INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2020, VOL. 66, NO. 3, PP. 561-569
Manuscript received January 12, 2020; revised July, 2020. DOI: 10.24425/ijet.2020.134013

Determination of the Optimal Threshold Value and Number of Keypoints in Scale Invariant Feature Transform-based Copy-Move Forgery Detection

R. Rizal Isnanto, Ajub Ajulian Zahra, Imam Santoso, and Muhammad Salman Lubis

***Abstract*—The copy-move forgery detection (CMFD) begins with the preprocessing until the image is ready to process. Then, the image features are extracted using a feature-transform-based extraction called the scale-invariant feature transform (SIFT). The last step is features matching using Generalized 2 Nearest-Neighbor (G2NN) method with threshold values variation. The problem is what is the optimal threshold value and number of keypoints so that copy-move detection has the highest accuracy. The optimal threshold value and number of keypoints had determined so that the detection has the highest accuracy. The research was carried out on images without noise and with Gaussian noise.**

***Keywords*—forgery, Gaussian noise, feature extraction, pattern matching, Euclidean distance**

I. Introduction

THE rapid development of technology makes digital information manipulation easier. The form of digital information that is often manipulated is digital image. To detect the manipulation of digital images, a digital forensic method is needed [1]. One form of manipulation that is often done is copy-move manipulation.

Copy-move manipulation is a method of digital image manipulation in which an object on a digital image is copied and overwritten it on the object to be moved on the same image [2]. Given that the copied area is included in the same image, the properties of the copied area such as the noise components, color palette, dynamic range, and other properties will be similar to that of the other part of the image [3].

An example of copy-move forgery is depicted in Fig. 1(b) from the original image shown in Fig. 1(a). Forgery is done to hide some pieces of important evidence. In Fig. 1(a), two traffic signs can be seen on the right side of the image. However, these traffic signs have been hidden by copying some areas from the same image and pasting it onto the two traffic signs, as depicted in Fig. 1(b).

Several methods have been used to detect copy-move forgery in images. These methods include the discrete cosine transform (DCT) [4], principal component analysis [5], and robust detection [6] methods.

In 1999, Lowe [7] introduced a method of detecting and describing the characteristics of digital images called the scale invariant feature transform (SIFT). The characteristics of this method do not change with the rotation, translation, and scaling treatments. The characteristics of digital images obtained by the SIFT method can be matched with the characteristics of other images that are also obtained by the SIFT method to detect the similarities between these images. This process is called the matching process.

(a)

(b)

Fig.1. An example of the effect of the copy-move forgery operation on an image: (a) original image with two traffic signs and (b) forged image without traffic signs

A method based on blur moment invariants was proposed by Mahdian and Saic [8] to localize duplicated regions in digital

This research was financially supported by The Faculty of Engineering, Diponegoro University, Indonesia through Strategic Research Grant 2019. This research received no external funding

R.R. Isnanto is with Computer Engineering Department of Diponegoro University, Semarang, Indonesia (e-mail: rizal@ce.undip.ac.id).

A.A. Zahra, I. Santoso, and M.S. Lubis are with Electrical Engineering Department of Diponegoro University, Semarang, Indonesia (e-mail:: ayub.ayul1an@elektro.undip.ac.id, imamstso@elektro.undip.ac.id, and muhammadsalmanlubis@students.undip.ac.id, respectively).

images automatically. First, one image is divided into several overlapping blocks, where the blur moment invariants are used to represent the blocks. The principal component transform or Karhunen Loeve transform is used to reduce the dimension of the blocks. To perform range queries efficiently, a *k-d* tree is used to analyze the similarity of blocks in multidimensional data. The output of the algorithm is a map of duplicated regions. The proposed method exhibits a high capability of detecting copy-move forgery in an image even when changes such as blur degradation, additional noise, and arbitrary contrast are present in the copied regions [8].

In 2011, Amerini, Ballan, Caldelli, Bimbo, and Serra [9] proposed a new feature matching method called the generalized 2 nearest-neighbor (G2NN) methodd. The SIFT method can be used to obtain the characteristics of an image that is suspected to have undergone the manipulation process. The characteristics obtained from these images are matched between one segment or region and another segment or region in the same image as the G2NN method. The use of this method is expected to produce an accurate copy-move forgery detection (CMFD) algorithm.

This research aimed to determine the optimal threshold value and number of keypoints so that the SIFT-based CMFD has the highest accuracy. This research was conducted on images without and with Gaussian noise. The results of this research are expected to identify the optimum values for both threshold and number of keypoints to obtain the highest level of accuracy in SIFT-based CMFD.

II. Literature Review

A new method for detecting copy-move forgery was proposed by Li, Li, and Zhu [10]. First, the image is filtered and divided into several overlapping circular blocks. Then the local binary patterns (LBP) is used to extract the circular block features. Subsequently, a comparison of feature vectors is performed and the location of forgery regions determined by tracking the corresponding blocks. The experimental results indicate that this scheme is robust to blurring noise, contamination, JPEG compression, and region flipping and rotation [10,11].

Meanwhile, Fridrich, Soukal, and Lukas [4] conducted an investigation on the CMFD problem. They described an efficient and reliable method to detect copy-move forgery. Their method can successfully detect the parts of the forgery even when the copied area is enhanced and the forged image is saved as a compressed formatted file, i.e., JPEG. The performance of the proposed method on several forged images has been demonstrated in their research [4].

Rinjani and Poovendran [12] utilized two techniques, i.e., DCT and inverse DCT using the row and column reduction method to detect copy-move forgery in an image. In this scheme, the original image is initially divided into some matrices. Then the DCT technique is implemented. Subsequently, the matrices are transformed into some blocks with various dimensions. Finally, the duplicated images are grouped on the basis of the obtained threshold values. This method reduces the complexity of computation related to both cost and time. At the same time, this method increases the efficiency of the processed image [12].

An algorithm with the good CMFD performance was reported by Al-Qershi and Bee Ee [13]. Some features within their category are selected. They achieve a good performance by considering two aspects. First, reducing the complexity, the execution time is reduced as well. This reduction is achieved by utilizing the small sized feature vectors. Second, the robustness of the algorithm against image processing operations is increased by adopting the robustness of the features that are invariant to a wide range of image processing operations [13].

A detailed review of existing CMFD techniques based on both discrete wavelet transform (DWT) and DCT was presented by Mukherjee and Mitra [14]. Their analysis proved that both techniques have advantages and disadvantages. The success of both techniques relies solely on the size of the block, size of the copy-moved region, type of sorting applied, geometrical transformations applied, and amount of compression introduced [14].

A hybrid approach was proposed by Ardizzone, Bruno, and Mazzola [15]. Their research compared triangle points with single points or blocks. Points of interest are extracted from the image. Then, on these points, objects are modeled as a set of connected triangles. Subsequently, the triangles are matched according to the shapes of their inner angles, local features, and content color information. Finally, the results are compared using a point-based method and a state-of-the-art block matching method. The results of their research indicated that the proposed method exhibits a good performance in case of simple scenes, where both the number of keypoints and triangles are low. In case of complex scenes, the poor performance of the matching process is influenced by the high number of detected triangles. A similar result can also be obtained by the keypoint-based approaches. However, these methods cannot be used when no points of interest are detected, for example, when a homogeneous region is used to hide an object in the image. Moreover, in case of anisotropic deformations, the proposed method can be used in the future [15].

Sharma, Abrol, and Devanand [16] investigated the feature-based analysis of copy-paste image tampering detection observation. The results of their research indicated that the proposed model works well for low to moderate levels of copy-paste tampering and identifies the tampered area for all of the observed images. The results obtained can be used to enhance the tampering detection process by identifying the most likely cases of possible image tampering and providing the initial verification of the tampered images [16].

A method to implement CMFD with particle swarm optimization (CMFD-PSO) was investigated by Wenchang, Fei, Bo, and Bin [17]. The SIFT-based framework integrates and implements the PSO algorithm in this method to generate the values of the customized parameters for image processing, which are used to detecting copy-move forgery. The experimental results show that CMFD-PSO has a good performance. Moreover, CMFD-PSO outperforms SIFT-based methods and can increase the number of true matched keypoints to ensure accurate decisions for region duplication [17].

A fast exploration method for Copy-Move forgery images was proposed by Shin [18]. This method can reduce computational complexity better than conventional methods. In the proposed scheme, the author used a half-block size in spatial domain, rather than use an 8x8 pixel block, frequency algorithm, or exhaustive search method, to reduce computational complexity [18].

The performance of different widely used features for CMFD methods was evaluated by Christlein and Jordan [19].

They evaluated the performance of previously proposed feature sets. They observed the 15 most prominent feature sets and analyzed the detection performance on a per-image basis and a per-pixel basis. The results of their experiments showed that the keypoint-based features have a similar performance to the block-based and keypoint-based methods. However, the keypoint-based features are sensitive to low-contrast regions and repetitive content of the image. In this case, the block-based methods can clearly improve the detection results [19].

CMFD using a system based on the color coherence vector was proposed by Ulutas and Ulutas [20]. Their algorithm can detect the forged areas with high accuracy ratios [20]. Other experiments conducted by Farooque and Rohankar [21] focused on various noises and techniques for denoising the color image. The results of their research showed that the method can detect forged region even when the forged image is hidden using Gaussian blurring [21].

A technique based on DWT was proposed by Khan and Kulkarni [3]. In this technique, initially, DWT is applied to the input image to reduce the representation dimension. Afterward, the image is divided into some overlapping blocks. These blocks are sorted. Then, the duplicated blocks are identified using phase correlation based on similarity criteria. This approach drastically decreases the detection time consumed. The results of their experiments indicated that the method is robust to ordinary post-processing operations. However, this method cannot detect the duplicated regions with rotation based on scaled regions and scaled angles [3].

A survey on keypoint-based methods based on various parameters was conducted by Chauhana, Katasb, Jainc, and Thakared [22]. They concluded that SIFT is an efficient technique and can detect forgery in both single and multiple regions of an image. The method obtains good results in case of both geometric transformation, such as translation, rotation, or scaling, and plain copy-move forgery. However, SIFT is invariant to affine transformation, scaling, and rotation. SIFT also exhibits a higher computational efficiency than speeded up robust features (SURF). However, the accuracy of SIFT is lower than that of SURF [22].

Pun, Yuanand, and Bi [23] conducted research on a CMFD scheme using feature point matching and adaptive over-segmentation. The proposed detection scheme integrates both keypoint-based and block-based forgery detection methods. Initially, the proposed algorithm adaptively segments the host image into both irregular and nonoverlapping blocks. Then, from each block regarded as block features, the feature points are extracted. Finally, the block features are matched with other block features to locate the labeled feature points. The experimental results show that the proposed CMFD scheme can achieve better detection results under various challenging conditions, such as JPEG compression, down-sampling, and geometric transformation, than previously existing CMFD schemes [23].

According to Kaur and Dutta [24], several approaches can be utilized to forge digital images. However, the main classes of digital image forgery are enhancing, splicing, morphing, retouching, and copy-move [24].

A method for detecting copy-move forgery was proposed by Ustubioglu, Ulutas, Ulutas, and Nabiyev [25]. They claimed that the method can calculate the threshold value automatically. To analyze the similarity of the blocks, the method uses element-by-element equality between the feature vectors, rather than the cross correlation or Euclidean distance. The method also utilizes the history of compression to determine automatically the threshold value of the current test image automatically. Their experimental results indicated that this method can detect the copied areas using different scenarios and achieve higher accuracy levels and lower false negatives than similar methods [25].

III. Materials and Methods

A. Test Images

The data that will be tested by the application that we have developed to detect forgery copy-move in digital images consists of 20 input data consisting of 14 images from the MICC-F220 dataset and six images that we made ourselves. From those 20 images, we made 20 images with Gaussian noise, for testing with Gaussian noise. Thus, there are 40 test images used.

B. Application Design

Broadly speaking, the detection process of copy-move forgery on digital images in this application consists of three stages, namely, preprocessing, feature extraction using SIFT, and CMFD using G2NN. Fig. 2 shows the block diagram of these stages.

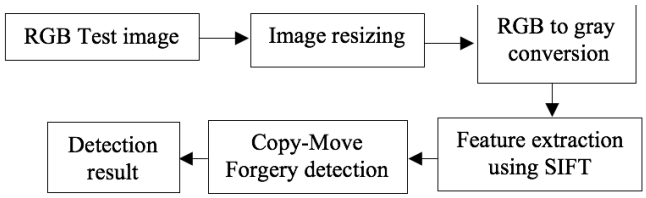

Fig. 2. Block diagram of the application design

In the preprocessing stage, the test image is resized to 400 × 300 pixels. This stage aims to accelerate the computational time because the size of the input image is varied and relatively large. Afterward, the RGB image is converted into a gray-level image so that the feature extraction process using the SIFT algorithm can be implemented because the SIFT algorithm can only be applied to gray level image. The processes that occur in this software simulation are shown in more detail in Fig. 3.

The flow diagram of the application software shown in Fig. 3 can be explained as follows: First, the test image is resized and converted into a gray-level image. Moreover, interference in the form of Gaussian noise can be added before the image undergoes the feature extraction process [3]. The addition of Gaussian noise aims to test its effect on the accuracy of CMFD.

The next step is the feature extraction stage using the SIFT method. Afterward, the process of selecting the input or the input threshold value T is implemented. The threshold values T that are available in this software are 0.3, 0.4, and 0.5.

After inputting the threshold value selected, the next step is to choose the minimum number of suitable features. The results of the selection of the minimum number of suitable features are stored in the min_feat variable. The three choices of min_feat that are available are min_feat = 2, min_feat = 5, and min_feat = 10.

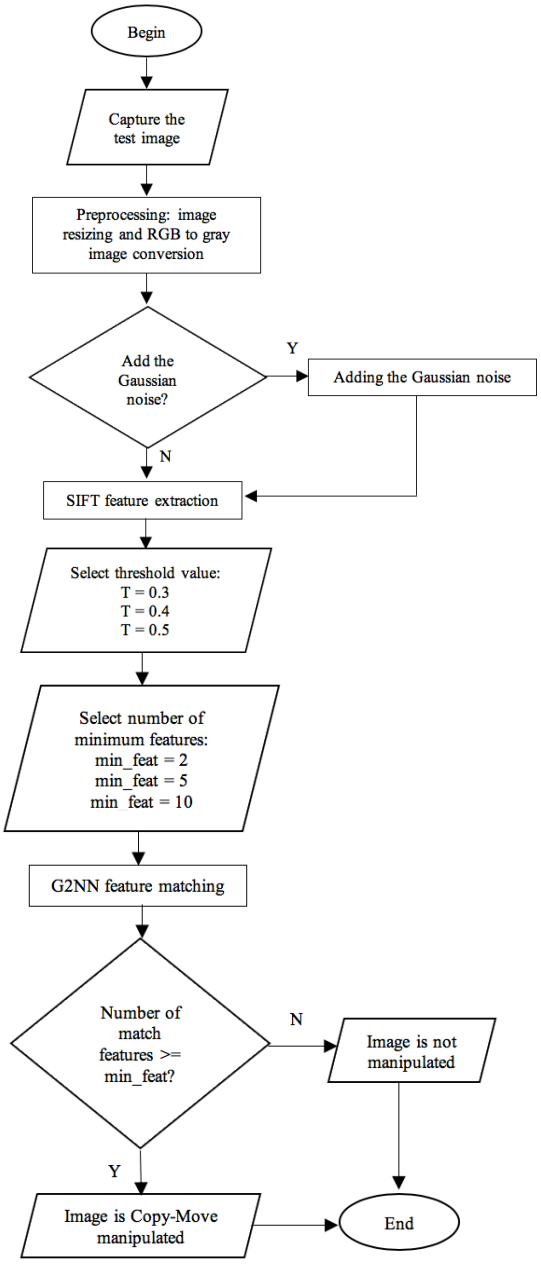

Fig. 3. Flow diagram of the application software.

After the minimum number of suitable features is selected, the next step is the characteristic matching process using G2NN. The number of suitable features will determine whether the input image has copy-move forgery or not.

In the final stage, i.e., the stage in which the status of the image is determined, the number of suitable features is compared with the variable min_feat. If the number of matching features is greater than or equal to the value of min_feat, then the image has undergone copy-move forgery. By contrast, if the number of matching features is smaller than the value of min_feat, then the image has not undergo copy-move forgery.

C. Scale-Invariant Feature Transform

The first stage of the SIFT algorithm is the detection of extreme values on the space scale. This extreme value is the keypoint of SIFT. The space scale of an image is defined as the function L (x, y, σ), which is the result of the convolution between Gaussian function G (x, y, σ) and input image I (x, y, σ), where σ is a constant factor for true scale invariance. After the space scale of the image is obtained, the next step is to calculate the difference of Gaussian (DoG) function [7] from the image using Equation 1.

$$D(x,y,\sigma) = \big(G(x,y.k\sigma) - G(x,y,\sigma)\big) \times I(x,y) \quad (1)$$

where $G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$ (2)

The illustration of spatial scale generation is shown in Fig. 4(a). To detect extreme values in DoG images, the value of each pixel on the DoG space scale is compared with eight pixels around it and nine pixels corresponding to the previous and following DoG images. This process is shown in Figure 4(b). The second stage of the SIFT method is determining keypoints. The location of the keypoints is determined using Equation 3. The keypoints are detected by analyzing the DoG images, i.e., by finding the local maxima or the local minima. For every pixel in a DoG image, it is compared to its eight surrounding neighbors in the same DoG image, and the nine surrounding neighbors in its upper-level DoG image, and the nine surrounding neighbors in its lower-level DoG image, as shown in Fig. 4(b). The pixel is identified as a keypoint candidate if it is the maximum or the minimum out of the total 26 neighboring pixels. Each keypoint candidate will have to pass a subsequent stability checking procedure in order to become a true keypoint. During this process, the candidates with relatively low contrasts (considered as flat points) are rejected, and the candidates located on the edges (considered as not distinct) are also eliminated as well [26].

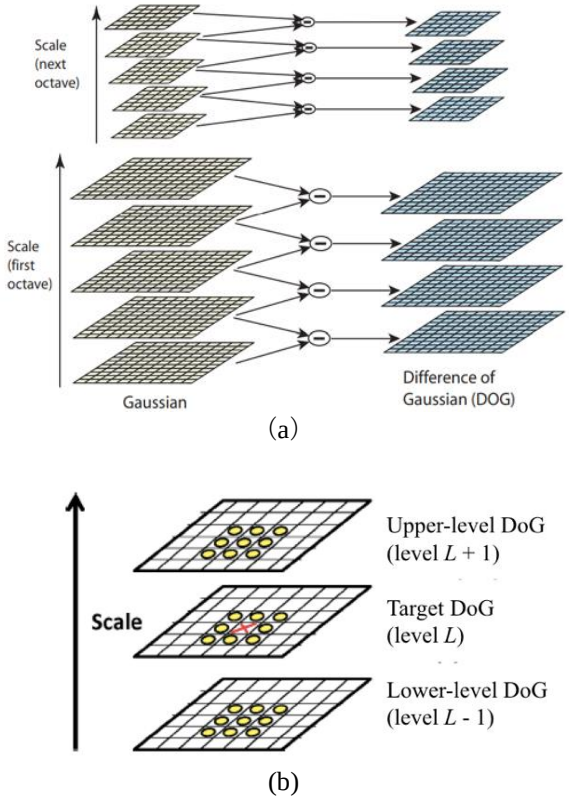

Fig. 4. (a) Illustration of spatial scale generation and (b) determination of extreme values in DoG images.

The second stage of the SIFT method is determining keypoints. The location of the keypoints is obtained by Equation 3, where the keypoint value is obtained using Equation 4.

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2}\frac{\partial D}{\partial x} \quad (3)$$

$$D(\hat{x}) = D + \frac{1}{2}\frac{\partial D^T}{\partial x}\hat{x} \quad (4)$$

To eliminate keypoints with low contrast, extreme values $|D(\hat{x})|$ lower than a threshold value are removed. To eliminate the candidate keypoints that are unclear and located along the edge, a Hessian matrix of second-order **H** is used, as shown in Equation 5:

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \text{ where } \frac{D_{xx}+D_{yy}}{D_{xx}D_{yy}-(D_{xy})^2} < \frac{(r+1)^2}{r}, \quad (5)$$

where *r* is the threshold of the principal curvature allowed. The keypoint that has a principal curvature value greater than *r* will be omitted. The keypoints will be used are that after the keypoint candidates with low contrast, which are unclear and located along the edge, are removed.

The third stage of the SIFT method is orientation determination, which aims to obtain an invariant or unchanged characteristic after rotation treatment. Orientation determination is done using Equations 6 and 7.

$$m(x,y)\sqrt{L(x+1,y)-L(x-1,y)+L(x,y+1)-L(x,y-1)} \quad (6)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right) \quad (7)$$

The final stage of the SIFT method is obtaining the keypoint descriptor. This stage is to transfer the detected key points and their neighboring pixels into specified feature descriptors. Descriptor is an orientation histogram with size of 4 × 4 pixels. This histogram is calculated from the magnitude and orientation value of the sample in the region of 16 × 16 around the keypoint. Magnitude is calculated by a Gaussian function with σ equal to one half the width of the descriptor. Taking a keypoint as the center, its keypoint-region is divided into 4 × 4 = 16 square sub-regions on the Gaussian-filtered image hosting the target keypoint, as illustrated in Fig. 5(a) and Fig. 5(b). The gradient histogram of orientation is computed for each sub-region, and each histogram now has eight orientation bins as shown in Fig. 5(c). In other words, each bin covers 45°. There is a subtle detail—the gradient histograms of orientations are weighted by a Gaussian function as specified in Lowe's algorithm [7]. To achieve rotation invariance, the pixels within each sub-region are further rotated with the key-point orientation as we have computed previously. Overall, these 16 histograms will be represented by 16×8 = 128 values [26].

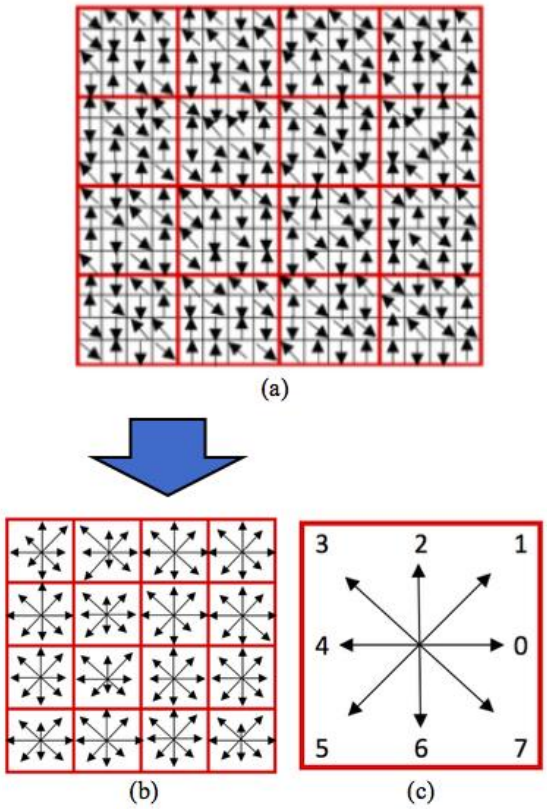

Fig. 5. Generation of feature descriptor: (a) keypoint region around a key point (b) 4x4 sub-regions within a keypoint region (c) Gradient histogram of one sub-region.

D. Generalized 2 Nearest-Neighbor

After the feature extraction process using the SIFT method is completed, the number of *n*-keypoints and their descriptors are obtained. The next step is inter-descriptor matching of each keypoint to identify the same segment in the test image. For example, the SIFT method is applied to a test image so that it produces a collection of keypoints $X = \{x_1, x_2, \dots, x_n\}$ with each descriptor $\{f_1, f_2, \dots, f_n\}$. The 2 Nearest Neighbor (2NN) method is applied to each f_i of the keypoints. The match for each keypoint is identified using the 2NN method by finding the nearest neighbor between keypoint x_i and all (n - 1) other keypoints found in the test image [10]. The closest neighbor is the keypoint with the smallest Euclidean distance. The Euclidean distance is the most well-known tool for measuring similarity. *D* is defined as the similarity vector of the descriptor of a keypoint containing Euclidean distances sorted from the smallest to the largest with the descriptor of another keypoint. The function used to calculate Euclidean distance is expressed in Equation 8 [6].

$$D = \{d_1, d_2, d_3, \dots, d_{n-1}\}, \text{ where } d = \sqrt{(f_a - f_b)^2} \quad (8)$$

where *D* is the similarity vector of a keypoint, *d* is Euclidean distance of a descriptor, f_a is the descriptor vector of a keypoint, f_b is the descriptor vector of another keypoint, and *n* is the vector length of the test image descriptor.

After the Euclidean distance is obtained, the next step is to calculate the operation of the division between the distance of the first and second closest neighbors. The results of the division are compared with the threshold value *T*. In this manner, the keypoint will be categorized as suitable if it meets the requirements [6], as follows:

$$\frac{d_1}{d_2} < T \quad \text{where} \quad T \in (0,1) \tag{9}$$

The 2NN matching method is unsuitable for detecting forgery in images with multiple duplicated regions because this method only evaluates the two closest neighbors at each keypoint. To overcome this problem, Amerini, Ballan, Caldelli, Bimbo, and Serra [9] proposed a new method, which is a development of the 2NN method called G2NN. In this method, the calculation d_1/d_2 is repeated until the results obtained are greater than the threshold value. If the looping process stops at this value, then each keypoint with distance $\{d_1, d_2, d_3, \dots, d_k\}$ where $1 \leq k < n$, is grouped as a "match" [9].

IV. Test Results and Analysis

A. Supporting Devices

The software used to simulate CMFD software on digital images is MATLAB R2016a. Meanwhile the hardware used is a personal computer with the following specifications:

1) Hardware: Laptop Lenovo Yoga IdeaPad 520, Intel Core i5-8250U Processor;
2) RAM 4GB DDR4;
3) Operating system: Microsoft Windows 10 Home.

B. Testing without Gaussian Noise

Testing without Gaussian noise involves testing images with variations in the use of the threshold value *T* and the number of keypoints in test images without Gaussian noise. The use of Gaussian noise was proposed in the research conducted by Farooque and Rohankar [21]. A total of 20 images were tested, with variations of the minimum number of keypoints, i.e., 2, 5, and 10, and variations of the threshold value T, i.e., 0.3, 0.4, and 0.5. Table I shows the results of detection accuracy testing without Gaussian noise. Meanwhile, Figure 6 shows the graph of the results of the test shown in the table.

Table I
Detection Accuracy Testing Without Gaussian Noise

Minimum number of matching keypoints	Threshold value *T*	Accuracy rate (%)
2	0.3	90
	0.4	100[1]
	0.5	90
5	0.3	60
	0.4	85
	0.5	85
10	0.3	55
	0.4	65
	0.5	80

[1]The best results

Figure 6 shows that the results of testing images with a minimum number of suitable features of two characteristics and variations of the threshold value T = 0.4 have the highest accuracy value of 100%. From this table, the average of accuracy rate from 9 observation is 78.89%. This accuracy value can be compared with that obtained by Amerini, Ballan, Caldelli, Bimbo, and Serra [9] using a SIFT-based forensic method. They claimed that the true positive rate (TPR) of gamma correction processing is 99.37%, whereas the TPR of both JPEG and SNR (dB) processing is 100% [26].

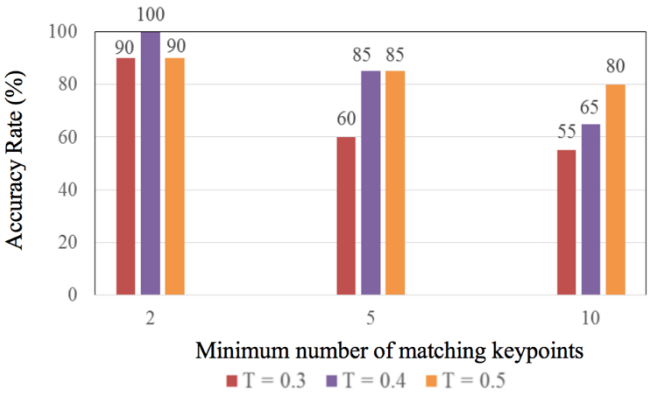

Figure 6. Graph of detection accuracy testing without Gaussian noise.

The highest accuracy of 100% reached with variations of the threshold value T = 0.4 can be compared with the results of the research conducted by Wang, Zhang, and Zhou [27]. They used an image CMFD scheme based on Accelerated-KAZE (A-KAZE) and SURF. Their experimental results indicated that the performance of the proposed scheme is superior to that of other tested CMFD methods [28]. However, they did not state the highest accuracy level of their proposed scheme. Variations of the threshold values T = 0.3 and T = 0.5 have the same accuracy value of 90%. The results of testing images with a minimum number of characteristics matching as many as five features and variations of the threshold values T = 0.4 and T = 0.5 have an accuracy value of 85%.

The highest accuracy of 100% reached in this research also can be compared with the results of the research conducted by Prakash, Panzade, Om, and Maheshkar [28]. They proposed a keypoint-based CMFD technique, which is a combination of A-KAZE and SIFT. Their experimental results showed that their proposed method can detect the duplicated regions even if the image is post-processed with scaling, rotation, noise, and JPEG compression operations [28]. However, they did not state the highest accuracy level of their proposed technique.

Variations of threshold value T = 0.3 have the lowest accuracy value of 60%. The results of testing images with a minimum number of features matching as many as 10 features and variations of the threshold value T = 0.5 have the highest accuracy value that is equal to 80%. Variations of the threshold value T = 0.3 have an accuracy value of 55%, and variations of the threshold value T = 0.4 have an accuracy value of 65%. Figure 7 shows an example of forgery detection in an image without Gaussian noise.

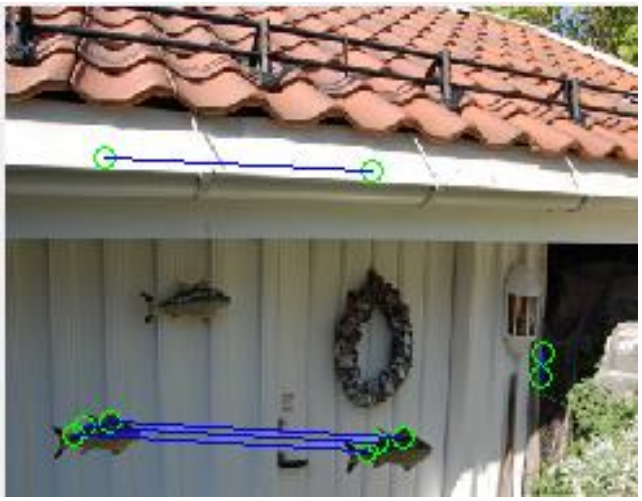

Figure 7. Detection results of test image without Gaussian noise

C. Testing with Gaussian Noise

Testing with Gaussian noise involves testing images with variations in the use of the threshold value *T* and the minimum number of keypoints in test images with Gaussian noise. As mentioned previously, the use of Gaussian noise was proposed in the research conducted by Farooque and Rohankar [21]. A total of 20 images were tested, with variations of the number of keypoints. i.e., 2, 5, and 10, and variations of the threshold value *T*, i.e., 0.3, 0.4, and 0.5. Table II shows the accuracy percentage for the detection of test images with Gaussian noise. Meanwhile, Figure 8 shows the graph of the results of the test shown in the table.

TABLE II
ACCURACY PERCENTAGE FOR THE DETECTION OF TEST IMAGES WITH GAUSSIAN NOISE

Minimum number of matching keypoints	Threshold value *T*	Accuracy rate (%)
2	0.3	50
	0.4	65[1]
	0.5	65[1]
5	0.3	30
	0.4	40
	0.5	55
10	0.3	30
	0.4	40
	0.5	40

[1]The best results

Figure 8 shows that the results of testing images with a minimum number of suitable features of two characteristics and variations of the threshold value T = 0.3 have the lowest accuracy value of 50%. Variations of threshold values T = 0.4 and T = 0.5 have an accuracy value of 65%. The results of testing images with a minimum number of suitable features matching as many as five features and variations on the threshold value T = 0.3 have an accuracy value of 30%. Meanwhile, variations of the threshold value T = 0.4 have an accuracy value of 40% and variations of the threshold value T = 0.5 have the highest accuracy value of 55%. The results of testing images with a minimum number of characteristics matching as many as 10 features and variations of the threshold value T = 0.4 and T = 0.5 have the same accuracy value that is equal to 40%. Variation of the threshold value T = 0.3 have the lowest accuracy value of 30%. Compared with the results of the research conducted by Elaskily et al. [29], the results of this research have a lower accuracy level. They claimed that the best result occurs when Gaussian noise with gamma correction processing is applied to the image, with TPR of 100% and false positive rate of 7.14% [29]. However, we cannot conclude that our results are worse because, in fact, the test images used and the conditions applied are different from those of Elaskily et al. [29].

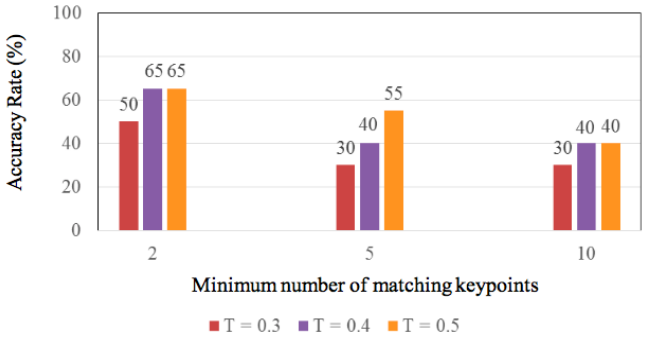

Figure 8. Graph of accuracy percentage for the detection of test images with Gaussian noise

The accuracy of testing with Gaussian noise is relatively lower than the accuracy of testing without Gaussian noise. This finding can be attributed to the fact that Gaussian noise leads to suboptimal operation of the SIFT algorithm, which is a keypoint-based algorithm, because the keypoints that can be detected become undetectable, resulting in the lower number of traits obtained from test images with Gaussian noise than test images without Gaussian noise. Figure 9(a) shows an example of an image that has Gaussian noise, whereas Fig. 9(b) shows an example of its detection results.

D. Discussion

Furthermore, our experiment to combine the SIFT and G2NN (shortnamed SIFT-G2NN from now on) is compared with other efforts from some researchers. Following are a few publications from other authors which we summarize the results of their experiments.

According to Wu, Abd-Almageed, and Natarajan who introduced BusterNet, a two-branch DNN (deep neural network architecture), they claimed that the accuracy of the proposed BusterNet jumps to 77.49% [30]. Whereas, D'Amiano, Cozzolino, Poggi, and Verdoliva had proposed a PatchMatch based dense-field algorithm for video copy-move detection and localization. Their experimental results show that the proposed method to detect and localize video copy-moves with good accuracy even in adverse conditions [31], without declaring specific number to indicate the accuracy level.

In their publication on CMFD based on PatchMatch. Cozzolino, Poggi, and Verdoliva declared that their experiment results show the proposed technique to perform almost uniformly all tested reference techniques in terms of both accuracy and speed. The basic PatchMatch implemented in RGB pixels (B-PM + RGB) provides a very good performance in general, with F = 0.906 in case of simple translation [32]. While, in their paper on CMFD based on polar cosine transform (PCT) and appropriate nearest neighbour searching and accomplished by means of locality-sensitive hashing (LSH), Li

declared that the proposed algorithm has a precision of 0.98, higher than that of Zernike-CMFD that has a precision of 0.92 [33]. We shortname the algorithm as PCT-LSH.

(a)

(b)

Fig. 9. Detection process on a noisy image: (a) an example of an image with Gaussian noise; (b) detection result of an image with Gaussian noise.

In another publication, Cozzolino, Poggi, and Verdoliva proposed a new algorithm named PatchMatch (PM) for the accurate detection and localization of copy-move forgeries, based on rotation invariant features computed densely on the image. With all features, the proposed technique behaves very well on rigid copy-moves, with *F*-measure going from 0.9 for Fourier-Mellin Transform (FMT) to 0.94 for Zernike-polar [34]. The method for detecting copy-move forgery also had been introduced by Bayram, Sencar, andd Memon. Their experimental results show that the proposed features can detect duplicated region in the images very accurately, even when the copied region was undergone severe image manipulations [35]. Again, in this paper, the authors did not declare what specific number to indicate the accuracy level.

Marra, Gragnaniello, Verdoliva, and Poggi had worked on research on a full-image full-resolution end-to-end-trainable convolutional neural network (CNN) framework for image forgery detection. They also claimed that the experiments on widespread image forensics datasets prove the good performance of the proposed approach, which largely outperforms all baselines and all reference methods. They compute the area under the ROC curve (AUC) as a synthetic measure of performance. Over the whole dataset, the best AUC, obtained with E2E-Fusion (end-to-end fusion), grows from 0.846 to 0.932 on NC2017 [36]. In another experiment, Li, Li, Yang, and Sun proposed a scheme to detect the copy-move forgery in an image, mainly by extracting the keypoints for comparison. The experimental results prove the good performance of the proposed scheme via comparing it with the state-of-the-art schemes on the public databases. In this research, false positive rate (FP) is 17/48 = 0.354, better than the other experiments using SIFT which results in FP = 9/48 = 0.188 or SURF with TP = 8/48 = 0.167 [37].

We conclude the above results in Table III. From 9 (nine) algorithms observed, there are only 6 (six) algorithms that declared their numerical results.

TABLE III
TEST RESULTS ON SOME CMFD ALGORITHMS

Algorithm	Test results	Conditions
BusterNet [31]	Accuracy = 77,49%	Overall accuracy is the ratio of corrected samples to total samples [31]
B-PM+RGB [33]	FM = 0.906 = 90.6%	FM = F-measure = $2TP/(2TP + FN + FP)$, where TP = true positive, FN = false negative, and FP = false positive [33]
PCT-LSH [34]	Precision = 0.98 = 98%	Precision = (Forged Region ∩ Detected Region)/Detected Region [34]
PM [35]	F-measure = 0.94 = 94%	$F = 2TP/(2TP + FN + FP)$; where TP = true positive, FN = false negative, and FP = false positive [35]
CNN [37]	AUC = 0.932 = 93.2%	AUC is the area under the ROC curve; as a synthetic measure of performance [37]
Proposed method: SIFT-G2NN	Average accuracy = 78.89%	Accuracy is the ratio of corrected samples to total samples; The highest accuracy of 100% reached with variations of the threshold value T = 0.4

From Table III it can be seen that the test results above are from different measurement parameters. Two algorithms: BusterNet and SIFT-G2NN use accuracy measurements with the same formula. From the two algorithms, it can be shown that the proposed method SIFT-G2NN has an accuracy rate of 78.89%, slightly higher than BusterNet which has an accuracy of 77.49% [30]. Thus, the SIFT-G2NN algorithm yields slightly improved results compared to the BusterNet algorithm.

Two other algorithms: B-PM + RGB [32] and PM [34] use the same size, i.e., F-measure, which is obtained by the same formula. While the other 2 (two) algorithms use different measures. The PCT-LSH algorithm uses a precision measure [33], while CNN uses an AUC measure, which is the area under the ROC curve [36].

V. CONCLUSION

The experimental results show that the detection accuracy decreases when the image has Gaussian noise. Simulations of CMFD without Gaussian noise achieve the highest accuracy with a value of 100% at a threshold value T = 0.4 and a minimum number of traits or keypoints of 2. At this scheme, it can be shown that the proposed method SIFT-G2NN has an average accuracy rate of 78.89%, slightly higher than BusterNet which has an accuracy of 77.49%. Simulations of CMFD with

Gaussian noise achieve the highest accuracy with a value of 65% at the threshold value of $T = 0.4$ and $T = 0.5$ and a minimum number of traits or keypoints of 2. The higher the minimum number of keypoints, the lower the detection accuracy. The threshold value $T = 0.3$ produces a relatively low accuracy value compared with other threshold value T in all test scenarios. In future research, this simulation can be combined with clustering methods, such as J-linkage or agglomerative hierarchical clustering after feature matching, to increase the detection accuracy.

REFERENCES

[1] G. Palmer, "A Road Map for Digital Forensic Research," Technical Report (DTR-T001-01) for Digital Forensic Research Workshop, New York, 2001.

[2] M. Puri and V. Chopra, "A Survey: Copy-Move Forgery Detection Methods." International Journal of Computer Systems (IJCS), vol. 3, no. 9, pp: 582-586, September 2016.

[3] S. Khan, and A. Kulkarni, "An Efficient Method for Detection of Copy-Move Forgery Using Discrete Wavelet Transform. International Journal on Computer Science and Engineering," Vol. 02, No. 05, 2010, pp. 1801-1806.

[4] J. Fridrich, D. Soukal, and J. Lukas, "Detection of Copy-Move Forgery in Digital Images," Proceedings of Digital Forensic Research Workshop, IEEE Computer Society, August 2003, pp. 55–61.

[5] P. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Duplicated Image Regions," Computer Science, Technical Report (TR2004-515), Dartmouth College, 2004.

[6] W. Luo and J. Huang, "Robust Detection of Region-Duplication Forgery in Digital Image," IEEE - The 18th International Conference on Pattern Recognition (ICPR'06), 2006.

[7] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, January 2004, pp. 91-110.

[8] B. Mahdian and S. Saic, "Detection of Copy-Move Forgery using a Method based on Blur Moment Invariants," Forensic Science International, an international journal dedicated to the applications of medicine and science in the administration of justice, vol.171, no. 2-3, September 2007, pp. 181-189.

[9] I. Amerini, L. Ballan, R. Caldelli, A.D. 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, September 2011, doi 10.1109/TIFS.2011.2129512, pp. 1099-1110

[10] L. Li, S. Li, and H. Zhu, "An Efficient Scheme for Detecting Copy-Move Forged Images by Local Binary Patterns," Journal of Information Hiding and Multimedia Signal Processing, vol. 4, no. 1, January 2013, pp. 46-56.

[11] V. Jabade, and S. Gengaje, "Modelling of Geometric Attacks for Digital Image Watermarking," IJIERT - International Journal of Innovations in Engineering Research and Technology, vol. 3, no. 3, March 2016.

[12] M.B. Ranjani, and R. Poovendran, "Image Duplication Copy-Move Forgery Detection Using Discrete Cosine Transforms Method," International Journal of Applied Engineering Research, vol. 11, no. 4, 2016, pp. 2671-2674.

[13] M. Osamah, A. Al-Qershi and K.B. Ee, "Passive Detection of Copy-Move Forgery in Digital Images: State-of-the-Art," Forensic Science International, vol. 231, no. 1, September 2013, pp. 284-295.

[14] P. Mukherjee, S. Mitra, "A Review on Copy-Move Forgery Detection Techniques Based on DCT and DWT," International Journal of Computer Science and Mobile Computing IJCSMC, vol. 4, no. 3, March 2015, pp.702 – 708.

[15] E. Ardizzone, A. Bruno, and G. Mazzola, "Copy-Move Forgery Detection by Matching Triangles of Keypoints," IEEE Transactions on Information Forensics and Security, vol. 10, no. 10, October 2015, pp. 2084 – 2094.

[16] K. Sharma, P. Abrol, and Devanand, "D. Feature Based Analysis of Copy-Paste Image Tampering Detection," International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), vol. 2, no. 6, 2017, pp. 555-562.

[17] S. Wenchang, Z. Fei, Q. Bo, and L. Bin, "Improving Image Copy-Move Forgery Detection with Particle Swarm Optimization Techniques," China Communications, vol. 13, no. 1, January 2016, pp. 139 – 149.

[18] Y.D. Shin, "Fast Exploration of Copy-Move Forgery Image," Advanced Science and Technology Letters, vol. 123, 2016, pp.1-5.

[19] V. Christlein and J. Jordan, "An Evaluation of Popular Copy-Move Forgery Detection Approaches," IEEE Transactions on Information Forensics and Security, 2012, pp. 1-26.

[20] G. Ulutas, and M. Ulutas, "Image Forgery Detection using Color Coherence Vector," Electronics, Computer and Computation (ICECCO), November 2013, pp. 107-110.

[21] M.A. Farooque and J.S. Rohankar, "Survey on Various Noises and Techniques for Denoising the Color Image," International Journal of Application or Innovation in Engineering & Management (IJAIEM), vol. 2, no. 11, November 2013.

[22] D. Chauhana, D. Kasatb, S. Jainc, and V. Thakared, "Survey on Keypoint Based Copy-Move Forgery Detection Methods on Image," Elsevier-International Conference on Computational Modeling and Security (CMS 2016), pp. 206 – 212.

[23] C.M. Pun, X.C. Yuanand, and X.L. Bi, "Image Forgery Detection Using Adaptive Over-Segmentation and Feature Point Matching," IEEE Transactions on Information Forensics and Security, vol. 10, no. 8, August 2015, pp. 1705 – 1716.

[24] G. Kaur and M. Dutta, "Digital Image Forgery: A Survey," International Journal of Computer Science Research and Technology (IJCSRT), vol. 1, no. 6, November 2013, pp.1-7.

[25] B. Ustubioglu, G. Ulutas, M. Ulutas, and V.V. Nabiyev, "A New Copy-Move Forgery Detection Technique with Automatic Threshold Determination," Elsevier - International Journal of Electronics and Communications, vol. 70, no. 8, August 2016, pp. 1076–1087.

[26] F.C. Huang, S.Y. Huang, J.W. Ker, and Y.C. Chen, "High-performance SIFT Hardware Accelerator for Real-time Image Feature Extraction," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 3, March 2012, pp. 340-351.

[27] C. Wang, Z. Zhang, and X. Zhou, "An Image Copy-Move Forgery Detection Scheme Based on A-KAZE and SURF Features," Symmetry 2018, 10, 706, doi:10.3390/sym10120706, Switzerland, 2018, pp. 1-20.

[28] C.S. Prakash, P.P. Panzade, H. Om, and S. Maheshkar, "Detection of Copy-Move Forgery using AKAZE and SIFT Keypoint Extraction," Multimedia Tools and Applications, August 2019, vol. 78, no. 16, pp 23535–23558.

[29] M.A. Elaskily, H.K. Aslan, M.M. Dessouky, F.E. Abd El-Samie, O.S. Faragallah, and O.A. Elshakankiry, "Enhanced Filter-based SIFT Approach for Copy-Move Forgery Detection," Menoufia Journal of Electronic Engineering Research (MJEER), vol. 28, no. 1, January 2019, pp. 159-181.

[30] Y. Wu, W. Abd-Almageed, and P. Natarajan, "BusterNet: Detecting Copy-Move Image Forgery with Source/Target Localization," Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 168-184.

[31] L. D'Amiano, D. Cozzolino, G. Poggi, and L. Verdoliva, "A Patchmatch-based Dense-field algorithm for video copy–move detection and localization," IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 3, 2018, pp. 669-682.

[32] D. Cozzolino, G. Poggi, and L. Verdoliva, "Copy-move Forgery Detection based on Patchmatch," 2014 IEEE International Conference on Image Processing (ICIP), October 2014, pp. 5312-5316.

[33] Y. Li, "Image Copy-move Forgery Detection based on Polar Cosine Transform and Approximate Nearest Neighbor Searching," Forensic Science International, vol. 1, no. 1-3, 2013, pp. 59-67.

[34] D. Cozzolino, G. Poggi, and L. Verdoliva, "Efficient Dense-field Copy–move Forgery Detection," IEEE Transactions on Information Forensics and Security, vol. 10, no. 11, 2015, 2284-2297.

[35] S. Bayram, H.T. Sencar, and N. Memon, "An Efficient and Robust Method for Detecting Copy-move Forgery," 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, April 2009, pp. 1053-1056.

[36] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi, "A Full-Image Full-Resolution End-to-End-Trainable CNN Framework for Image Forgery Detection," September 2019, arXiv preprint arXiv:1909.06751.

[37] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based Image Copy-move Forgery Detection Scheme," IEEE Transactions on Information Forensics and Security, vol. 10, no. 3, 2014, pp. 507-518.