# Copy-Move Forgery Detection Based on Modified Multi-scale Feature Extraction and CMFD-SIFT

**Mohammed Ikhlayel[†], Mochamad Hariadi[††] and I Ketut Eddy Pumama[††]**

[†] Department of Electrical Engineering Institut Teknologi Sepuluh Nopember(ITS), Surabaya, Indonesia.
[†] Department of Information Technology and Communications, Al-Quds Open University,Palestine.
[††] Department of Electrical Engineering Institut Teknologi Sepuluh Nopember(ITS), Surabaya, Indonesia.
[††] Department of Computer Engineering Institut Teknologi Sepuluh Nopember(ITS), Surabaya, Indonesia.

**Summary**

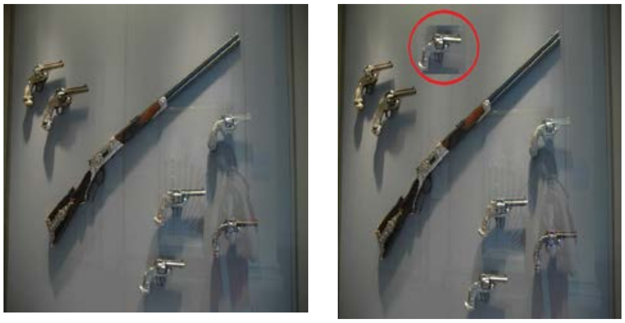
The copy-move attack is a more common method in digital tampering. When copy-move forgery image occur, many important objects add or remove from the image. In order to implement forensic of the images, In the literatures many methods of copy-move forgery detection (CMFD) have been improved. The different approaches of CMFD feature-based was prosed in recent years. but, still more place to enhancement performance further. the problem of Many methods are suffering insufficient matched key points, but forgeries performance on the mirror transformed, then many feature-based methods when the forged region is of uniform texture it might hardly expose the tempering. In this paper we proposed a now scheme, in this scheme the criteria of block and keypoint features will integrate to gather in our scheme, then multiple copy-move regions or objects will be work very well and especially when regions and objects are different sizes and contain both detailed textures and smooth

***Key words:***

*Copy-move Forgery, Tempering, Segmentation, CMFD-SIFT.*

## 1. Introduction

Image forgery detection is one of vary important and major section of digital forensics. The forgery is to produce object in order to basis prejudice or make unlicensed adjustments. Many examples of forgery image in history. lately, the dark rooms were unwieldly used to perform image forgery. but todays, the forgery in digital image no need to dark room because there are many tools available to make forgery in image processing software. The forgery seen very day in press and social media. As a result, from 2001 until now, there was a formidable increase methods developed for image fraud and these days, image fraud has developed a major monotonous in forensic study. Generally, we have classified the forgery techniques into two approaches: active and passive. digital signatures and watermarking are the active were the active approaches requirements specification of dedicated hardware contracts in area of application, while the passive approaches were use image statistics. The passive technique contains many types, as Copy-Move, Retouching, Splicing, etc. Among the various types of digital image forgeries, copy-move is a common image tampering. The type of forgery is occurring by part of the image is taken and placed in the same picture this type named copy-move forgery. Since source and target regions are same properties such as noise, illumination condition, color temperature etc. will be compared between source and target regions. The forgery maybe done to hide some object or authenticity or may be to boost the visual effect of the image. By using image editing software such as Adobe Photoshop a forger can easily tamper the image and hide tamper trace, thus the image authenticity is lost.



(a) (b)

Fig. 1 the original image and forgery image

Mostly the forgery occurs by making some geometric transformations such as rotation, scaling etc. The forger may hide the tampering by noise addition, loss compression or blurring. The above operations are done to make copy-move forgery detection more difficult. The CMFD is a very important process in many areas such as medical imaging, criminal investigation, surveillance systems, transportation sector, scientific publications, intelligence services, financial document, etc. Fig. 1 shows a copy-move forgery appeared in different datasets as an example. Fig.1 (a) The original image the picture contains seven types of pistols, while The Fig.1 (b) is a forged image contains eight types of pistols, a pistol was added as shown in red circle the in Fig.1 (b).

Manuscript received June 5, 2019
Manuscript revised June 20, 2019

## 2. Literature Studies

In general, the detection forgery of copy-move, called CMFD, this technique can be classified to block-based and key-point-based approaches as given in [1] and [2], and recently they are many new techniques based on segmentation.

### 2.1 Block-based Approaches

The authors in [3], and [4] were the Earliest developed of the block-based methods. The method in [3] It has been dependent on DCT coefficients, while method in [4] It has been dependent on PCA. An efficient block-based method used Zernike moments was presented in [5], this method expose copy rotate move forgery, it was start to be poor to contra scaling and affine conversion. The method in [6] was propose by rotating blocks and polar sine convert, this process is demanded It must be similar durability as in [5], with ability to expose fraud with scaling and spectacle convert. The way was proposed in [4],[5] where the direct block comparison Inability to extracting any type of features. This method compare each block with all blocks of the same bucket, then blocks were be grouped into buckets. The manner heading the issue of unimportant oppress or skinny lightening of the copy area before existence pasted. key-point-based approaches.

### 2.2 Key-point based Approach

For complexity of subdivion block system, the author go to use a new method in CMFD, this method based on key-point extraction and feature matching. It depends to find place where the maximum number of entropy regions in image and represent them by using feature vector as key-points. where the amount of feature vector few, Accounts complication will be few than block-based process. There are many general systems were present in [7] evaluated key-point-based process. The rotation, scaling, and shearing can be estimated using this method. there are many methods using SIFT features was improved in [8] to grip numerous fraud of copy-move. The new method proposed in [9] to solve the problem of false matching.

### 2.3 Segmentation-based Approach

The image can segment into meaningful regions. The author in [10] are used superpixels Simple Linear Iterative Clustering (SLIC) algorithm and make comparing about three different image segmentation methods, to extracted features of SIFT and to make over-segment form each segment of the image, first built a k-d tree then find the matching between patches bay used the k-nearest neighbor. The author in [10] different sizes were used for fragmentation by SLIC are used. The author in [11] presented a study taking a rotation invariant DAISY descriptors and segmentation based approach to discover copy-move forgery. In this paper, they are three methods used the same as followed in [10] to get image segmention. Many research as [8], [9] are used SLIC algorithm, which this algorithm is used to segment the image to abnormal areas and not nested, seeing that this algorithms can collection all pixels to the main aria the image. The author in [12] make a Study and Comparisons between many type of CMFD Scheme, which the Segmentation is the main process in this methods.

## 3. Proposed Scheme

In this section we introduce new system for CMFD, this system collects the criteria of block and keypoint feature and it have integrated in this scheme, this scheme will handgrip the problem of lack of key-points when the forgery occurs in textureless area by using CMFD-SIFT

### 3.1 Related Techniques

The techniques that we used in our system are as follows:
1) Multi-scale feature extraction and adaptive matching (MSFE): The author in [13], proposed a new system for CMFD this system Contain three steps, The results show that, this system works much better compared to previous methods.
2) Reveal method based on CMFD-SIFT: In this type as in [14], the author gives a novel method for CMFD, this method was modified SIFT-based detector the Key-points are detected, while duplicated regions can accurately detect by this method. In our scheme and our previous scheme [15] we will integrate the above two methods in one scheme to get the good results for lack of keypoint the textureless area.
The proposed scheme using Modified multi-scale feature extraction and matching integrates the characteristics of both block features and keypoint features and performs very well when there are multiple copy-move objects/regions and especially when the objects/regions are of different sizes and contain both smoothed and detailed textures, Fig .2 shows our proposed scheme. The steps of our scheme are introduce as follows:
**STEP-1**: The Modified Multi-Scale Feature Extraction (MMSFE) segment the image into three scale, then apply the CMFD-SIFT to development the previous system.
**STEP-2**: We used and apply the APM algorithm as introduce in [12], that to obtain the matched aria in the image, which the conformity Keypoints are calculated.
**STEP-3**: Depend to conformity Matched Keypoints algorithm, we can determine the forgery Areas.

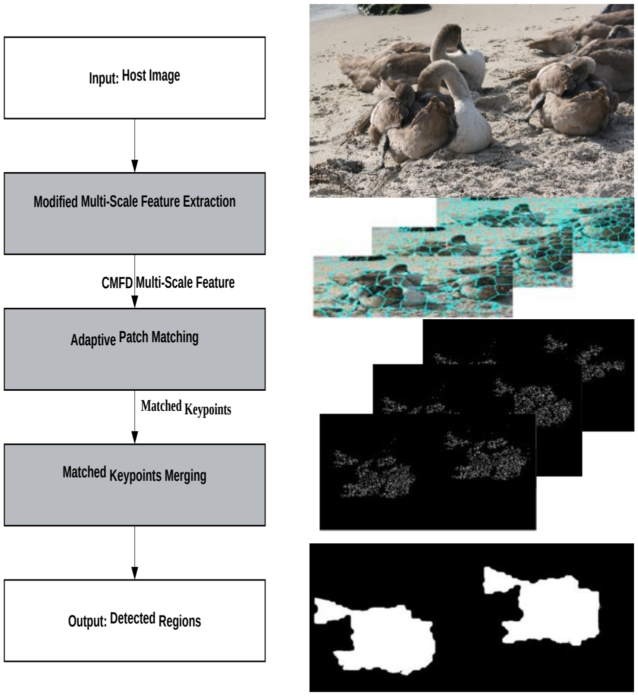


Fig. 2 The framework of the proposed copy-move forgery detection scheme

## 3.2 Modified Multi-Scale Feature Extraction (MMSFE)

In this part we will explain the main parts of (MMSFE) as shown in Fig.3 First, we discuss the Superpixel, then the Modified- Feature Extraction.

1) Superpixel Segmentation: In our scheme the (MMSFE) algorithm will segment the image to the areas with three different scales. from each areas we can extracted the feature.

2) Feature Extraction CMFD-SIFT: We choose CMFD-SIFT to modified the proposed scheme, the key-points it can be extract and located by feature-based algorithms. By using scale-space representation we can detect different scales of the SIFT feature.

The Gaussian smoothing are used to obtained The pyramid levels but key-points in the scale-space will be chose as local extrema. The CMFD approaches is depends on several steps:

**STEP-1**: Key-point detection: The key-points uniformity measurement φ is calculated by two steps:

1) Define K(i, j) as a matrix, then we set T as a number of discover key-point in I as image; number S is standard key-points then studied (e.g., see Eq. 1):

$$S = T / \left( \lceil M/n \rceil \times \lceil N/n \rceil \right) \tag{1}$$

2) The value of φ is studied (e.g., see Eq. 2):

$$\varphi = \sqrt{ \sum_{i=1}^{\lceil M/n \rceil} \sum_{j=1}^{\lceil N/n \rceil} \left( K_{i,j} - s \right)^2 / T } \tag{2}$$

**STEP-2**: Key-point distribution: The sub-image $S_{ij}$ must be selected from an image I, There is a temporary list in order to be saved the key-points as list Lt = [p1, 1, p1 , 2, …, p2 , 1, p2 , 2,…].

**STEP-3**: Key-point description.

In the selected region the orientation and magnitude of the image gradient computed (e.g., see Eq. 3 and 4):

$$m(x1, x2) = \sqrt{ I_{x1}(x1, x2)^2 + I_{x2}(x1, x2)^2 } \tag{3}$$

$$\theta(x, y) = \tan^{-1}\left( \frac{I_{x2}(x1, x2)}{I_{x1}(x1, x2)} \right) \tag{4}$$

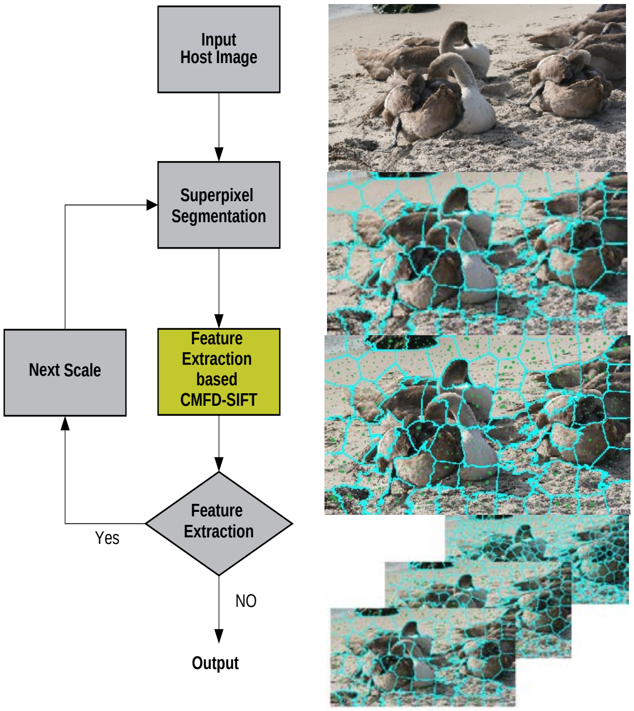


Fig. 3 Three different scales Feature extraction flowchart.

The Modified Multi-Scale Feature Extraction (MMSFE) Algorithm calculated by the following steps where the input is Host image and the output is Modified Multi-Scale Feature MMSF.

**STEP-1**: Load the host image and initialize the initial scale n=1, the initial number of blocks $B_n$=B, the initial set of patch feature, $PF^n = \emptyset$, and the initial set of multi-scale feature $MSF = \emptyset$.

**STEP-2**: Apply the SLIC algorithm to segment the input image into $B^n$ patches $P^n$,

$$P^n = \left\{ P_1^n, P_2^n, P_3^n, P_4^n, P_5^n, \dots\dots\dots, P_{B_n}^n \right\}$$

**STEP-3**: Apply CMFD SIFT algorithm to each patch to extract feature points $F^n$,

$$F^n = \left\{ F_1^n, F_2^n, F_3^n, F_4^n, F_5^n, \dots\dots\dots, F_{B_n}^n \right\}$$

**STEP-4**: Organize the set of patch feature $PF^n = \{P^n, F^n\}$; and the set of multi-scale feature MSF as $MSF = MSF \cup PF^n$

Check the existence of the extracted feature points $F^n$, $F^n \neq \emptyset, n = n + 1$

$B_n = 4^{n-1} * B_{n-1}$, repeat STEP-2 to STEP-4, otherwise, output the set of multi-scale feature *MSF*, $MSF = \{PF^1, PF^2, PF^3, \dots\dots, PF^n\}$

In STEP-1 of (MMSFE) Algorithm, the appreciate initialization of B can avoid segmenting the host image into excessive scales. In the experiments, by experiments, the B is initially set as 200 when the size of host image M × N is larger than1500×1500; otherwise, the B is initially set as 100.

## 3.3 Adaptive Patch Matching Algorithm

After the description procedure, the expressive paths for all key-points are produced. Assumed a trial image I, a regular of key-points X = (x1, x2,…, xn) with their consistent descriptors F = (F1, F2, …, Fn) is mined.

to improving the existing matching process we used Adaptive spot Matching algorithm. by [13], then it used by modification located of the threshold. In Fig. 4 we show the diagram of the APM. The $i^{th}$ measure ($i \in$ 1,2,3), the number of corresponding keypoints of each spot pair is counted agree to $PF^i = [P^i, F^i]$ and the correlation coefficient map $CC^i$ will generate; then the identical spot threshold $TP^i$ specified modification; the identification spot pairs MPi will be situated by $TP^i$; and finally the conformity keypoints $MK^i$ will be chosen from $MP^i$.

The steps of the Adaptive patch matching
algorithm are explained as shows:

**STEP-1**: Load the Multi-Scale Feature $MSF=[PF^1,PF^2,\dots\dots PF^n]$, where n means the number of scales, $PF^n=[P^n,F^n]$ is the set of patch feature.

**STEP-2**: In each scale, calculate the numbers of matched keypoints between each two patches, which are defined as correlation coefficient of the corresponding patch pair; and thus generate the correlation coefficient map $CC=[CC^1,CC^2,\dots..CC^n]$.

**STEP-3**: According to CC, adaptively calculate the value of patch matching threshold as $TP=[TP^1,TP^2,\dots\dots,TP^n]$.

**STEP-4**: According to the corresponding matching threshold TP, locate the matched patch pairs MP as $MP=[MP^1,MP^2,\dots\dots..MP^n]$.

**STEP-5**: Extract the matched keypoints MK in MP as $MK=[MK^1,MK^2,\dots.MK^n]$.

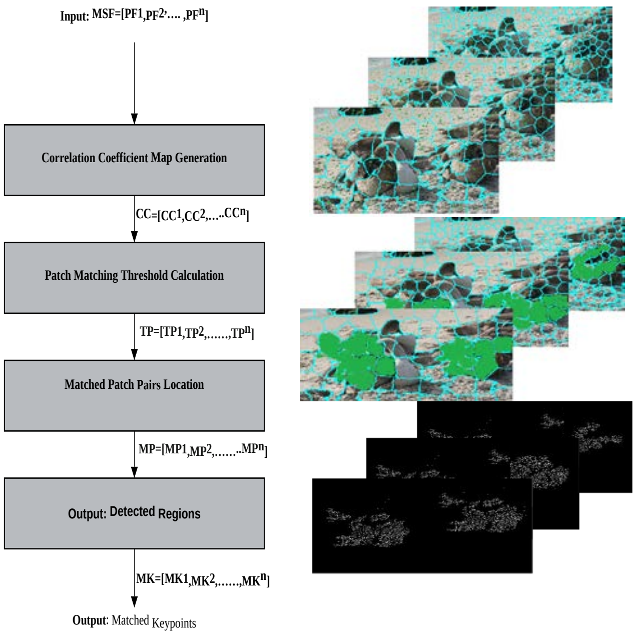

Input: MSF=[PF1,PF2,.... ,PFn]
Correlation Coefficient Map Generation
CC=[CC1,CC2,.....CCn]
Patch Matching Threshold Calculation
TP=[TP1,TP2,......,TPn]
Matched Patch Pairs Location
MP=[MP1,MP2,.......MPn]
Output: Detected Regions
MK=[MK1,MK2,......,MKn]
Output: Matched Keypoints

Fig. 4 Flowchart of the Adaptive Patch Matching (APM) algorithm.

In STEP-2 of Algorithm APM, the keypoints are matched using the best-bin-first algorithm with their Euclidian distance, which means that a keypoint $f_a$ is matched to the keypoint $f_b$ only if they can meet the following condition. $d(f_a, f_b) \cdot TK \leq d(f_a, f_i)$ , Where $d(f_a, f_b)$ means the Euclidian distance between the keypoints $f_a$ and $f_b$, and it is defined in as shown above; $d(f_a, f_i)$ means the Euclidian distances between the keypoints $f_a$ and all other keypoints, and it is defined in above formula. TK is the keypoints matching threshold; when TK becomes larger, the matching accuracy will be higher, but meanwhile the ratio outliers will be higher accordingly, which will cause greater miss probability. Therefore, in the experiments, we set TK= 2 by experiments to provide a good trade-off between matching accuracy and miss probability.

$$d(f_a, f_b) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$$

$$d(f_a, f_b) = \sqrt{(x_a - x_i)^2 + (y_a - y_i)^2}$$

$$, i = 1,2, \dots \dots n; i \neq a, i \neq b$$

Correlation coefficient means the number of matched keypoints between the two patches Assuming there are Bi patches in the $i^{th}$ scale, we can generate $t = B_i(B_i - 1)/2$ correlation coefficients, which form the correlation coefficient map $CC^i$. After generating $CC=[CC^1,CC^2,\dots..CC^n]$, we need to calculate the patches matching threshold TP as stated in STEP-3 of APM Algorithm.

The procedures of the adaptive calculation of the patch matching threshold TP in each scale are explained and will be calculated based on [13] by the following steps:

STEP-1: Sort the correlation coefficients in ascending order as $CC\_S^i = [CC^i_1, CC^i_2, \dots. CC^i_t]$

where i means in the ith scale, and t means the number of correlation coefficients in the corresponding scale, $t = B_i(B_i - 1)/2$ , and filter out the repeated correlation coefficients as
$CC\_F^i = [CC_1^i, CC_{2,}^i, \dots . CC_f^i]$, where$f \leq B_i(B_i - 1)/2$.
STEP-2: Calculate the first derivative of $CC\_F^i, \nabla(CC\_F^i)$, the mean value of the first
derivative vector, $\overline{\nabla(CC\_F^i)}$, and the second derivative of $CC\_F^i, \nabla^2(CC\_F^i)$.
STEP-3: Select the correlation coefficients $CC\_F_j^i$, of which their second derivative is larger
than the mean value of the corresponding first derivative vector, as defined (e.g., see Eq. 5).

$$\nabla^2(CC\_F_j^i) > \overline{\nabla^2(CC\_F^i)} \tag{5}$$

STEP-4: Extract the minimum value from $CC\_F_j^i$ and set its correlation coefficient value as the corresponding patch matching threshold TP$^i$.
After calculating the patch matching threshold of each scale adaptively, we can locate the matched patch pairs in each scale if their correlation coefficients are larger than the corresponding matching threshold. From those matched patches in each scale, we selected the matched keypoints to form the Matched Keypoints (MK).

### 3.4 Matched Keypoints Merging Algorithm

After obtaining the matched keypoints MK, we need to determine the forgery regions by turning the independent pixels/keypoints into regions. Fig 5. shows the flowchart of the MKM algorithm.

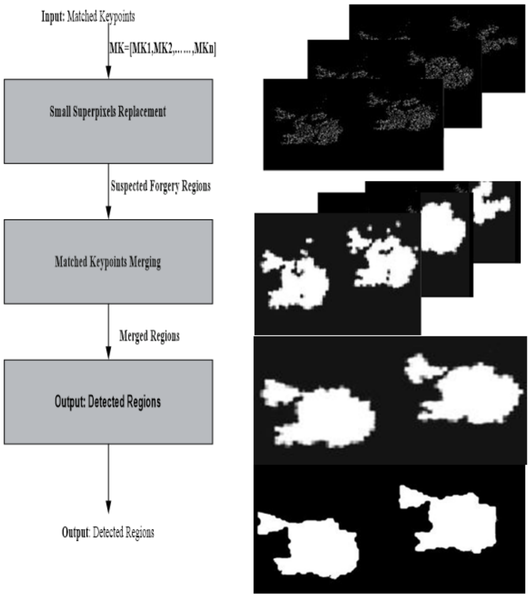


Fig. 5 Flowchart of the Matched Keypoints Merging (MKM) algorithm

First, the host image is segmented into small superpixels; then, MK are replaced by the small superpixels to form the suspected forgery regions. The size of small superpixels is related with the size of the host image; when the host image is of higher resolution, the size of small superpixels will be larger. In our test dataset, the average size is approximate3000 × 2000, therefore, we set the size of small superpixel as 20 by experiments. Next, the suspected forgery regions in all scales are merged. If the suspected forgery regions are merged together by using 'OR' operation, the miss rate of the forgery detection will be reduced, however, the probability of error detection will be bigger. Therefore, we need to filter out some regions which may be wrongly detected during the merging process.

## 4. Experimental Study

In this section, we present the results of the proposed copy-move forgery detection approach. For this purpose, an experimental version of the proposed method was implemented in Matlab2017a. The algorithm is coded in MATLAB 2017a on a machine equipped with Intel i5 2.2GHz CPU with 4GB DDR2RAM.

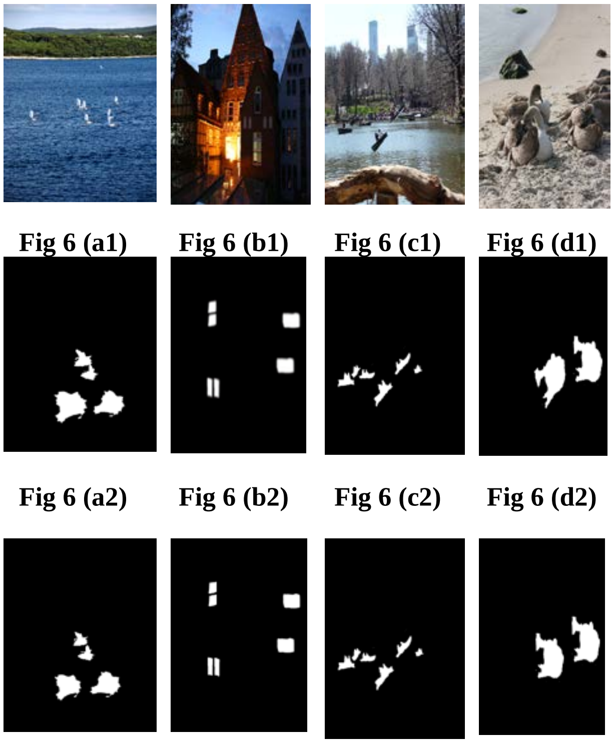


**Fig 6 (a1)** **Fig 6 (b1)** **Fig 6 (c1)** **Fig 6 (d1)**

**Fig 6 (a2)** **Fig 6 (b2)** **Fig 6 (c2)** **Fig 6 (d2)**

**Fig 6 (a3)** **Fig 6 (b3)** **Fig 6 (c3)** **Fig 6 (d4)**

Fig. 6 The copy-move forgery detection results of the proposed scheme. The 1st row: the four selected images in the dataset; 2nd row: ground truth images; 3rd row: The detected forged regions

The experimental and results show that the proposed scheme performs much better than the existing state-of-the-art copy-move forgery detection algorithms, even under various challenging conditions, including the geometric transforms, such as scaling and rotation, and the common signal processing, such as JPEG compression and noise addition; in addition, the special cases such as the multiple copies and the down-sampling are also evaluated, the results indicate the very good performance of the proposed scheme. The experiments are conducted to evaluate the effectiveness and robustness of the proposed copy-move forgery detection scheme.

In the following experiments, the benchmark dataset which consists the realistic copy-move forgeries is used to test the proposed scheme. Fig 6. (a1), (b1), (c1) and (d1) shows a selection of images from the dataset.

The dataset comprises 48 uncompressed PNG true color images. The average size of forgery regions is about 6% of each image. These images have a size of 3000×2300 pixels. The copied regions are of categories of living, nature, man-made and even mixed, and they range from smooth to highly texture; the copy-move forgeries are created by copying, scaling and rotating semantically meaningful image regions. JPEG compression and down-sampling are also added on the forgery images; in addition, the combined transformations and multiple copies forgeries are included in the image dataset. Therefore, we choose this dataset to objectively evaluate our scheme. Fig 6 (a2), (b2), (c2) and (d2) shows the ground truth for the image selection. Fig 6 (a3), (b3), (c3) and (d1) shows the copy-move forgery detection results of the proposed scheme.

In order to evaluate the performance of the proposed scheme, the two characteristics precision and recall [9] are calculated (e.g., see Eq. 6) and (e.g., see Eq. 7) respectively. We also give the F-Measure [7], (e.g., see Eq. 8), as a measure which combines the precision and recall in a single value.

$$\text{Precision} = \frac{|\Omega \cap \grave{\Omega}|}{|\Omega|} \tag{6}$$

$$\text{Recall} = \frac{|\Omega \cap \grave{\Omega}|}{|\grave{\Omega}|} \tag{7}$$

Where Ω means the set of forgery regions detected by the proposed scheme for the dataset; and $\grave{\Omega}$ means the set of all forgery regions for the dataset.

$$\text{F} - \text{Measure(F1)} = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}$$

To reduce the effect of random samples, the average precision/recall is computed over all the images in the dataset. Since Christlein et al. [7] have particularly recommended all benchmark methods, we use the dataset they provided and compare our experimental results with several state-of -the-art algorithms: the SIFT based detection method [7], which combined the methods of [8]; the SURF based detection method[7]; Zernike moments based forgery detection method [16]; the method proposed by Bravo [17]; the SBFD method proposed in [16]; and the ASFPM method [18] which we have proposed in our previous work. We mainly compare the performances of our scheme with the state-of -the-art algorithms under different scenarios: the plain copy-move forgery; the forgery with distortion by various attacks including: scaling, rotation, Gaussian noise addition JPEG compression, and even combined attacks; the multiple copies forgery and the down-sampling forgery.

### 4.1 Detection Results under Plain Copy-Move Forgery

Basically, we firstly evaluate the proposed scheme when under the ideal condition, that is the plain copy-move forgery. We have 48 original images and 48 forgery images, where one to one copy-move forgery is implemented. The detection methods distinguish the original images from the forgery images in this case. We evaluate the scheme at both pixel level and image level, and Table 1 and Table 2 show the detection results of the 96 images at the image level and the pixel level, respectively. In general, higher precision as well as higher recall indicates the superior performance.

Table 1: Detection results of the plain copy-move forgery at the image level

| ***Image level*** | ***precision (% )*** | ***recall ( % )*** | ***F ( % )*** |
|---|---|---|---|
| **SIFT** [7] | 88.37 | 79.17 | 83.52 |
| **SURF** [7] | 91.49 | 89.58 | 90.52 |
| **Zernike** [19] | 92.31 | 100.0 | 96.00 |
| **Bravo** [17] | 87.27 | 100.0 | 93.20 |
| **SBFD** [16] | 70.16 | 83.33 | 76.18 |
| **ASFPM** [18] | 96 | 100.0 | 97.96 |
| **MSFPM** [13] | 90.57 | 100.0 | 95.05 |
| **My Scheme** | **92.22** | **100.0** | **93.16** |

Table 2: Detection results of the plain copy-move forgery at the pixel level

| **Pixel level** | **precision(%)** | **recall(%)** | **F( % )** |
|---|---|---|---|
| **SIFT [7]** | 60.80 | 71.48 | 65.71 |
| **SURF [7]** | 68.13 | 76.43 | 72.04 |
| **Zernike [19]** | 95.07 | 87.72 | 91.25 |
| **Bravo [17]** | 98.81 | 82.98 | 89.34 |
| **SBFD [16]** | 84.90 | 54.095 | 65.16 |
| **ASFPM [18]** | 89.195 | 83.73 | 86.38 |
| **MSFPM [13]** | 95.22 | 90.6 | 92.85 |
| **My Scheme** | **95.88** | **90.8** | **93.15** |

In Tables 1. and 2, the results in bold indicate the results of the proposed scheme and the results in bold and italic indicate the best ones. It can be easily seen that My scheme can achieve 92.22% precision and meanwhile 100% recall, which performs better than the most of existing state-of-the-art methods at image level, except the Zernike moments based method [16] which can achieve precision up to 92.31%

and recall up to 100%. Meanwhile, the advantage of the proposed Modified multi-scale detection method is particularly prominent at pixel level, when comparing with the existing state-of-the-art methods, as indicated in Table 2. The proposed method achieves precision up to 95.88% and recall up to 90.8%, which is much better than the existing state-of-the-art methods. The results indicate the good accuracy of My proposed copy-move forgery detection scheme by using Modified multi-scale feature extraction and matching.

## 4.2 Detection Results under Various Attacks.

Besides the one to one plain copy-move forgery, we also test My proposed scheme when the copied regions are attacked by various attacks including geometric distortions, image degradations, and even combined attacks. That means, the forgery images are generated by using each of the 48 images in the dataset, and the copied regions are attacked by attacks as follows:

1) **Scaling:** The copied regions are scaled with the scale factor varies from 90% to 110%, with the step as 2%, as shown in Fig (7, 8 and 9). In this case, we need to test totally 48 × 10 = 480 images. In Fig (7, 8 and 9), show the results under the scaling at the pixel level, where the x-axis indicates the scale factor. Where the results indicated in blue show the results of the proposed scheme**.**

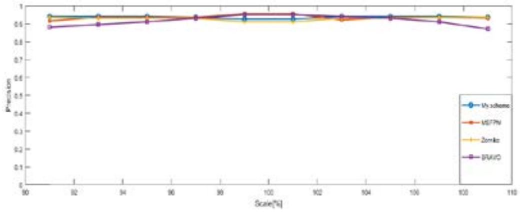


Fig. 7 Detection results under scaling with Precision.

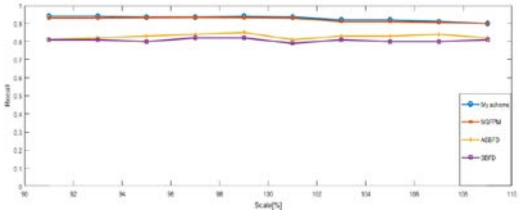


Fig. 8 Detection results under scaling with Recall.

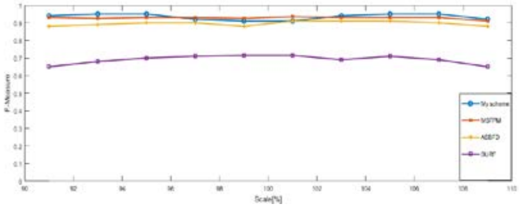


Fig. 9 Detection results under scaling with F-Measure.

2) **Rotation:** The copied regions are rotated with the rotation angle varies from 2° to 10°, in step of 2°, as shown in Fig (10, 11 and 12). In this case, we need to test totally 48 × 5 = 240 images. In Fig (10, 11 and 12), show the results under the rotation at the pixel level, where the x-axis indicates the rotation angle. Where the results indicated in blue show the results of the proposed scheme.

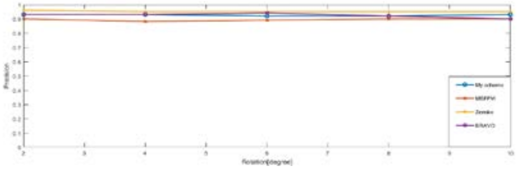


Fig. 10 Detection results under rotation with Precision.

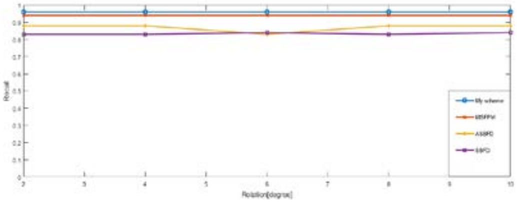


Fig. 11 Detection results under rotation with Recall.

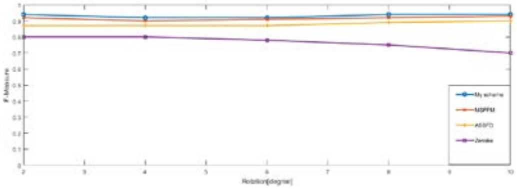


Fig. 12 Detection results under rotation with F-Measure.

3) **Gaussian Noise addition:** The image intensities are normalized between 0 and 1 and added zero-mean Gaussian noise with standard deviations of 0.02, 0.04, 0.06, 0.08 and 0.10 to the inserted snippets before splicing, as shown in Fig (13, 14 and 15). In this case, we need to test totally 48 × 5 = 240 images.

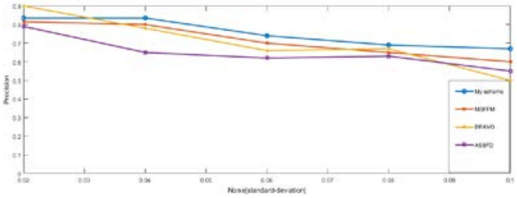


Fig. 13 Detection results under the Gaussian Noise addition with Precision.

In Fig (13, 14 and 15), show the results under the Gaussian Noise addition at the pixel level, where the x-axis indicates the standard deviations. Where the results indicated in blue show the results of the proposed scheme.

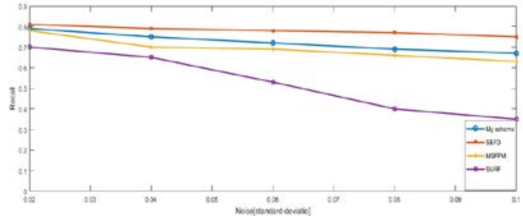


Fig. 14 Detection results under the Gaussian Noise addition with Recall.

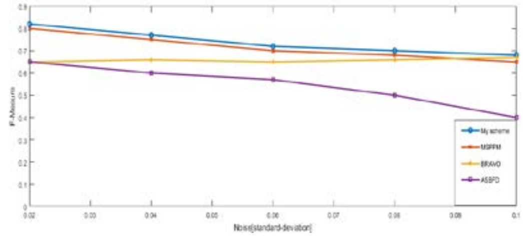


Fig. 15 Detection results under Gaussian Noise addition with F-Measure.

4) **JPEG compression:** The JPEG compression is applied to the forgery images and original images, with the qualify factor varies from 100 to 20, with the step as -10, as shown in Fig (16, 17 and 18). In this case, we need to test totally 48 × 9 = 432 images. In Fig (16, 17 and 18), show the results under the JPEG compression at the pixel level, where the x-axis represents the quality factor. Where the results indicated in blue show the results of the proposed scheme.

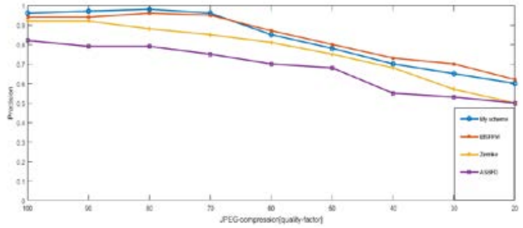


Fig. 16 Detection results under the JPEG compression with Precision.

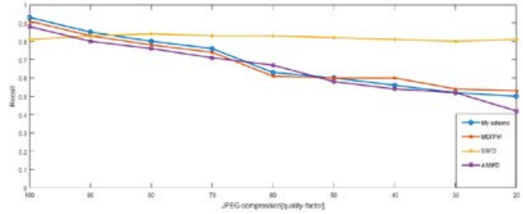


Fig. 17 Detection results under the JPEG compression with Recall.

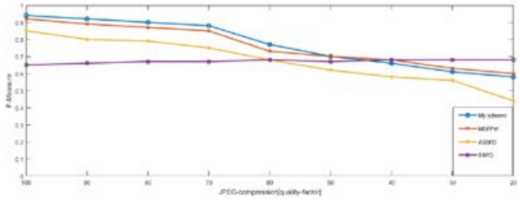


Fig. 18 Detection results under the JPEG compression with F-Measure.

5) **Combined transforms:** The combined transforms are applied into the copied regions to evaluate the proposed scheme: In this case, we use totally 48 × 6 = 288 images. The detection results under various attacks are displayed in Fig (19, 20 and 21). In Fig (19, 20 and 21), indicate the precision rate, recall rate and F-Measure, respectively and where the results indicated in blue show the results of the proposed scheme. In Fig (19, 20 and 21), show the results under the combined transforms at the pixel level.

We compare the proposed scheme with the existing state-of-the-art methods, it can be seen from the Fig (7, 8 and 9), and Fig (10, 11 and 12), all the precision, recall, and F-Measure of the proposed scheme are greater than 92%, which indicates that the proposed scheme performs much better than the existing state-of-the-art forgery detection methods under the geometric transforms. As well, the proposed scheme performs well under the common signal processing such as Gaussian Noise addition and JPEG compression, as shown in the Fig (13, 14 and 15), and Fig (16, 17 and 18). Note that, although our recall rates are worse than which of the SBFD method, the F-Measure are better than it under the Gaussian Noise addition and the JPEG compression.

In Fig (19, 20, and 21) the proposed scheme is evaluated under six combined attacks we defined. It is obviously that the proposed scheme performs much better than the other methods.

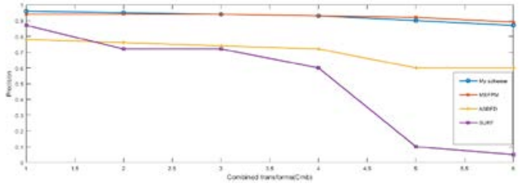


Fig. 19 Detection results under combined transforms with Precision.

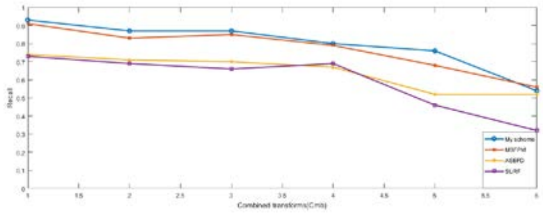


Fig. 20 Detection under combined transforms with Recall.

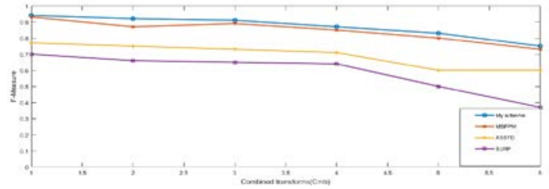


Fig. 21 Detection results under combined transforms with F-Measure.

6) **Detection results under multiple copies and down-sampling**: Besides the plain copy-move forgery and the forgeries attacked by various attacks, we also evaluate the proposed scheme when the forgery images have multiple copies. In order to test the multiple copies forgery, we have copied an 64 × 64 image region five times and moved them to the random locations in the image itself.

Table 3 shows the comparison of the detection results in this scenario. It can be easily seen that the proposed scheme outperforms the most of existing detection methods except the method proposed by Bravo [17] which can achieve precision up to 88.75%, however, our scheme can achieve much higher recall. The results indicate the good performance of the proposed Modified multi-scale feature extraction and the adaptive matching for copy-move forgery detection.

Table 3: Detection results under the multiple copies forgery at the pixel level

| Pixel level | precision ( % ) | recall ( % ) | F ( % ) |
|---|---|---|---|
| **SIFT** [7] | 11.37 | 4.95 | 6.90 |
| **SURF** [7] | 37.49 | 21.86 | 27.62 |
| **Zernike** [19] | 83.15 | 22.00 | 34.79 |
| **Bravo** [17] | 88.75 | 58.27 | 67.58 |
| **ASFPM** [18] | 50.91 | 47.63 | 49.22 |
| **MSFPM** [15] | 58.2 | 73.2 | 64.83 |
| **My Scheme** | 59.3 | 74.22 | 66.52 |

Considering that the performance of forgery detection algorithms usually matters with the quality of the resources, we evaluate the proposed scheme and compare it with the mentioned state-of-the-art methods under the down-sampling, as shown in Fig (22, 23 and 24), where Fig (22, 23 and 24) display the precision, recall and F-Measure, respectively. We scale down all the images in the plain copy-move forgery in step of 20%. Note that the parameters of detection methods are globally fixed to avoid over-fitting. In Fig (22, 23 and 24), the x-axis means the down-sampling factor and the results in blue indicate which of the proposed scheme while the results in other colors indicate which of the above-mentioned state-of-the-art methods. The proposed scheme performs much better than the existing methods in this case.

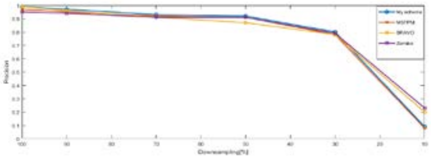


Fig. 22 Detection results under down-sampling with Precision.

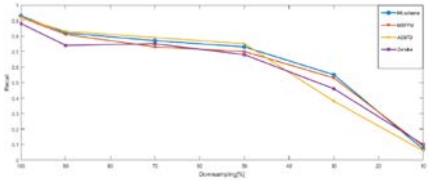


Fig. 23 Detection results under down-sampling with Recall.

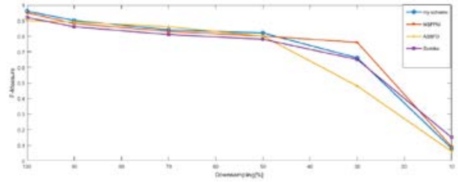


Fig. 24 Detection results under down-sampling with F-Measure.

## 5. Conclusions

Experimental results show that the proposed scheme performs much better than the existing state-of-the-art copy-move forgery detection algorithms, even under various challenging conditions including: the geometric transforms, such as scaling and rotation; and the common signal processing, such as JPEG compression and noise addition. In addition, the special cases such as the multiple copies and the down-sampling are also evaluated and the results indicate the very good performance of the proposed scheme.

## 6. Future work

Deep learning uses neural networks to learn useful representations of features directly from data. Perform supervised learning with series and Directed Acyclic Graph (DAG) Convolutional Neural Networks (CNNs or ConvNets) for classification and regression. Recent advances in deep learning have improved to the point where deep learning outperforms humans in some tasks like classifying objects in images. We will use deep learning (semantic segmentation) to detect and localize CMF. Semantic segmentation describes the process of associating each pixel of an image with a class label, (such as CMF parts, unforged part).

## References

[1] A. Kashyap, R. S. Parmar, M. Agrawal, and H. Gupta, "An evaluation of digital image forgery detection approaches," arXiv preprint arXiv:1703.09968, 2017.

[2] Z. Zhou, Y. Wang, Q. J. Wu, C.-N. Yang, and X. Sun, "Effective and efficient global context verification for image copy detection," IEEE Transactions on Information Forensics and Security, vol. 12, pp. 48-63, 2017.

[3] A. J. Fridrich, B. D. Soukal, and A. J. Lukáš, "Detection of copy-move forgery in digital images," in in Proceedings of Digital Forensic Research Workshop, 2003.

[4] A. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image region [Technical Report]. 2004-515," Hanover, Department of Computer Science, Dartmouth College. USA, p. 32, 2004.

[5] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using zernike moments," in International Workshop on Information Hiding, 2010, pp. 51-65.

[6] L. Li, S. Li, H. Zhu, and X. Wu, "Detecting copy-move forgery under affine transforms for image forensics," Computers & Electrical Engineering, vol. 40, pp. 1951-1962, 2014.

[7] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," IEEE Transactions on information forensics and security, vol. 7, pp. 1841-1854, 2012.

[8] 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, pp. 1099-1110, 2011.

[9] Y. Zhu, X. Shen, and H. Chen, "Copy-move forgery detection based on scaled ORB," Multimedia Tools and Applications, vol. 75, pp. 3221-3233, 2016.

[10] K. Minakshi, "Digital Image Processing: In: Satellite Remote Sensing and GIS Applications in Agricultural Meteorology," World Meteorological Organization Publishing, pp. 81-102, 2003.

[11] R. Sekhar and R. Shaji, "A study on segmentation-based copy-move forgery detection using DAISY descriptor," in Proceedings of the International Conference on Soft Computing Systems, 2016, pp. 223-233.

[12] M. Ikhlayel, M. Hariadi, and K. E. Pumama, "A Study of Copy-Move Forgery Detection Scheme Based on Segmentation," International Journal of Computer Science and Network Security, vol. 18, p. 27, 2018.

[13] X. Bi, C.-M. Pun, and X.-C. Yuan, "Multi-scale feature extraction and adaptive matching for copy-move forgery detection," Multimedia Tools and Applications, vol. 77, pp. 363-385, 2018.

[14] B. Yang, X. Sun, H. Guo, Z. Xia, and X. Chen, "A copy-move forgery detection method based on CMFD-SIFT," Multimedia Tools and Applications, vol. 77, pp. 837-855, 2018.

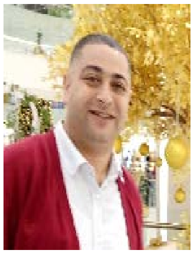
[15] M. Ikhlayel, M. Hariadi, and I. K. E. Pumama, "Modified Multi-scale Feature Extraction for Copy-Move Forgery Detection Based on CMD-SIFT," in 2018 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM), 2018, pp. 260-264.

[16] 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, pp. 507-518, 2015.

[17] P. Kakar and N. Sudha, "Exposing postprocessed copy–paste forgeries through transform-invariant features," IEEE Transactions on Information Forensics and Security, vol. 7, pp. 1018-1028, 2012.

[18] C.-M. Pun, X.-C. Yuan, and X.-L. Bi, "Image forgery detection using adaptive oversegmentation and feature point matching," IEEE Transactions on Information Forensics and Security, vol. 10, pp. 1705-1716, 2015.

[19] S.-J. Ryu, M. Kirchner, M.-J. Lee, and H.-K. Lee, "Rotation invariant localization of duplicated image regions based on Zernike moments," IEEE Transactions on Information Forensics and Security, vol. 8, pp. 1355-1370, 2013.



**Mohammed Ikhlayel** is a Faculty member at Al-Quds Open University (QOU). He was Action Head of Department of Information Technology and Communications from 2009 until 2012, in QOU. Mr. Ikhlayel received his Bachelor degree in electrical engineer (Communications) in 2004, he received his master degree in Communications in 2007 from Egypt, and he Studies PhD in electrical engineering at Institut Teknologi Sepuluh Nopember in indonisia. (e-mail: missaikhlayel8080@gmail.com)

**Mochamad Hariadi** He is academic staff of Electrical and Computer Engineering Department Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. He is the project leader in joint research with PREDICT JICA project Japan.(e-mail: mochar@te.its.ac.id).

**I Ketut Eddy Pumama**, He is the Head of Computer Engineering Department Institut Teknologi Sepuluh Nopember, Surabaya, resarch intrsted in computer vision and understanding, Medical Image Analysis and Microscopic Image Analysis. Currently, he is academic staff of Computer and Electrical Engineering Department Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. (e-mail: ketut@te.its.ac.id).