to forge an image, but the cloning forgery is the hardest in its detection and the widespread one [3]. The cloning forgery is also known as the copy-move forgery (CMF) in which a source region(s) in an image is copied and pasted to different location(s) within the image itself [4, 5].

Image cloning usually intends to modify the image's semantics by either repeating or omitting significant objects in the image [6]. Cloned regions often undergo two different sorts of operations: geometric transformations (e.g. rotation, and scaling) and post-processing operations (e.g. noise addition, blurring, and compression) [7]. These operations aim to deceive the eyes and make the forgery detection unattainable [8]. Fig. 1 shows a CMF example in which the source region is duplicated twice using different scaling factors. The source region is marked in green and the forged regions are marked in yellow.

The problem of copy-move forgery detection (CMFD) has recently attracted the attention of many researchers, and their works can be partitioned into several categories: keypoint-based, block-based, deep learning-based, and hybrid techniques [5]. Each category has some advantages and



(a) (b)

Figure. 1 CMF example: (a) authentic image and (b) tampered image

disadvantages.

Both keypoint-based and block-based techniques share the disadvantage of relying on numerous manually created thresholds. On the other hand, deep learning-based techniques are powerfully able to select the suitable thresholds [5]. However, deep learning-based techniques have a number of disadvantages such as its lengthy training process that requires high processing capacity [9]. In addition, a massive number of images with constant dimensions are needed in their learning process [1]. Furthermore, if the deep learning-based system is tested with images of a different nature from what it is learned on, its performance will decrease [10].

Block-based techniques are ineffective in dealing with geometric operations [10]. Additionally, the high complexity of the block-based techniques makes them inefficient as well [11]. In contrast, keypoint-based techniques are time-saving and have the ability to extract local image features being invariant to geometric operations [12].

The standard phases in keypoint-based CMFD techniques are as follows [3, 7, 13]: First, the image undergoes pre-processing in order to improve the features that will be extracted from it. Second, a set of local invariant features (keypoints) along with their descriptors are extracted. Third, matched pairs are obtained by matching the keypoints descriptors. Finally, matched pairs are further verified through the post-processing in order to get rid of wrong matches and determine whether CMF has occurred.

To extract keypoints from an image, there are numerous methods. However, the scale-invariant feature transform (SIFT) and the speeded up robust features (SURF) have experienced the most usage [14]. SURF is characterized by its speed in detecting keypoints, whereas SIFT is distinguished by its strong description ability [1].

Because of the advantages of keypoint-based CMFD techniques, especially those based on SIFT, we employ SIFT in this research paper. Despite these advantages, there is a possibility of wrong matching which is typically brought on by image continuity or self-similarity. Spatially close keypoints are mismatched due to image continuity, while image self-similarity leads to a false matching between original similar parts of the image. In this work, we aim to minimize false matches through our new matching strategy. This work's main contributions can be summed up as follows:

- A keypoint is modeled as an entire region determined by its center location and scale. This modeling facilitates using the intersection over union measure to handle image continuity. Thus, there is no need to use any thresholds or segmentation methods.
- The cross-matching test is combined with a modified distance ratio test in a manner capable of coping with multiple clones. With this combination, it is possible to decrease the number of false matches brought on by image self-similarity while reducing the need to apply external methods that require some thresholds.
- A support vector machine is trained to find the suitable decision boundary that distinguishes images with truly cloned regions from authentic images that might have original similar regions.
- The comparative evaluation proved the superiority of the proposed methodology in detecting CMF in MICC-F600, coverage, and MICC-F220 datasets while being able to deal with various types of attacks.

This research paper is structured as follows: First, we review the related works of keypoint-based CMFD techniques in section 2. Section 3 represents the proposed CMF detection methodology. The experimental analysis of the proposed methodology

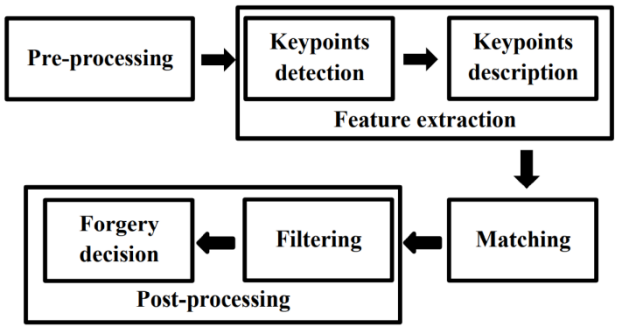


Figure. 2 Standard phases of keypoint-based CMFD techniques

and the conclusion are presented in sections 4 and 5, respectively.

2. Keypoint-based CMFD techniques

Fig. 2 depicts the standard phases of keypoint-based CMFD techniques. A detailed explanation of each phase is provided in the following paragraphs.

2.1 Pre-processing phase

Converting colored images to gray scale is an initial preparation step for images before the feature extraction phase because the image keypoints are often localized within the luminance channel [6, 8, 12]. Additionally, numerous filtering methods are utilized during the pre-processing for reducing noise or boosting the image contrast [13]. In [6] the stationary wavelet transform is applied to the input image for denoising.

Keypoint-based techniques can't recognize the CMF existing in smooth regions since insufficient number of keypoints is located in smooth regions. In order to localize more keypoints in smooth regions by improving the image contrast, the dynamic histogram equalization algorithm and the contrast-limited adaptive histogram equalization method are employed in [3, 15], respectively.

2.2 Feature extraction phase

The method used to localize and describe the image keypoints, as well as the detection threshold(s) applied to detect the keypoints, are two key factors that influence the feature extraction phase [5].

To improve the CMF detection performance, many hybrid feature extraction approaches are presented. In [4] KAZE is combined with SIFT to extract the image keypoints because KAZE has a better ability to extract more keypoints from edges and boundaries of the objects. In [13] keypoints are described by fusing the histogram of the reduced local binary pattern with the SIFT descriptor to improve robustness to pixels variations.

Some works have chosen the keypoints detection threshold(s) differently from the default settings in order to better cover images with keypoints. In [1] a low value is selected for the keypoints detection threshold. In [2, 9] image is divided into non-overlapping superpixels. Two categories of superpixels—smooth and rough—are distinguished using the entropy measure. For each category, a different value of the keypoints detection threshold is chosen to cover the image with enough keypoints in a uniformly distributed manner.

2.3 Matching phase

The matching phase aims to find similar keypoints in an image that signify regions where CMF is initially suspected [5]. Two steps make up the matching phase. First, the nearest neighbours of each keypoint in the image are determined. Then, a matching method is applied to decide whether or not a given keypoint is matched to certain nearest neighbour(s). The distance ratio test and the generalized two nearest neighbor test (g2NN) are common matching methods [4,16].

For a keypoint kp_i, the basis for determining its nearest neighbors is calculating the distances between the descriptors of those neighbors and the descriptor of kp_i. Such distances $DList_{kp_i}$ are sorted in ascending order and referred as follows [7, 9, 13]:

$$DList_{kp_i} = \{d_1, d_2, \dots, d_N\},\ d_1 \le d_2 \le \dots \le d_N \quad (1)$$

The distance ratio test (also known as the two nearest neighbor 2NN test) [17] assumed that a keypoint could only have one match which is the first nearest keypoint and the second nearest keypoint is being noise. The purpose of the 2NN test is to determine whether the first nearest keypoint can be distinguished from the second nearest keypoint. A keypoint kp_i is matched with its first nearest keypoint only if this condition is met:

$$\frac{d_1}{d_2} < T\ ,\ T\epsilon\ [0,1] \quad (2)$$

Where d_1 and d_2 denote the distance from kp_i to its first nearest keypoint and second-nearest keypoint, respectively. Although the 2NN test is very useful in differentiating between truly cloned regions and similar but genuine regions, this test can't deal with multiple cloning.

The g2NN method [18] is suggested to be

effective in handling multiple cloning. For each keypoint kp_i associated with its nearest neighbors and their sorted distances $DList_{kp_i}$, the g2NN method repeats the 2NN test as long as the ratio d_j/d_{j+1} is less than a threshold T (where $1 \geq j < N-1$). Then, all neighbors having $d_j < T \times d_{j+1}$ are matched with kp_i. But in multiple cloning, in many cases, the similar keypoints belonging to the cloned regions have very nearby distances and thus have high distance ratios. So, the g2NN matching method can't match them [3]. To address this drawback, we propose a modified distance ratio test in this research. Unlike the g2NN matching method, our modified distance ratio test does not demand that every match of a particular keypoint should have a distance ratio less than a given threshold.

Speeding up the matching phase and choosing the suitable matching threshold are areas of focus for many researchers. In [19] customized values of the matching threshold and the keypoints detection parameters are selected for each image using the particle swarm optimization (PSO) algorithm. Nevertheless, PSO algorithm can easily fall into local optimum. In contrast, we employ support vector machine in this work to select the optimal matching threshold that encourages a large margin between authentic and forged images. In [9], K-dimensional (KD) trees and superpixels segmentation are employed to expedite the matching phase. Using KD trees, the keypoints that are located in smooth superpixels are matched separately from those in the rough superpixels. But, the superpixels segmentation of an image is a lengthy process and thus the computational complexity of [9] is still high. So, in this work, we present an easier way to divide the keypoints into two clusters and match each cluster separately, depending only on the contrast values of the keypoints.

2.4 Post-processing phase

The post-processing phase aims to filter out erroneous matches as well as deciding if the CMF exists or not. There are two causes of false matches: image continuity or image self-similarity [5].

Image continuity results in some false matches since spatially neighboring keypoints are most likely have similar descriptors and it's possible to match them by mistake [12]. Accordingly, numerous works require a minimum spatial distance between keypoints to be matched through a threshold as in [9, 19]. However, such spatial distance threshold assumes a minimum spatial separation between cloned regions which is unknown. Thus, false negatives could occur if this threshold is chosen incorrectly.

Different approach for handling image continuity is introduced in [4, 14]. In [14] two keypoints could be matched if they are located in separate superpixels in the image. In [4] image is segmented into bounding boxes or objects. Keypoints within the same bounding box can't be matched. An image is considered as tampered when the number of matches between two bounding boxes exceeds a certain threshold. The drawback of this approach is that the CMF will be undetectable if the cloned regions fall into the same image segment/bounding box.

In this work, we did not use a spatial distance threshold or even a segmentation method, and thus we avoided their drawbacks that were mentioned earlier. Instead, we managed image continuity more effectively by relying on the keypoints properties and the intersection over union measure.

Image self-similarity makes it possible for an image to contain genuine but similar regions that could be mismatched [16]. To handle image self-similarity and verify the matched pairs, there are various methods. These verification methods are either based on the spatial density, the geometric consistency, or the pixel-level correlation [3].

2.4.1. Spatial density-based methods

Matched pairs resulting from CMF are typically concentrated in specific patches of the image. In other words, wrong matches that arise from similar but authentic image regions are relatively dispersed [5]. Motivated by this idea, the hierarchical agglomerative clustering (HAC) is performed on the matched pairs based on their location coordinates in [1, 6, 7, 12, 16]. If there is a little number of matches in certain cluster, such cluster will be eliminated because it most likely contains erroneous matches [1, 16]. An image is treated as tampered if there are two or more clusters with three or more matching pairs [6]. Although many CMFD methods employ the HAC algorithm, it should be emphasised that it is susceptible to noise and outliers.

As an alternative to HAC, The density-based spatial clustering of applications with noise algorithm is utilized in [3, 8, 15]. Similarly, in [9] wrong matches are filtered out using superpixels segmentation. If there is a little number of matched pairs in certain superpixel, these matched pairs are ignored [9].

Spatial density-based methods are usually based on clustering or segmentation algorithms. They thus experience high complexity. In addition, selecting a

clustering or segmentation algorithm and its parameters that are appropriate for all images is challenging.

2.4.2. Geometric consistency-based methods

Matched pairs are usually verified whether they are really belong to cloned regions by finding the geometric transformation between them and measuring their consistency with such estimated geometric transformation [5]. Thus, matched pairs are categorized as inliers or outliers depending on how well they conform to the estimated geometric transformation.

Given the presence of mismatches, random sample consensus (RANSAC) is the most effective algorithm for estimating the geometric transformation between matched pairs. Also, RANSAC is widely utilized to exclude the incorrectly matched pairs as in [6, 12, 15, 16]. The success of RANSAC in estimating the geometric transformation between matched pairs is sufficient to declare the image as tampered as in [7]. However, there are few matched pairs in the case of small or smooth cloned regions. As a result, the CMF becomes undetectable because RANSAC probably fail to estimate the geometric transformation corresponding to these few matched pairs.

2.4.3. Correlation-based methods

Cloned regions in an image often exhibit a higher pixel-based correlation than similar but real regions [5]. Therefore, it is usual to evaluate the correlation coefficient between the corresponding image regions of matched pairs after geometric transformation. Then, compare it to a predetermined correlation threshold to indicate whether these matched pairs belong to cloned regions or not [7, 13, 19]. In order to avoid confusion between cloned regions and similar but real regions, the correlation threshold must be set at an adequate value. In [19] the correlation map average is utilized to customize the correlation threshold value.

In this research, we avoid using all these verification methods, and thus many thresholds have been eliminated. Instead, we were able to handle image self-similarity efficiently by including the cross-matching test in our new matching strategy and learning the suitable matching threshold using support vector machine.

3. Proposed methodology

Fig. 3 depicts the entire framework of the proposed methodology. Each process of the proposed methodology is covered in detail in this section.

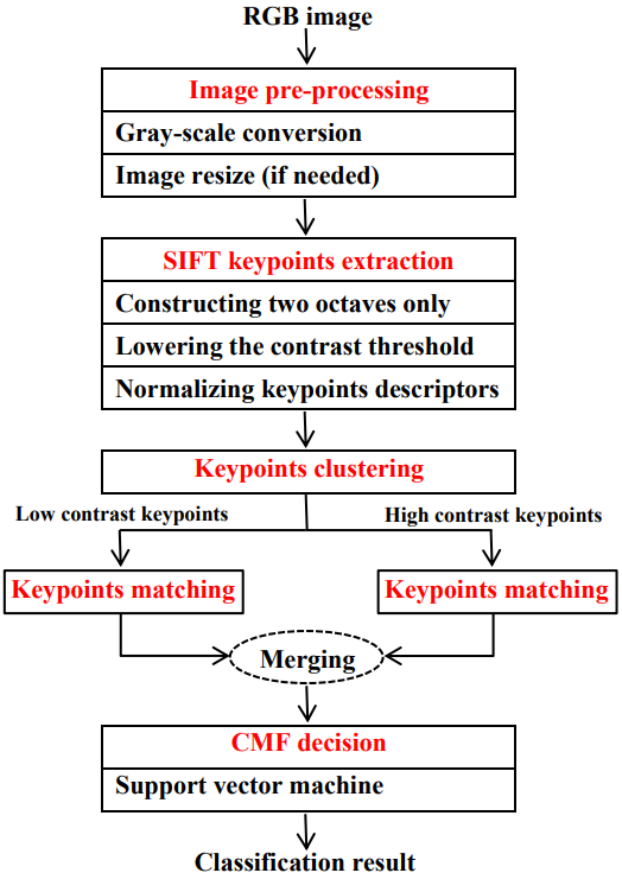


Figure. 3 Proposed methodology framework

3.1 Image pre-processing

We perform minimal pre-processing on the input image before extracting the SIFT keypoints from it. First, RGB images are converted to grayscale. Second, high resolution images having dimensions greater than 1500 pixels are resized by half in each dimension to reduce the processing time.

3.2 SIFT keypoints extraction

After pre-processing the input image, we extract the SIFT keypoints along with their descriptors to represent the image. The following is a concise summary of the SIFT algorithm [17]:

- The SIFT detector generates a number of octaves. From one octave to the next one, the image dimensions are decreased by half. Each octave is a multi-scale representation for an image with fixed image dimensions. More specifically, an octave is a set of increasingly gaussian-smoothed layers.
- The difference of the gaussians (DoG) is

derived from each octave.
- SIFT keypoints are detected by locating the points corresponding to any local extrema in the DoG space whose contrast value is greater than a contrast threshold.
- For each SIFT keypoint, a descriptor of length 128 is formed by describing the keypoint local neighborhood.

In the following paragraphs, we concentrate on some crucial steps that we follow during the SIFT keypoints extraction process.

3.2.1. Constructing two octaves only

In this work, we solely extract keypoints from the first two octaves because they have higher image resolution, more details, and the majority of keypoints are extracted from them. Octaves above the second octave have a low resolution and are extremely blurry. Their extracted keypoints are few and each keypoint represents a big region in the original image. Thus, their keypoints are less useful and matching them is probably misleading.

3.2.2. Lowering the contrast threshold

One of the most crucial thresholds in the SIFT detector is the contrast threshold. It is known that employing the contrast threshold with its default value causes a scarcity of keypoints in the smooth regions, and thus the inability to detect the CMF in these regions. However, it must be taken into account that applying a zero contrast threshold would generate keypoints in extremely smooth background regions of the image. This poses a problem as these keypoints can be easily matched by mistake.

Through our experiments, we chose the contrast threshold value to be 0.005 (for the SIFT implementation provided by opencv-python). This value prevents keypoints from being located in extremely smooth background regions while ensuring that the image is adequately covered with keypoints.

3.2.3. Normalizing keypoints descriptors

It's essential to normalize the keypoints descriptors to ensure invariance to variations in illumination and to prevent high value features from dominating other features. We utilize the L1 norm for normalizing the keypoints descriptors.

$$f_i = \frac{f_i^{'}}{\sum_{l=1}^{128}|f_i^{'}(l)|} \quad (3)$$

Where $f_i{}', f_i$ denote the original and the normalized descriptors of keypoint kp_i respectively.

3.3 Keypoints clustering

Image regions of different roughness degree can't be matched, and calculating the distance between them is a waste of time. Some researchers applied this idea by dividing the image into two parts with different roughness degrees using an image segmentation method. Then, they applied the matching phase in each part separately. However, we propose a more straightforward alternative to implement the same idea based on the keypoints properties and without employing any image segmentation method.

The contrast value of a keypoint reflects the roughness degree of the image region represented by that keypoint. Based on the contrast values of the keypoints, we employ a threshold to divide the keypoints into two clusters, one is smooth and the other is rough. The value of this threshold is the same value of the contrast threshold in the SIFT detector's default settings before we decrease it.

3.4 Keypoints matching

After clustering the keypoints into two clusters based on the keypoints contrast values, we carry out the keypoints matching in each cluster separately as shown in Fig. 4. First, we calculate the net-similarities between keypoints in a way that considers handling image continuity. Second, net-similarities are converted into distances to find initial matches using a modified distance ratio test. Finally, during the refined matching step, we filter-out the initial matches using the cross-matching test. Each step is fully explained in the paragraphs that follow.

3.4.1. Net-similarity calculation

For each keypoint kp_i, we calculate the distance between it and the rest of keypoints in the same cluster. For high dimensional descriptors like SIFT, manhattan distance (L1 distance) is preferred. Thus, we define the distance function $D(kp_i, kp_j)$ that compares two keypoints kp_i, kp_j as the L1 distance between their normalized descriptors (f_i, f_j). Due to normalizing the keypoints descriptors, the distance between any two keypoints can't exceed the value of 2. Mathematically, we have

$$D(kp_i, kp_j) = \sum_{l=1}^{128}|f_i(l) - f_j(l)| \quad (4)$$

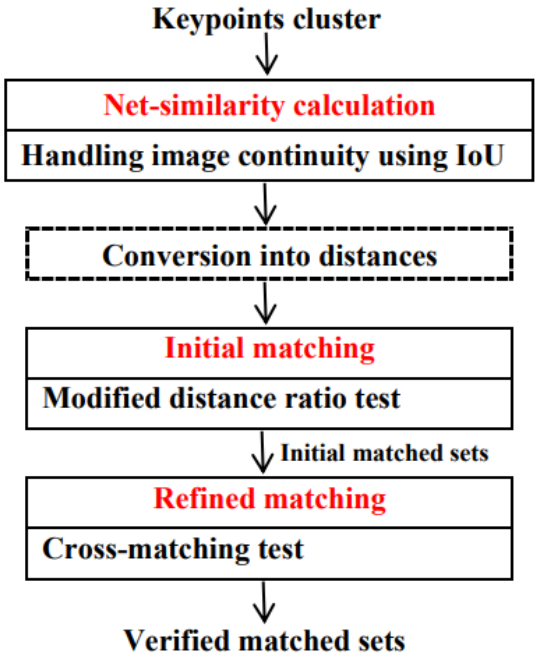


Figure. 4 Steps of the keypoints matching process

The absolute similarity $S`(kp_i, kp_j)$ between two keypoints kp_i, kp_j is defined as the complement of their distance. Formally,

$$S`(kp_i, kp_j) = \left(1 - \frac{D(kp_i, kp_j)}{2}\right) \quad (5)$$

Due to image continuity, neighboring keypoints are highly similar and should be avoided in the matching process. Most of the previous works measure the spatial separation between keypoints and depend on threshold(s) to avoid matching of neighboring keypoints. Furthermore, only the keypoints center locations are taken into account while deciding neighboring keypoints.

To deal with image continuity, we employ the overlapping rather than the spatial separation between keypoints for two reasons. First, keypoints are not just single points but whole regions that are defined by the keypoints properties. Second, the amount of overlapping makes better use of the keypoint properties. Through this new modeling, we take into consideration the keypoints scales as well as its center locations while deciding if two keypoints are neighboring or not. Moreover, we avoid using any threshold.

When calculating the net-similarity between two keypoints, their overlap is embedded and taken into consideration. More specifically, our idea is based on suppressing the absolute similarity score of overlapped keypoints by the amount of overlapping. We utilize the intersection over union (IoU) to measure the amount of overlapping between keypoints. We define the net-similarity $S(kp_i, kp_j)$ between two keypoints kp_i, kp_j as their absolute similarity multiplied by the complement of their IoU. Formally,

$$S(kp_i, kp_j) = S`(kp_i, kp_j) \times \left(1 - IoU(kp_i, kp_j)\right) \quad (6)$$

For simplicity, we approximate a SIFT keypoint kp_i as a square around its center location (x_i, y_i) of length $2r_i$ where r_i is positively correlated to the keypoint scale σ_i. The intersection over union $IoU(kp_i, kp_j)$ between two keypoints kp_i, and kp_j is calculated as

$$IoU(kp_i, kp_j) = \frac{I(kp_i, kp_j)}{A(kp_i) + A(kp_j) - I(kp_i, kp_j)} \quad (7)$$

Where $A(kp_i)$ and $A(kp_j)$ denote the area of kp_i and kp_j, respectively. $I(kp_i, kp_j)$ denotes the intersection area between kp_i, kp_j.

3.4.2. Initial matching

Let's assume that the cluster we are now matching its keypoints has N keypoints. For each keypoint kp_i in the current cluster, we obtain its initial matched keypoints using a modified distance ratio test as follows: First, we calculate the net-similarities between kp_i and the remaining N-1 keypoints, as previously explained. Second, in order to perform our modified distance ratio test, we convert the net-similarities into distances again (where $distance = 1 - net\text{-}similarity$). Third, we arrange these distances in ascending order and denote them by $DList_{kp_i}$, as in Eq. (1). Thus, the nearest neighbors of kp_i are determined and denoted as follows:

$$NNList_{kp_i} = \{NN^1_{kp_i}, NN^2_{kp_i}, \dots, NN^{N-1}_{kp_i}\} \quad (8)$$

Where $NN^j_{kp_i}$ denote the j^{th} nearest neighbor of keypoint kp_i. Fourth, we calculate the ratio between each two consecutive distances in $DList_{kp_i}$. Assuming that cloning doesn't occur more than five times in an image, we only compute the first five ratios. Such ratios are denoted as $RatioList_{kp_i}$. Formally,

$$RatioList_{kp_i} = \{ratio_1, ratio_2, \dots, ratio_5\}, ratio_j = \frac{d_j}{d_{j+1}} \quad (9)$$

Finally, we choose the minimum ratio from $RatioList_{kp_i}$. If the minimum ratio is found at $ratio_J$, the first J nearest neighbors are regarded as initial matches of kp_i. Define S_{IM} as the set of initial matched keypoints. In case of single cloning, S_{IM} should contain only two keypoints, while it contains more keypoints in case of multiple cloning. Namely,

$$S_{IM} = kp_i \, U \, \{NN_{kp_i}^k : 1 \leq k \leq J\} \quad (10)$$

The reason for choosing the minimum ratio is that the distance values before and after it differs significantly. Such significant difference probably reflects a switching from candidate matches to noisy matches of a certain keypoint.

Our modified distance ratio test differs from the basic distance ratio test as it's able to deal with multiple cloning. Additionally, it differs from the g2NN method as it isn't required that each initial match of certain keypoint should have a distance ratio below some threshold which doesn't always happen in case of multiple cloning.

3.4.3. Refined matching

According to the cross-matching test, two keypoints are cross-matched if each is being the best match (the first nearest) of the other. Although the cross-matching test is one method for reducing false matches, it is rarely utilized because it is ineffective in case of multiple cloning. However in this research, we incorporate the cross-matching idea in a way that addresses multiple cloning.

After performing the initial matching step, each keypoint yields a set of initial matched keypoints. Some of these initial matched sets are repeated several times. In other words, the same initial matched set could be obtained through more than one keypoint which is considered a form of cross-matching. Depending on this phenomenon and using the cross-matching test, we can eliminate some initial matched sets, while verifying others during the refined matching step. More specifically, a set of initial matched keypoints S_{IM} is verified if all its member keypoints yield the same set. In other words, a set S_{IM} is verified if we get it a number of times equal to its cardinality.

We use a distance feature and a distance ratio feature to describe each verified matched set. The distance feature is determined by averaging the distances between the keypoints of the matched set. It should be noted that the occurrences of the same matched set are usually detected with slightly different minimum distance ratios. Thus, the distance ratio feature of a matched set is computed by averaging the minimum distance ratios associated with its occurrences.

3.5 CMF decision

We perform the following steps in order to determine whether an input image contains CMF or not: First, we merge the verified matched sets that were acquired from the high and low contrast keypoints. Second, we select three verified matched sets with minimum distance feature. Third, image is represented by the features average of its selected matched sets. Using multiple matched sets and averaging their features enabled us to achieve resistance to outliers. Finally, we use the distance average and the distance ratio average to train a support vector machine, which will determine whether or not CMF has happened. The technical details that we followed for training the support vector machine are:

- A small regularization parameter (C=1) is utilized to promote wide margin, and thus more general decision boundary.
- The radial basis function (RBF) is utilized as a kernel function because of its excellent discrimination power and widespread.

4. Experimental results

In this section, we present the results achieved by the proposed CMFD methodology. The utilized datasets and evaluation metrics are explained in detail. Test results are compared with other state-of-art CMFD systems. In addition, a visual analysis of our proposed matching strategy is provided to confirm its effectiveness against a variety of attacks and challenges.

4.1 Test datasets

Three challenging datasets are utilized to assess the competitive performance of the proposed CMFD methodology: MICC-F600 dataset [18], coverage dataset [20], and MICC-F220 [18].

MICC-F600 [18] dataset contains 440 authentic images and 160 tampered images. This dataset consists of images of JPEG and PNG file formats. The dimensions of these images range in size from 800 × 533 to 3888 × 2592 pixels. Cloned regions within the tampered images are arbitrary shaped and vary in size. Tampered images can be divided into four categories. Each category contains 40 images and has a separate tampering attack. The first category consists of tampered images with single

plain cloning, while the second category includes samples of multiple cloning. The third category contains images with rotated cloned regions, while images in which the cloned regions have been scaled and rotated are found in the fourth category.

Coverage [20] dataset consists of 100 authentic images and 100 tampered images with single cloning. The dataset's images are saved in TIFF format and have an average size of 400 × 486 pixels. Dealing with this dataset is a difficult challenge because its authentic images intensively introduce similar but genuine objects. Obviously, this dataset applies rotation and/or scaling to cloned regions as tampering attacks. Additionally, in some images, cloned regions are subjected to tampering attacks such as illumination change and free-form transform.

MICC-F220 [18] dataset contains 110 authentic images and 110 tampered images with single cloning. The dataset's images are saved in JPEG format and range in size from 722 × 480 to 800 × 600 pixels. Ten geometric transformations with different rotations and scaling factors were used to construct this dataset. The cloned regions are either square or rectangular in shape and account for approximately 1.2% of the size of the tampered images.

4.2 Evaluation metrics

To measure the performance of the proposed CMFD methodology and to perform comparative analysis, we utilize the following standard evaluation metrics [1]:

$$precision\ (P) = \frac{T_P}{T_P+F_P} \quad (11)$$

$$recall\ (R) = \frac{T_P}{T_P+F_N} \quad (12)$$

$$false\ positive\ rate\ (FPR) = \frac{F_P}{F_P+T_N} \quad (13)$$

$$F1\ score\ (F1) = \frac{2\times P\times R}{P+R} \quad (14)$$

Where T_P represents the number of tampered images correctly recognized as tampered. F_P represents the number of authentic images erroneously recognized as tampered. T_N represents the number of authentic images correctly recognized as authentic. F_N represents the number of tampered images erroneously recognized as authentic. A better CMFD performance is indicated by higher values of *P, R, F1*, and by a low value of *FPR*.

Table 1. Comparative evaluation on MICC_F600 dataset

Method	P	R	FPR	F1
Elaskily et al. (2019) [12]	78.02	91.48	9.37	84.22
Sunitha et al. (2022) [1]	91.74	96.8	3.18	94.2
Harshith et al. (2023) [11]	73.71	89.37	11.59	80.79
Proposed	99.38	97.5	0.23	98.42

Table 2. Comparative evaluation on coverage dataset

Method	P	R	FPR	F1
Park et al. (2020) [13]	64.46	78.0	43.0	70.59
Yue et al. (2022) [10]	58.3	91.0	65.0	71.1
Proposed	84.92	76.0	14.0	80.05

4.3 Comparative evaluation

Especially when the utilized datasets are limited in size, it is important to conduct a reliable evaluation of the support vector machine that is responsible for the CMF decision. Therefore, K-fold cross validation (where K=2) is utilized to prevent over-fitting, and provide reliable performance evaluation with low variance.

For MICC_F600 dataset, we compare the performance of the proposed methodology with the following methods: (Elaskily et al., 2019), (Sunitha et al., 2022), and (Harshith et al., 2023). Such comparative evaluation on MICC_F600 dataset is summarised in Table 1 based on precision, recall, false positive rate, and F1 score. Table 1 shows that our proposed methodology achieves a precision of 99.38%, recall of 97.5%, false positive rate of 0.23% and 98.42% of F1 score. Therefore, in terms of all the evaluation metrics, our proposed methodology outperformed all the methods compared here.

Table 2 summarizes the comparative evaluation on Coverage dataset where the proposed methodology's performance is compared with the following methods: (Park et al., 2020), and (Yue et al., 2022). On coverage dataset, our proposed methodology achieves a precision of 84.92%, recall of 76%, false positive rate of 14% and 80.05% of F1 score. Although the methods of (Park et al., 2020), and (Yue et al., 2022) achieve higher recall rates than us, our proposed methodology is superior to

them in terms of precision and false positive rate. In addition, the proposed methodology generally achieves the best performance since it has the highest F1 score which is a generic evaluation metric that incorporates precision and recall.

On coverage dataset, neither the proposed methodology nor the other methods have achieved comparable performance results to those obtained on MICC_F600 dataset. To investigate this phenomenon, Fig. 5 compares between the feature spaces of MICC_F600 and coverage datasets. The feature space of one partition of the MICC_F600 dataset is shown in Fig. 5(a), while the feature space of one partition of the Coverage dataset is shown in Fig. 5(b). Where, authentic and tampered images are represented by green and red dots, respectively. In addition, the decision boundary that separates tampered images from authentic images is marked in gray. From Fig. 5, it is evident that authentic and tampered images overlap more in the coverage dataset's feature space. Naturally, this makes it difficult to distinguish between them, which results in poor results. More specifically, there are two reasons for these poor results on coverage dataset: the dataset's challenging attacks which were previously mentioned, and the low resolution of the dataset's images.

4.4 Detection results under attacks

One advantage of the MICC_F220 dataset is that it applies different geometric transformations to tampered images with varying attack levels. Therefore, we utilize this dataset to prove our robustness against geometric transformations. Furthermore, we provide an in-depth evaluation of our methodology and compare it to the work of [4] on the MICC_F220 dataset.

To assess the proposed methodology's performance against rotation, forty-four tampered images and 11 authentic images were examined. Under rotation operation, we achieved a better F1 score, which is 98.88%, while the work of [4] achieved 95.34% of F1 score. Thirty-three tampered images and 11 authentic images were examined to determine how well the proposed methodology performed against scaling. We achieved a higher F1 score under the scaling operation, which is 96.97%, as opposed to the work of [4] achieved F1 score of 95.37%.

Under combined rotation and scaling, twenty-two tampered images and 11 authentic images were examined. In this experiment, the work of [4] outperformed us and achieved F1 score of 93%, while our proposed methodology achieved 90.47% of the F1 score. But in general, our proposed methodology is superior to the work of [4], as we achieves better F1 score on the whole dataset, which is 95.49%, while the work of [4] has F1 score of 94.87% on the whole MICC_F220 dataset. Table 3 details the comparative evaluation between our proposed methodology and the work of [4] on MICC_F220 dataset, including the results of each geometric attack as well as the aggregate results.

Table 3. In-depth comparative evaluation on MICC_F220 dataset

Transform	Method	P	R	F1
Rotation	[4]	97.61	93.18	95.34
	Proposed	97.78	100	98.88
Scaling	[4]	96.87	93.93	95.37
	Proposed	96.97	96.97	96.97
Rotation + Scaling	[4]	95.2	90.9	93
	Proposed	95	86.36	90.47
Overall	[4]	96.22	93.57	94.87
	Proposed	94.64	96.36	95.49

In order to visualize the ability of our new matching strategy to deal with CMF with different attacks and challenging conditions, we generate a gray-scale similarity map for each input image. This similarity map is constructed from the verified matched keypoints that are resulted from the matching process. The construction of the similarity map is based on our ability to localize any keypoint within the image as a complete region given the keypoint properties.

As mentioned earlier, each verified matched set of keypoints is associated with a distance feature and a distance ratio feature. However, since the distance ratio is more discriminative feature, we utilize it to build the similarity map. The similarity map is first initialized as a black image of the same size as the input image. Then, for each matched keypoints, we modify its corresponding regions within the similarity map with the complement of their distance ratio. The complement is used because similarity increases as the distance ratio decreases. We finally multiply the similarity map by a value of 255 to display it as a gray-scale image because the distance ratio's value can't exceed 1.

Fig. 6 shows our matching strategy's effectiveness on four challenging tampered images from MICC_F600 dataset. The first row of Fig. 6 shows the tampered images. In Figs. 6(a) and 6(b), tampered images contain single cloning where both

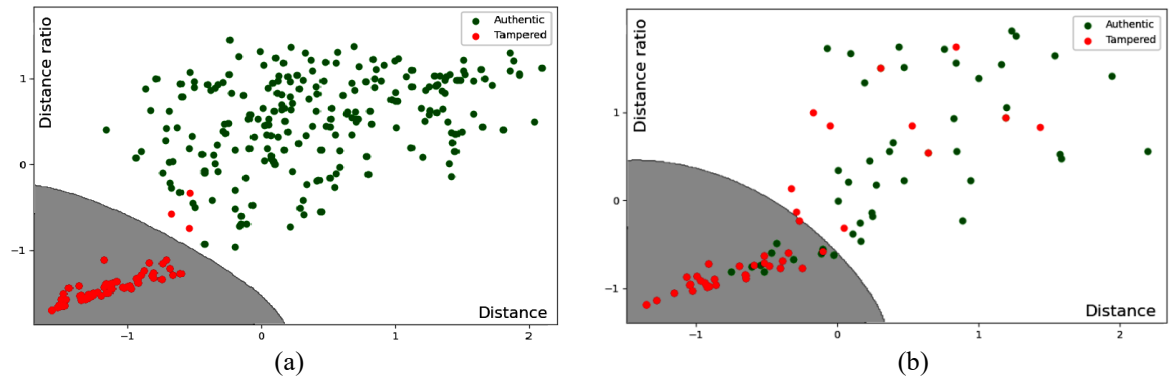


Figure. 5 Feature spaces comparison (a) MICC-F600 dataset and (b) Coverage dataset

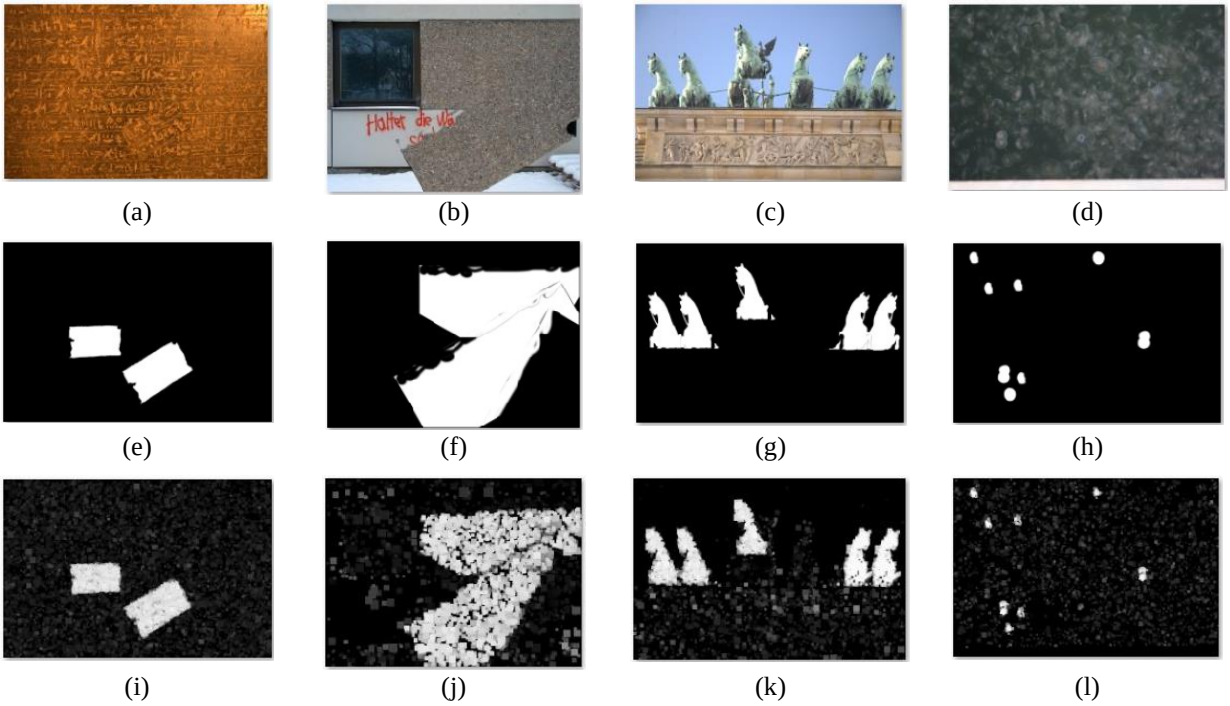


Figure. 6 Detection results of MICC-F600 dataset under different attacks: (a)-(d) the tampered images, (e)-(h) the ground truth localization masks, and (i)-(l) the obtained similarity maps

rotation and scaling are taken place. However, Fig. 6(b) depicts a case in which the cloned regions overlap with each other. In Figs. 6(c) and 6(d), tampered images contain multiple cloning. The example shown in Fig. 6(d) shows a smooth tampered image with relatively small cloned regions. The ground truth localization masks of the cloned regions and the similarity maps obtained from our matching strategy are shown in the second and third rows of Fig. 6. By comparing the obtained similarity maps with the ground truth localization masks, it's clear that the verified matched keypoints localize the cloned regions effectively.

Fig. 7 shows the obtained similarity maps of two pairs of images from Coverage dataset. The first and second columns in Fig. 7 represent the authentic images and its obtained similarity maps. The tampered images and its obtained similarity maps are shown in the third and fourth columns. In each tampered image, the authentic source region is marked in green, while its duplicated forged region is marked in red. The forged region in Fig. 7(c) is a scaled-down copy of its source region, whereas in Fig. 7(g), an illumination change has been applied to the forged region. The similarity maps shown in Fig. 7 demonstrate the ability of the proposed matching strategy to differentiate between authentic images that contain highly similar regions and forged images that really contain cloned regions.

By viewing the similarity maps we obtained in Figs. 6 and 7, it is clear that the proposed matching strategy is able to deal with many challenges such as: multiple cloning, geometric transforms, similar but genuine regions, illumination change, and

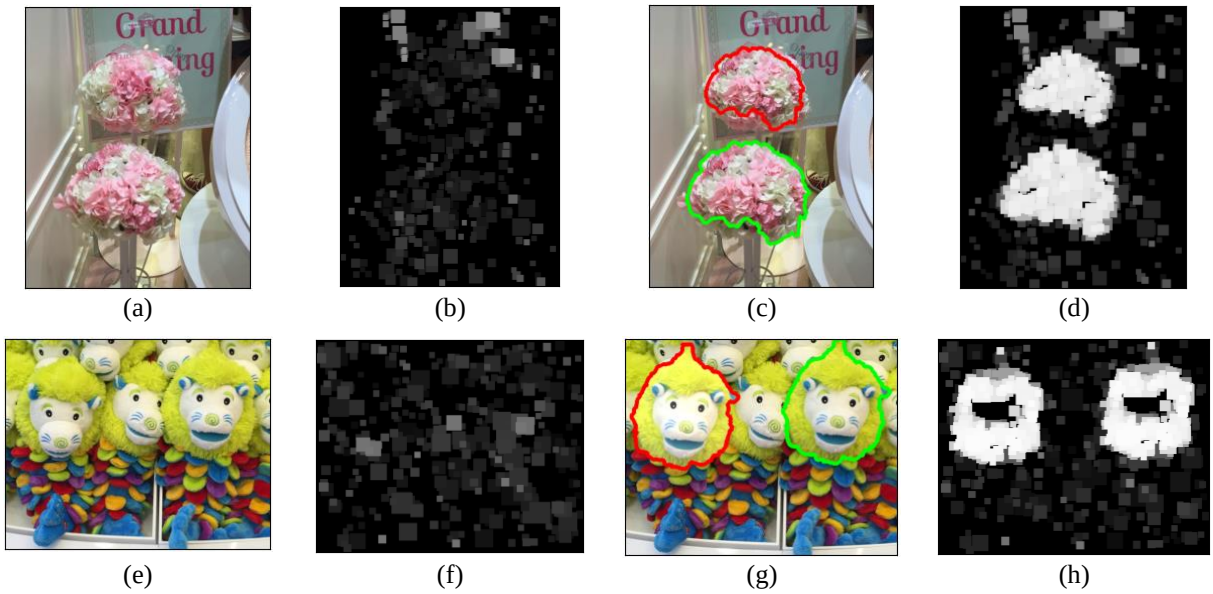


Figure. 7 Detection results of coverage dataset under different attacks: (a),(e) the authentic images; (b),(f) the obtained similarity maps from the authentic images; (c),(g) the tampered images; and (d),(h) the obtained similarity maps from the tampered images

overlap between cloned regions. It is worth noting that neither morphological operations nor even a thresholding operation were applied to the resulting similarity maps. This explains the discontinuity observed in the similarity maps which is due to the fact that keypoints are usually cover an image in a scattered manner. Particularly, in the similarity map shown in Fig. 7(h), a hole is observed inside the cloned region(s). This hole is caused by the lack of keypoints in its corresponding image region, which has undergone an illumination change and has become very smooth.

5. Conclusion

The proposed CMF detection methodology has achieved promising results while being able to cope with geometric transforms, multiple cloning, and intersection between cloned regions. More specifically, we achieved F1 score of 98.42% and 95.49% on MICC-F600 dataset and MICC-F220 dataset, respectively. The false positive rate is reduced and thus the overall F1 score is improved by two issues. First, instead of using fixed matching thresholds, they are more appropriately chosen through a machine learning process. Secondly, the cross-matching test is integrated during the matching process, and then the matching process became less error-producing. In addition, the image continuity is handled efficiently without the need for any thresholds by maximizing the utilization of the keypoints properties through the IoU measure. All these procedures contributed to the lack of the need for many external methods, such as the spatial density-based methods, the geometric consistency-based methods and the correlation-based methods, which were commonly used in literature to enhance performance.

It is worth noting that the proposed methodology has a reasonable ability to differentiate between truly cloned regions and similar but genuine regions. This is evident by our achievement of F1 score of 80.05% on the Coverage dataset. However, this performance needs improvement. More specifically, in future works, we plan to improve the ability to differentiate truly cloned regions from similar but genuine regions, especially in the case of low-resolution images.

Notations

Notation	Description
N	Number of keypoints
kp_i	i^{th} keypoint
(x_i, y_i)	Center location of keypoint kp_i
σ_i	Scale of keypoint kp_i
$A(kp_i)$	Area of keypoint kp_i
$f_i^{'}$	Original descriptor of keypoint kp_i before normalization
f_i	Normalized descriptor of keypoint kp_i
$D(kp_i, kp_j)$	Distance between two keypoints kp_i , and kp_j
$S^{`}(kp_i, kp_j)$	Absolute similarity between kp_i, kp_j
$S(kp_i, kp_j)$	Net-similarity between kp_i, kp_j
$I(kp_i, kp_j)$	Intersection area between kp_i, kp_j
$IoU(kp_i, kp_j)$	Intersection over union between kp_i, kp_j

$NN_{kp_i}^{j}$	j^{th} nearest neighbor of keypoint kp_i
$NNList_{kp_i}$	List of nearest neighbors of kp_i
d_j	Distance between a keypoint and its j^{th} nearest neighbor
$DList_{kp_i}$	Sorted distances between kp_i and its nearest neighbors
$ratio_j$	Ratio between each two consecutive distances in $DList_{kp_i}$
$RatioList_{kp_i}$	First five distance ratios of keypoint kp_i
S_{IM}	Set of initial matched keypoints
T	Matching threshold

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The paper conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, have been done by 1st author. The supervision and project administration have been done by 2nd, 3rd and 4th author.

References

[1] K. Sunitha, A. N. Krishna, and B. G. Prasad, "Copy-move tampering detection using keypoint based hybrid feature extraction and improved transformation model", *Applied Intelligence*, Vol. 52, pp. 15405–15416, 2022.

[2] M. M. A. Alhaidery and A. H. Taherinia, "A passive image forensic scheme based on an adaptive and hybrid techniques", *Multimedia Tools and Applications*, Vol. 81, pp. 12681–12699, 2022.

[3] A. Hegazi, A. Taha, and M. M. Selim, "An improved copy-move forgery detection based on density-based clustering and guaranteed outlier removal", *Journal of King Saud University - Computer and Information Sciences*, Vol. 33, No. 9, pp. 1055–1063, 2021.

[4] N. Kumar and T. Meenpal, "Salient keypoint-based copy – move image forgery detection", *Australian Journal of Forensic Sciences*, pp. 1–24, 2022.

[5] I. A. Zedan, M. M. Soliman, K. M. Elsayed, and H. M. Onsi, "Copy Move Forgery Detection Techniques : A Comprehensive Survey of Challenges and Future Directions", *International Journal of Advanced Computer Science and Applications*, Vol. 12, No. 7, pp. 248–264, 2021.

[6] T. Das, R. Hasan, M. R. Azam, and J. Uddin, "A Robust Method for Detecting Copy-Move Image Forgery Using Stationary Wavelet Transform and Scale Invariant Feature Transform", In: *Proc. of International Conf. on Computer, Communication, Chemical, Material and Electronic Engineering*, Rajshahi, Bangladesh, pp. 1–4, 2018.

[7] A. Hegazi, A. Taha, and M. M. Selim, "Copy-Move Forgery Detection Based on Automatic Threshold Estimation", *International Journal of Sociotechnology and Knowledge Development*, Vol. 12, No. 1, pp. 1–23, 2020.

[8] M. F. M. Mursi, M. M. Salama, and M. H. Habeb, "An Improved SIFT-PCA-Based Copy-Move Image Forgery Detection Method", *International Journal of Advanced Research in Computer Science and Electronics Engineering*, Vol. 6, No. 3, pp. 23–28, 2017.

[9] C. Wang, Z. H. I. Zhang, Q. LI, and X. Zhou, "An Image Copy-Move Forgery Detection Method Based on SURF and PCET", *IEEE Access*, Vol. 7, pp. 170032–170047, 2019.

[10] G. Yue, Q. Duan, R. Liu, W. Peng, Y. Liao, and J. Liu, "SMDAF : A novel keypoint based method for copy-move forgery detection", *IET Image Processing*, Vol. 16, No. 13, pp. 3589–3602, 2022.

[11] N. B. S. P. S. Harshith, D. Sindhuja, C. R. Reddy, A. Deepthi, and G. Gopakumar, "Copy-Move Forgery Detection Using K-Means and Hu ' s Invariant Moments", In: *Proc. of International Conf. on Innovative Computing and Communications*, pp. 611–619, 2023.

[12] M. A. Elaskily, H. K. Aslan, M. M. Dessouky, F. E. A. E. Samie, O. S. Faragallah, and O. A. Elshakankiry, "Enhanced Filter-based SIFT Apprroach for Copy-Move Forgery Detection", *Menoufia Journal of Electronic Engineering Research*, Vol. 28, No. 1, pp. 159–182, 2019.

[13] J. Y. Park, T. A. Kang, Y. H. Moon, and I. K. Eom, "Copy-Move Forgery Detection Using Scale Invariant Feature and Reduced Local Binary Pattern Histogram", *Symmetry*, Vol. 12, No. 4, pp. 1–16, 2020.

[14] C. Lin, W. Lu, W. Sun, J. Zeng, T. Xu, and J. H. Lai, "Region duplication detection based on image segmentation and keypoint contexts", *Multimedia Tools and Applications*, Vol. 77, pp. 14241–14258, 2018.

[15] M. Bilal, H. A. Habib, Z. Mehmood, R. M. Yousaf, T. Saba, and A. Rehman, "A robust technique for copy-move forgery detection from small and extremely smooth tampered regions based on the DHE-SURF features and

mDBSCAN clustering", *Australian Journal of Forensic Sciences*, Vol. 53, No. 4, pp. 459–482, 2021.

[16] K. B. Meena and V. Tyagi, "A hybrid copy-move image forgery detection technique based on Fourier-Mellin and scale invariant feature transforms", *Multimedia Tools and Applications*, Vol. 79, pp. 8197–8212, 2020.

[17] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, Vol. 60, No. 2, pp. 91–110, 2004.

[18] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A SIFT-Based Forensic Method for Copy – Move Attack Detection and Transformation Recovery", *IEEE Transactions on Information Forensics and Security*, Vol. 6, No. 3, pp. 1099–1110, 2011.

[19] F. Zhao, W. Shi, B. Qin, and B. Liang, "A Copy-Move Forgery Detection Scheme with Improved Clone Region Estimation", In: *Proc. of the Third International Conf. on Trustworthy Systems and Their Applications*, Wuhan, China, pp. 8–16, 2016.

[20] B. Wen, Y. Zhu, R. Subramanian, T. T. Ng, X. Shen, and S. Winkler, "COVERAGE – a novel database for copy-move forgery detection", In: *Proc. of the International Conference on Image Processing*, Phoenix, AZ, USA, pp. 161–165, 2016.