

Implementation of feature extraction algorithms for image tampering detection

Nooraini Yusoff^{1*} and Loai Alamro²

Associate Professor, School of Computing, UUM College of Arts and Sciences, Universiti Utara Malaysia¹

Assistant Lecturer, Al Turath University College, Al Mansour, 10011 Baghdad, Iraq²

Received: 12-September-2018; Revised: 22-November-2018; Accepted: 30-January-2019

©2019 Nooraini Yusoff, and Loai Alamro. This is an open access article distributed under the Creative Commons Attribution (CC BY) License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

There are three main steps to detect fake images, namely feature extraction, feature matching, and feature masking based on the similarity between two or more images, or parts of images. The detection accuracy of fraudulent images highly depends on the feature extraction. Thus, the quality of the extracted features plays a key role in the image forgery detection. Multiple feature extraction methods have been proposed for detecting fake images with different level of success, however, not many have the ability to extract geometrical transformation such as rotation and scaling. Hence, to overcome the issue, this paper presents two separate groups of feature extraction, namely the dimensional reduction and keypoint. Initially, we ran a series of experiments to reveal the best feature extraction methods, involving five methods, from both groups in detecting copy-moved image forgeries. Then in our experiments, we implemented the integration of singular value decomposition (SVD) and speeded up robust features (SURF), discrete cosine transform (DCT) and SURF, and discrete wavelet transform (DWT) and SURF, to formulate a more accurate and robust copy-move detection approach. The best performance of copy-move detection was achieved by deploying DWT and SURF. The integration of DWT and SURF solves the rotation and the scaling issues in copy-move image detection with higher high accuracy and shorter execution time.

Keywords

Image forgery, Dimensional reduction, Keypoint, discrete wavelet transform and Speeded up robust features.

1.Introduction

Nowadays, people live in an age of digital images where images consume the largest part of our digital world. With the massive development of photo editing software such as Photoshop and Corel PaintShop, modifying and tampering images have become very easy [1, 2]. Image tampering is defined as “adding, changing, or deleting some important features from an image without leaving any obvious trace” [2]. There have been various techniques used for forging an image. Based on the techniques used to create forged images, digital image forgery can be divided into three main categories: copy-move forgery, image splicing, and image re-sampling [3, 4]. Given the advanced algorithms used in forging, determining the authenticity and integrity of digital images is becoming a real challenge to the naked eye, as well as to machines. Thus, it is becoming very important to develop robust detection methods to identify image tampering operations.

In general, the efficiency and effectiveness of any detection, recognition, matching, and classification processes strongly depend on the size of the dataset and the features extracted [5, 6]. Thus, the quality of the extracted feature is one of the key issues [7]. The drawback of feature selection lies in possible information loss in the process of selecting relevant features. Besides, it is difficult to determine the optimal number of significant features [7, 8]. The dimensionality of image size is reduced after choosing a set of features from the original feature set in feature extraction [6, 9]. Unlike feature ranking-based selection, feature extraction takes the relationship among the distinctive features into account. Hence this offsets the drawbacks in feature selection [7, 9]. Several methods have been proposed to extract the most beneficial features, however, they have yet to achieve good performances. The main difference between these methods of detecting forgery is in the choice of specific features and adoption to detect fraud, which plays a role in the success of particular algorithms, and their

*Author for correspondence

performance in detecting and identifying fraudulent regions. In this study, initially, a comparative study was performed on five feature extraction methods that were identified, namely SVD, PCA, DCT, DWT, and SIFT, with SURF based feature extraction technique for detecting image forgery. Those are the most widely used methods in feature extraction and have been successfully applied to solve a wide range of problems such as object detection, shape detection, and facial recognition. We conducted a series of experiments on these methods to determine the performance in extracting features and reducing dimensions for each one.

Using the results from the comparison experiment, we also implemented the integration of the best feature extraction methods to find out which combined methods could extract the most robust and well representable features. This would contribute toward and enrich the image forgery field of study since digital images have been widely used as evidence or proof in many applications, including journalism, health, governmental, and crime investigations, which means their validity and originality is paramount.

2.Previous studies

Copy-move forgery has been addressed as an urgent problem. Several methods for detection of copy-move attacks were introduced. These methods are grouped into a block-based and keypoint methods. Under block-based group, Bin, et al. [10] introduced an efficient forensic approach for identifying copy-move activities. This method is built on DWT and fast Walsh–Hadamard transforms (FWHT) and it includes only two procedures. Firstly, DWT is utilized to reduce the image dimensionality. Secondly, FWHT is used for feature extraction to reduce the computational difficulty and false positive rate. This method has been widely used in processing images with additive Gaussian noise, blurs, and distortions. Bin et al. (2013) [10] specified that their method is accurate for JPEG compression represented by Q factor, where the Q factor ranges from 50% to 100%. However, the method is weak to detect transformation attacks.

Based on SVD and DCT, Zhao and Guo [11] developed a technique for copy-move manipulation. In this technique, the tested image is first split into overlapping blocks. Then two dimensional DCT is applied to every block. The DCT coefficients are then quantized to find a solid representation of every block. Consequently, the quantized blocks are split

into non-overlap sub-blocks, and SVD is used on every sub-block. Afterward, the features are computed to minimize the dimension of every block by using its biggest singular value. The feature-vectors are then lexicographically sorted. In the final step, the duplicated image blocks are matched with a predefined shift frequency threshold. This method can effectively detect a duplicated area even if the image is distorted by JPEG compression and blurring. Nevertheless, the transformation attacks (i.e., rotation and scaling) were not reported in the experimental results. By contrast, Sunil, et al. [4] introduced a method based on DCT and PCA. In this approach, the input image is mostly grayscale. The input image is then divided into blocks and DCT is applied to each block. The resulting row vector is saved with a zigzag order and stored within a matrix. DC is used to obtain samples, and PCA is applied to a matrix of the row vector to minimize the matrix dimensions. The authors claimed that this method has limitations in detecting scale and rotation.

Under the keypoint group, Huang, et al. [12] developed an approach depending on matching the SIFT feature descriptors, since the SIFT descriptors are invariant to lighting adjustments, noise, scaling, and rotations. The aim was to compare standard SIFT descriptors of earlier located keypoints in the tested image to detect the forged regions. However, this approach did not perform well in detecting small duplicated regions. Likewise, the authors in [13, 14]) presented a method based on SIFT to detect duplicated zones. This method begins with estimating the transform-matching without considering lighting and geometric transformations. The pixels for the counterfeit zones are then determined by subtracting the estimated transformation. The method showed good result with Gaussian noises and JPEG compression, but it failed to determine the credible keypoints in zones with minimal visual structures and is limited to detecting small zones with a slight keypoint. Meanwhile, Bo et al. [15] developed a method to determine copy-move regions using the SURF algorithm. This method extracts SURF keypoints from an image and then matches the SURF descriptors after dividing the extracted keypoints into two sets; determining nearest neighbors in such sets; saving the identical keypoints; and repeating the process for every sub-set until each and every one of them becomes unitary. The authors reported experiments with white Gaussian noise, blurring scaling, and rotation. However, the homogeneous segments challenged the proposed method and it

could not produce sufficient keypoints for suitable analysis.

3.Feature extraction and classification methods

This study divided the experimental feature extraction methods into two groups. These groups are classified into the dimensional reduction and keypoint. On the one hand, dimensional reduction is a technique that reduces the dimensionality of analyzed data. On the other hand, the keypoint methods extract the features from the interesting points of the whole image. All the methods under each group were performed, and the results of each experiment were discussed to formulate a strong method for detecting fraud images-several methods were integrated to achieve copy-move detection. An experiment was conducted to ascertain which method performed better in detecting forgery.

A.DWT

The DWT is a linear transformation that works on an information vector whose length is a whole (integer) number force of two, changing it into a numerically diverse vector of the same length [16]. It is a tool that differentiates information in diverse frequency segments, and afterward examines every segment with the determination coordinated to its scale. Based on Shukla and Tiwari [16], DWT is calculated with a series of filtering taken after by a factor-2 subsampling (*Figure 1*).

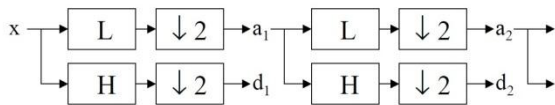


Figure 1 DWT Tree (reproduced from [17])

H and L indicates high and low-pass channels respectively, while 2 signifies subsampling. Outcomes of these filters are computed by the following Equations [18]:

$$a_{j+1}[P] = \sum_{n=-\infty}^{+\infty} l[n-2P]a_j[n] \quad (1)$$

$$d_{j+1}[P] = \sum_{n=-\infty}^{+\infty} h[n-2P]a_j[n] \quad (2)$$

Components a_j are utilized for next step (scale) of the transform and components d_j , called wavelet coefficients, locate the outcomes of the transform. Meanwhile, $h[n]$ and $l[n]$ are separate coefficients of

high and low-pass filters, respectively. One can accept that on scale $j + 1$, it is just half from value number of a and d components, on scale j . This means that DWT could be possible with just two a_j components staying in the analyzed signal, and these components are called scaling function coefficients [19]. DWT calculation for two-dimensional pictures is comparable. The DWT is computed firstly for all picture rows and after that for all columns as shown in *Figure 2*.

The principle highlight of DWT is the multi-scale representation of capacity. By utilizing the wavelets, the given capacity can be analyzed at different levels of determination [20]. The DWT is likewise invertible and can be orthogonal. Wavelets appear to be compelling for analysis of textures recorded with different resolution. This is an essential issue in NMR imaging, on the grounds that high-resolution pictures take longer times to process.

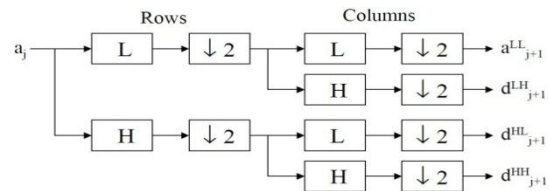


Figure 1 Wavelet decomposition for two-dimensional pictures (reproduced from[17])

B. DCT

The DCT was presented firstly by [21] in 1974, as the consequence of an endeavor to discover a useful close estimation of the “Karhunen Loève Transform” (KLT), which was viewed as ideal with respect to the conservativeness of the coefficients. These days, its application incorporates such a mixed bag of subjects in advanced digital signal processing, for example, equation resolution, partial differentials, feature extraction, data compression, etc. [22].

Owing to its recurrence distribution qualities, which have a tendency to focus the sign energy nearest to the lower recurrence coefficients, the (DCT) is utilized as center component in video standards and digital image compression, for example, H.264/MPEG-4 and JPEG [23]. Based on Salomon [24] the DCT can be determinant for the real valued input by the following Equation 3:

$$G_n = u_n \sum_{k=0}^{N-1} X_k \cdot \text{COS} \pi \frac{(2k+1)n}{2N}, 0 \leq n \leq N-1 \quad (3)$$

C.SVD

SVD is an effective strategy in numerous matrix

calculations and investigations [25]. Utilizing the SVD of an array in processes, as opposed to the first matrix, has the preference of being more solid to numerical error [26]. Moreover, SVD uncovers the geometric structure of an array, a paramount part of numerous matrix computation. A lattice can be depicted as a change starting with one vector space then onto the next. The segments of SVD measure the subsequent change between the basic geometry of those vector spaces. SVD is utilized in a mixed bag of utilizations, from minimum squares issues to a fathoming matrix of linear-equations. Each of these applications endeavor key properties of the SVD—its connection to the rank of an array and its capacity to convergent matrix of a given rank [27]. Numerous essential parts of straight variable based mathematics depend on deciding the rank of a matrix, making the SVD a vital and broadly utilized system. Based on [28], the SVD of an $m \times n$ matrix A is a factorization of

$$A = U \sum V^* \quad (4)$$

where U is an $m \times m$ unitary matrix, \sum an $m \times n$ rectangular slanting matrix, and V^* an $n \times n$ unitary matrix. V^* is the conjugate transpose of V , i.e., the $(i; j)$ th entrance of V is the complex conjugate of the $(j; i)$ th section of V^* . Note that a square matrix $A \in \mathbb{C}^{(m \times m)}$ is unitary if $A^* = A^{-1}$, and the solitary estimations of the square bases of the eigenvalues of $A^* A$, all of which are genuine and non-negative.

Since V is unitary, $V^* = V^{-1}$, and in this manner, we can revise the SVD as $AV = U\sum$, which can be separated into individual segments as $Av_i = \sigma_i u_i$. Utilizing this arrangement, it is simple to think about the SVD of a matrix A by separating its activity on the unit-sphere. Thus, let A be some genuine matrix $A \in \mathbb{R}^{m \times n}$, where $m \geq n$. At that point, A maps the unit-sphere $S \in \mathbb{R}^n$ to a hyper ellipse $AS \in \mathbb{R}^m$. The unit-sphere is “extended” by a few variables $\sigma_1, \dots, \sigma_m$ in some orthogonal bearings $u_1, \dots, u_m \in \mathbb{R}^m$. The vectors $\{\sigma_i u_i\}$ are known as the principal semi-axes of the hyper ellipse, with lengths $\sigma_1, \dots, \sigma_m$. In the event that A has rank r , then precisely r of the lengths σ_i will be non-zero, and if $m \geq n$, at most n will be non-zero.

D. SIFT

In 1999, the SIFT algorithm was presented by David Lowe to detect and describe images features. The algorithm was very effective in several applications in computer vision, such as object recognition, image stitching, face recognition, video tracking, and match

moving. The SIFT algorithm has mainly four phases, which will be described in the following:

1. Scale-space extrema detection

This phase of the filtering recognizes scales and locations that are identifiable from various perspectives to the same object. This can be accomplished by using a scale-space function. Furthermore, it has been demonstrated under sensible presumptions that it must depend on a Gaussian function. The scale-space is characterized by the following Equation 5:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (5)$$

where $*$ is the detour factor, $G(x, y, \sigma)$ is a scalar variable (Gaussian), and $I(x, y)$ is the input picture. Different methods can then be utilized to identify stable keypoint locations in the scale space. Gaussian difference is one such approach that spots scale space extrema, $D(x, y, \sigma)$, by figuring the contrast between two pictures, namely one with scale k times the other. $D(x, y, \sigma)$ can be computed by the following Equation 6:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (6)$$

To identify the neighborhood maxima and minima of $D(x, y, \sigma)$, every point is contrasted with its eight neighbors in the same scale, and its nine neighbors down and up one scale. In the case that the value is the maximum or minimum for all these points, consequently this point is an extremum.

2. Keypoint localization

This phase tries to exclude more points from the set of keypoints by discovering the points that contain low contrast or are poorly centralized on an edge. This is accomplished by computing the Laplacian value for every keypoint detected in phase 1. The area of extremum, z , can be computed by the following Equation 7:

$$z = \frac{a^2 D^{-1}}{aX^2} \frac{aD}{aX} \quad (7)$$

In cases that the function values of z are less than (underneath) the limit value, then these points are eliminated. This eliminates extrema with low contrast. To dispose of extrema with destitute localization, it was noticed that in these instances there is a huge principle curvature crossing over the edge, however, there is also a tiny bend in the vertical direction in the variances of the Gaussian function. In cases where the variance is less than (underneath) the degree of the largest to smallest eigenvector from the 2×2 Hessian array at the position and scale of the keypoint, the keypoint is refused.

3. Orientation assignment

This phase specifies a harmonic orientation to the keypoints in light of nearby picture properties. The keypoint descriptor, portrayed underneath can then be adapted and prorated with the orientation, thus accomplishing rotation of invariance. The technique taken to discover an orientation is as in the following:

- use the scale of keypoints to select the Gaussian smoothed picture L from up,
- calculate gradient magnitude, m ,
- $m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$
- calculate orientation, θ ,
- $\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y)))$
- design the histogram of orientation from tendency orientations of sample points,
- determine the highest summit in the histogram and use this summit and any other domestic summit within 80% of the rising of this summit to initialize a keypoint with that orientation,
- a few points will be appointed double orientations, and
- equip a parabola to the three values of histogram that are nearest to every summit to insert the peak positions.

3. Keypoint descriptor

The domestic tendency data, utilized above, are additionally used to initialize keypoint descriptors. The tendency data are rotated into alignment with the orientation of the keypoint and then oscillated by a Gaussian with a change of 1.5* keypoint scale. The data are used to initialize a group of histograms through a window focused on the keypoint. Keypoint descriptors regularly use a group of 16 histograms, adjusted in a 4x4 matrix, where everyone has eight orientation bins—one for every one of the major compass directions and one for every one of the midpoints of these directions. This outcome in a feature-vector consists of 128 elements. These outcome vectors are acquainted as SIFT keys and are utilized in a nearest neighbor's issue to determine bearable objects in the picture. The sets of keys are accepted on a bearable model which is determined when three or more keys are accepted by the model parameters, which is obvious in the picture with high likelihood. Because of the substantial number of SIFT keys in a picture of an object, ordinarily a 500x500 pixel picture will produce in the region of 2000 features, with intrinsic levels of obtuseness that are bearable, while the picture is still recognized by

this system.

4. SURF

SURF is a robust local feature detector, and can be utilized as a part of computer vision assignments, like 3D reconstruction or object recognition [29]. It is partially excerpted by the SIFT descriptor, therefore, comparable to SIFT algorithm, since those characteristic perspective identification for surf algorithm is still in light of the scale space hypothesis [30]. Meanwhile, the SIFT algorithm adopts Distinction of Gaussians (DoGs) to extricate feature-points, while SURF utilizes a number close estimations as the determinants of Hessian blob locator, which can be figured rapidly within a vital picture [30]. Based on Guo, et al. [31], a pixel point with scale σ in picture will have its Hessian array indicated as:

$$H(X, \sigma) = \begin{bmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{yx}(X, \sigma) & L_{yy}(X, \sigma) \end{bmatrix} \quad (8)$$

where L is the convolution of the picture with the second subordinate of the Gaussian. So as to accelerate the calculation over SIFT algorithm, the box channel is utilized to roughly supplant the Gaussian channel. In addition, the SIFT calculation simplifies the count of determinants, which no longer computes the weight from claiming every area separately, hence the determinant can be acquired by the following Equation 9 [31]:

$$\det(H) = \frac{\partial^2 f}{\partial^2 x^2} \frac{\partial^2 f}{\partial^2 y^2} - \left(\frac{\partial^2 f}{\partial^2 x \partial y} \right)^2 \quad (9)$$

where $\partial f / \partial x$ is the convolution effect of the picture with the template. This is due to the use of complete image and the box-filter, where those measures of the filter will be changed in the scale space developed by the SURF algorithm, while the picture size is steady. Interestingly, the filtered picture is incessant to be filtered in the SIFT algorithm. SURF algorithm calculation scale space is separated into several requests, and every request includes various layers.

4. Results

The experiments conducted in this study were twofold. First, the performance of forgery detection using single feature extraction algorithms for dimensionality reduction and keypoints was observed. Then the integrations of the methods were implemented, and the results were compared.

Implementation of single feature extraction algorithms

This section discusses several feature extraction techniques adopted in the experiments. The experiment investigated each technique from both dimensionality reduction and key point categories.

A-Dimensionality reduction

Dimensionality reduction is a technique that reduces the dimensionality of analyzed data. In image processing, a dimension refers to the width, height, and detail of an image. Many methods for reducing the image dimensionality have been developed to achieve this task. This study investigated four methods for dimensionality reduction, namely, PCA, DCT, SVD, and DWT. The performance of each method in reducing image dimensionality was investigated, and the results of each method of the experiments were discussed to observe the effect of each method on the image dimensions, as well as their execution time. *Table 1* provides the resulting values (width and height) after applying each method on the image; the table also provides the execution

time of the process. As shown in *Table 1*, the execution of the SVD algorithm of the tested images did not affect the image dimensions because the dimensions remained 512x512 and 2,048x1,536 pixels. As for the details of the resulting images, they depend on low-rank approximation rank k. The value of rank k can be determined by values ranging from 1 to 298 based on the percentage of the desired detail. When the value of rank k increases, the image detail increases, and vice versa. For instance, if the value of rank k = 10, then the image will contain less detail; if the value of rank k = 50, then the image will contain more detail. Some examples of the tested images with different rank k values are presented in *Figure 3*. The rank k values of 10 and 50 clearly indicate the effect of rank k values on image detail. Images with low rank k values (*Figure 3A* and *3B*) are less detailed, whereas the images with high rank k values (*Figure 3C* and *3D*) are more detailed.

Table 1 Results of SVD, DCT, PCA, and DWT methods on two image samples

METHOD	BMP width pixel	BMP height pixel	BMP time sec	JPG width pixel	JPG height pixel	JPG time sec
SVD	512	512	0.460	2,048	1,536	7.893
DCT	512	512	0.752	2,048	1,536	2.439
PCA	512	512	17.589	2,048	1,536	259.166
DWT	256	256	0.285	1,024	768	0.985

Similar to the SVD algorithm, the DCT algorithm did not reduce the dimensions (width and height) of the test images, but instead retained them. As shown from the experiments, the DCT algorithm could reduce the amount of detail in the images and consequently determine the magnitude values of the coefficients that would comprise the DCT matrix. The coefficients with magnitude values less than those of the determined values were ignored by setting them to 0, after which the image was produced. Some examples of the tested images with different magnitude values are presented in *Figure 4*. Specifically, two magnitude values (i.e., 30 and 200) were tested. These figures provide a clear overview of how different magnitude values could affect image detail.

A low magnitude value results in a heavily detailed image (*Figures 4A* and *4B*), whereas a high magnitude value reduces image detail (*Figures 4C* and *4D*). The PCA algorithm does not differ from its predecessors, that is, it also did not affect image dimensions. The main effect of the PCA algorithm is evident on the tested image components. Studies that use the PCA algorithm often need one, two, or three

of these principal components to describe images in the analysis process. The number of principal components that will characterize an image in the analysis process is determined by the required compression. *Figure 5* shows how different compression ratios can affect image detail. Setting a low compression ratio increases image detail (*Figure 5A* and *5B*), whereas setting a high compression ratio decreases image detail (*Figure 5C* and *5D*).

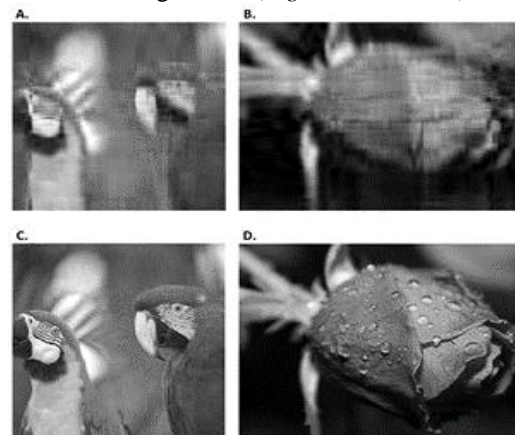


Figure 2 Different low-rank approximation values: rank k=10 in (A) and (B), rank k=50 in (C) and (D)

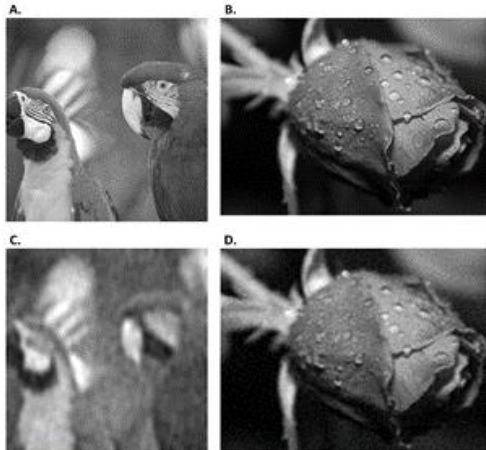


Figure 3 Different values of ignored coefficients: magnitude value of less than 30 in (A) and (B), magnitude value of less than 200 in (C) and (D)

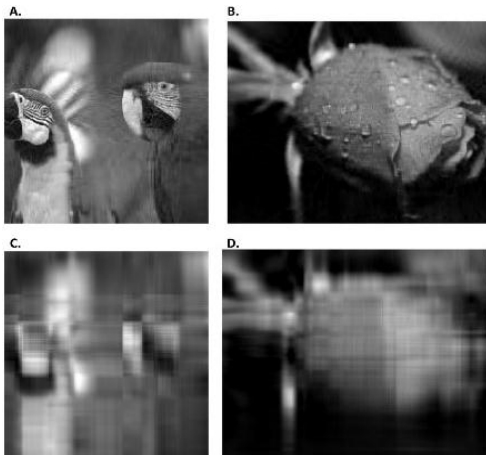


Figure 4 Various compression ratios: compression ratio=4.33898 in (A) and (B), compression ratio=23.2727 in (C) and (D)

The results of the DWT algorithm in *Table 1* show that there is a significant difference from those of the other algorithms. The DWT algorithm demonstrated a dual effect on the tested images. Specifically, the dimensions (width and length) of the tested images were reduced by half, that is, from 512×512 to 256×256 for the image in *Figure 6A* and from $2,048 \times 1,536$ to $1,024 \times 768$ for the image in *Figure 6B*. This reduction sped up the analysis process.

In addition, the DWT algorithm conservatively reduced the image detail with a sufficient amount of resolution, as shown in *Figure 6*. *Figure 7* shows the execution time of all the test methods. The PCA algorithm was the slowest among all the algorithms. Specifically, the PCA algorithm required an execution time of 17.589 seconds for the first image and an execution time of 259.166 seconds for the second image. The execution times of both the SVD and DCT algorithms were 0.460 seconds and 0.752 seconds, respectively. For the JPG image, the execution times of the SVD and DCT algorithms were 7.893 seconds and 2.439 seconds, respectively. As shown in *Figure 7*, the DWT algorithm was the fastest algorithm. Its execution times for the first and second images were 0.285 seconds and 0.985 seconds, respectively.

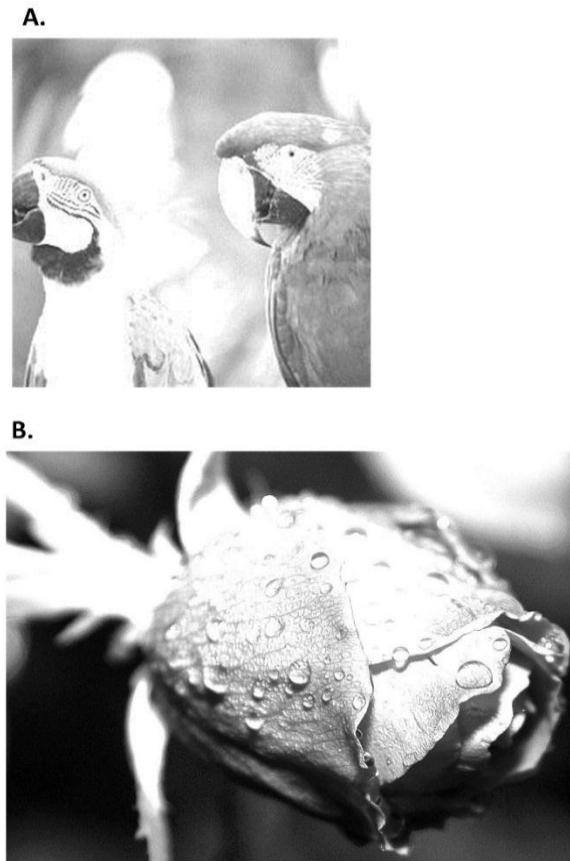


Figure 6 DWT performance on the test images: (A) BMP and (B) JPG images



Figure 7 Execution time of the methods under study

B. Keypoint methods

Keypoint is a technique for reducing the size of the data analyzed by selecting a sample of points from the data to be adopted in the analysis. This study investigated two keypoint methods to verify the performance of each one of the tested images. This study also adopted the scale-invariant feature transform (SIFT) and speeded up robust features (SURF) algorithms to detect these keypoint features on the experimental images. The detection time for each algorithm was also calculated.

Figure 8 illustrates the number of keypoints detected in the experimental images with each algorithm. *Figure 8* shows that the SIFT algorithm detected more keypoints on both experimental images compared to the SURF algorithm. Specifically, the SIFT algorithm detected 722 keypoints on the BMP image and 5,822 keypoints on the JPG image, whereas the SURF algorithm detected 217 keypoints on the BMP image and 1,701 keypoints on the JPG image. The SIFT algorithm detected numerous

features in the background of the experimental images, which did not contain sufficient information for the matching process. Meanwhile the SURF algorithm detected points in the foreground of the experimental images and ignored the blurred zones in the background. Although the SURF algorithm identified less features than the SIFT algorithm, the detection of the former was more accurate, as shown in *Figure 9*.

Figure 10 also shows the required time for revealing the keypoints. As shown in the experimental results, the SURF algorithm was faster than the SIFT algorithm; in addition, the SIFT algorithm was slow, particularly when executed on large images. As shown in the *Figure 10*, the execution times of the SIFT algorithm were 0.525 seconds and 3.803 seconds for the BMP and JPG images, respectively, whereas those of the SURF algorithm were 0.507 seconds and 2.334 seconds, respectively.

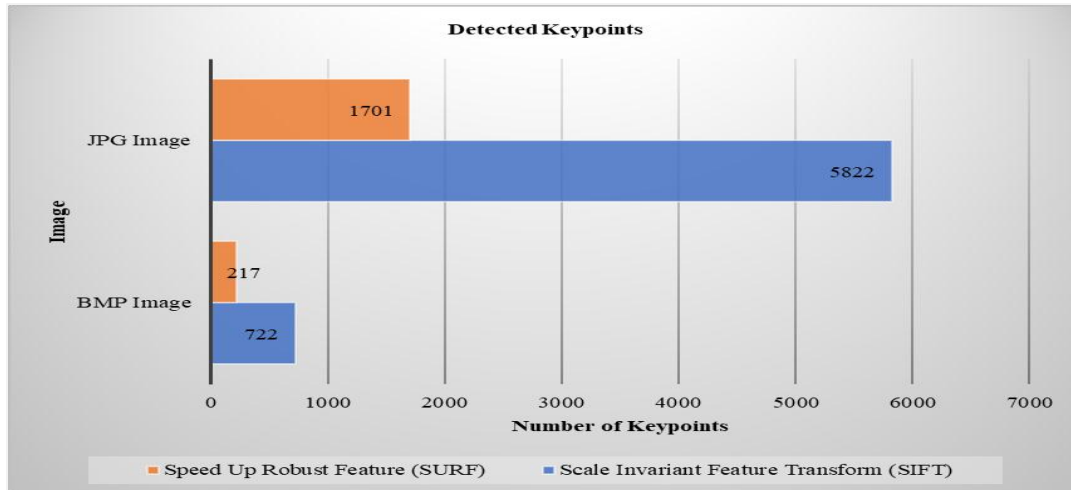


Figure 8 Number of detected keypoints

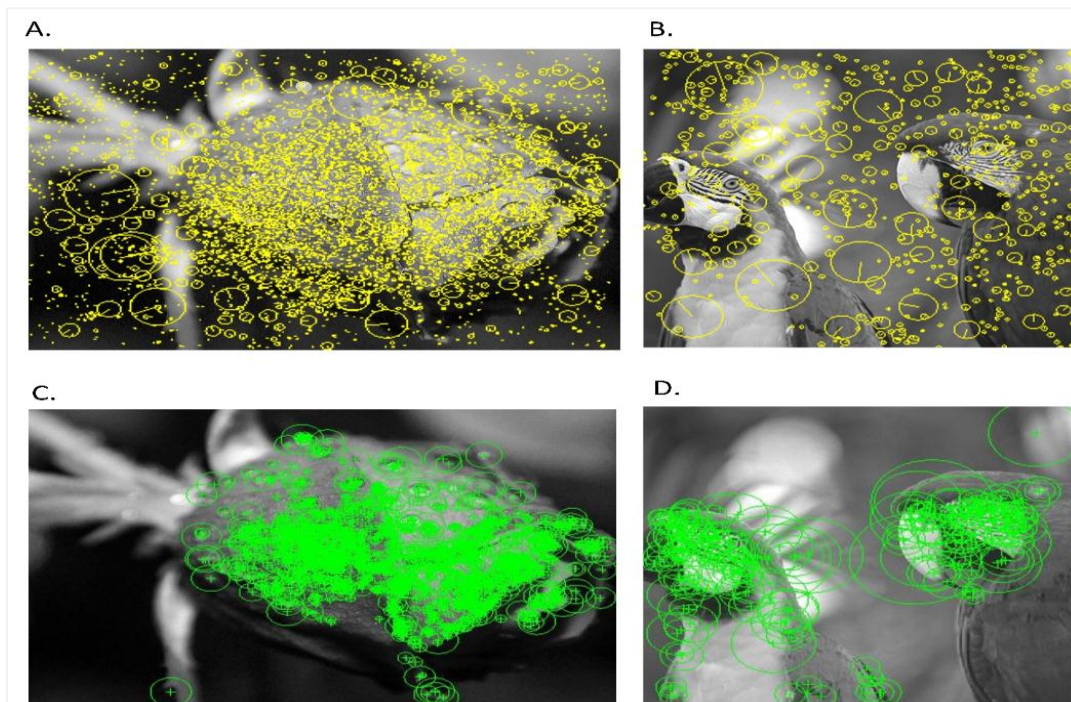


Figure 9 Execution of the SIFT algorithm on the experimental (A) JPG and (B) BMP images and the execution of the SURF algorithm on the experimental (C) JPG and (D) BMP images

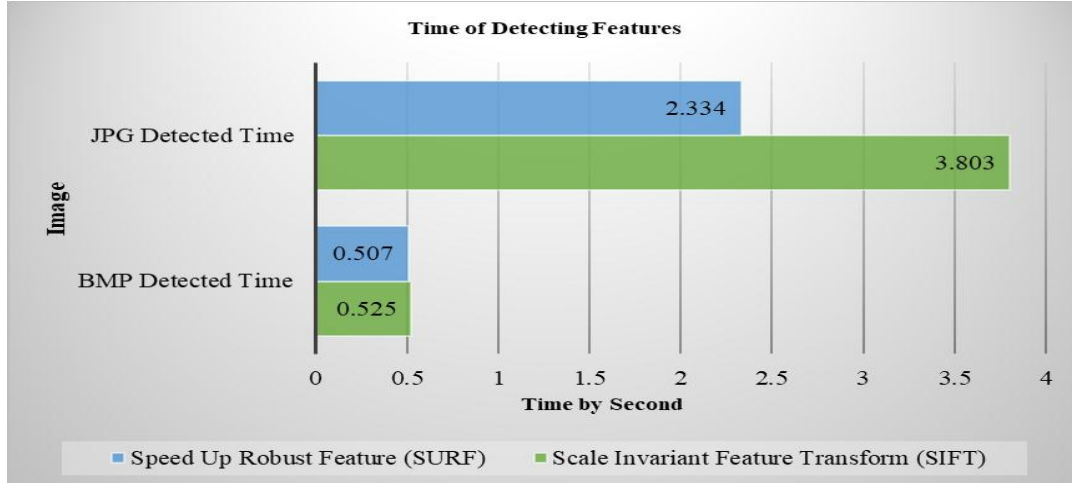


Figure 10 Duration of feature detection in the experimental images

5. Integration of feature extraction methods

Several detection methods were constructed in this section with different integration of feature extraction methods, based on the findings and discussion in the previous sections. The experiments were executed using MATLAB 2014b on a computer equipped with an Intel Core i5 2.20 GHz processor and 8 GB DDR3RAM. The discussion in this section is based on four sample images, as depicted in Figure 11. Images 1 and 2 were in BMP format with 512x512 pixels. Images 3 and 4 were in JPG format with 2,048x1,536 pixels. A total of 50 forged images from both datasets were used in the experiment. The study

used different images to clarify the different copy-move forgery attack. One image explained one type of a copy-move forgery attack while the others identified the other copy-move forgery attacks. This study selected four methods for executing the experiments based on the result of the previous section. The DCT, SVD, and DWT algorithms were chosen from the dimensionality reduction category. Furthermore, the study selected the keypoint category of the SURF algorithm. The study integrated the SVD with SURF, DCT with SURF, and DWT with SURF. Table 2 contains the experimental results of the integrated methods.

Table 2 Experimental results of the integrated methods for the images in Figure 11

Algorithm	Detected feature points				Matching feature points			
	Image 1	Image 2	Image 3	Image 4	Image 1	Image 2	Image 3	Image 4
SVD + SURF	186	121	4,304	2,075	0	3	38	35
DCT + SURF	177	92	6,241	1,241	0	0	53	14
DWT + SURF	175	99	57	317	1	3	4	10



Figure 5 Four images used in the experiments: (A) Image 1 and (B) Image 2, are BMP images with 512×512 pixels, and (C) Image 3 and (D) Image 4, are JPG images with $1,536 \times 2,048$ pixels

The results showed the number of detected features and the number of matching feature points in the experimental images. As shown in *Table 2*, the SVD and SURF integrated method, and the DCT and SURF integrated method detected 186 and 177 feature points on image 1, respectively. However, these two integrated methods did not produce matching feature points (i.e. matching feature points are marked by the yellow lines), as illustrated in *Figure 12 (A and B)*. For image 1, the DWT and SURF integrated method detected 175 feature points and 1 matching feature point, as illustrated in *Figure 12 (C)*.

For image 2, the DCT and SURF integrated method detected 121 feature points and 0 matching points, as shown in *Figure 13 (A)*. On the one hand, the SVD and SURF integrated method detected 92 feature points and 3 matching points, but it did not specify all forged areas. Moreover, the SVD and SURF integrated method identified non-forged areas, as

shown in *Figure 13 (B)*. On the other hand, the DWT and SURF integrated method detected 99 feature points and 3 matching points; thus, it correctly detected all the forged regions on the image, as shown in *Figure 13 (C)*. However, these integrated methods did not identify the forged regions on the images, as shown in *Figure 14 (A and B)*. Meanwhile the DWT and SURF integrated method accurately detected the forged regions on image 3; specifically, it detected 57 keypoints and 4 matching keypoints, as shown in *Figure 14 (C)*.

The SVD and DCT algorithms integrated with the SURF algorithm were unable to detect the forged regions on image 4, but they managed to identify 35 and 14 random matching keypoints, respectively, as shown in *Figure 15 (A. and B.)*. In contrast, the DWT and SURF integrated method detected the forged regions on image 4 and specified all the matching keypoints, as shown in *Figure 15 (C)*.

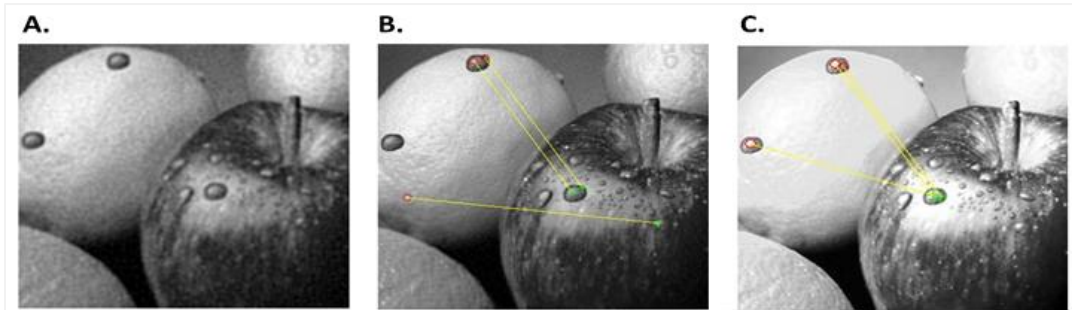


Figure 12 Results of method detection on image 1: (A) SVD and SURF, (B) DCT and SURF, and (C) DWT and SURF

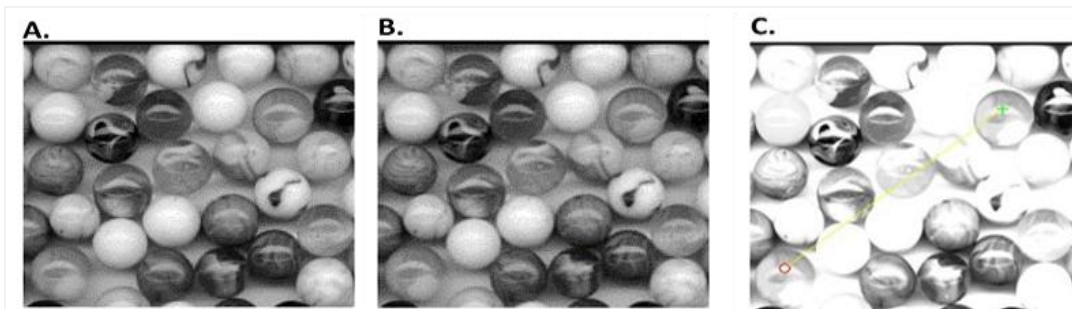


Figure 6 Results of method detection on image 2: (A) DCT and SURF, (B) SVD and SURF, and (C) DWT and SURF

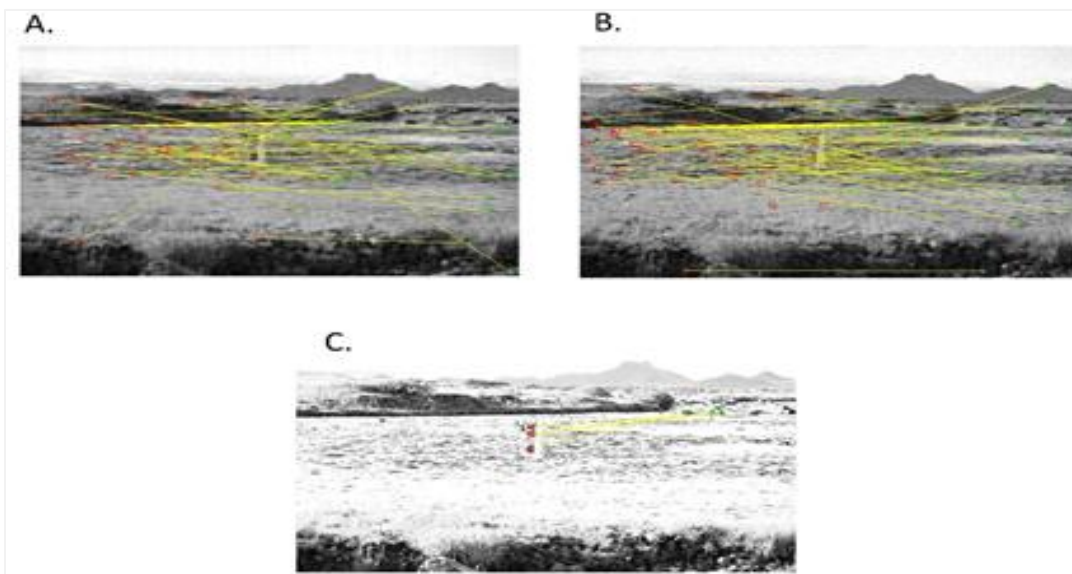


Figure 7 Results of method detection on image 4: (A) SVD and SURF, (B) DCT and SURF, and (C) DWT and SURF

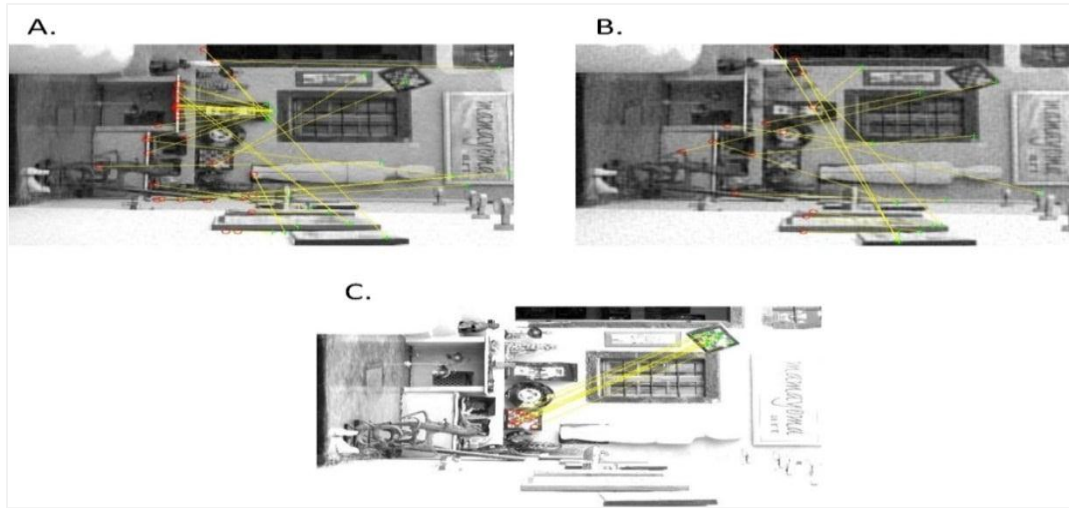


Figure 15 Results of method detection on image 3: (A) SVD and SURF, (B) DCT and SURF, and (C) DWT and SURF

6. Conclusion and future work

In this article, two groups of image feature extraction methods were discussed. These groups are classified into dimensional reduction and keypoints. Theoretically, this study greatly contributes to the literature through its proposed method for detecting copied–moved images. This study has performed multiple experiments on all methods involved under each group and discussed the results of each one. The DWT algorithm was found to be the best algorithm for reducing the image dimensionality (as compared with SVD, DCT, and PCA), and the SURF algorithm was found to be an accurate and quick method for extracting features from an image, as compared with the SIFT algorithm.

Constructing the method is done by integrating two feature extraction algorithms resembling the DWT and SURF algorithms. Particularly, the experiments were conducted on the integration of SVD and SURF, DCT and SURF, and DWT and SURF detect one type of image forgery (copy-move). The result of each integrated method was discussed. Based on the test results, it seems that the integration of the DWT with SURF is more accurate than other integrated methods to detect the fraudulent areas.

The proposed method is able to identify accurately forged points when used to detect normal copy–move forgery. The method also solves the rotation issue in copy–move forgery because it can detect forged images with copies regions rotated at various angles. The proposed method was also found to be an accurate method for detecting the multi-duplicated

regions of fake images (discussed in [32]). From a practical viewpoint, the developed method could contribute to the enhancement of image forgery detection. In the future, the methods discussed in this study will be tested for other types of image forgery, such as image splicing.

Acknowledgment

None.

Conflicts of interest

The authors have no conflicts of interest to declare.

References

- [1] Kumar S, Desai J, Mukherjee S. A fast DCT based method for copy move forgery detection. In second international conference on image information processing 2013 (pp. 649-54). IEEE.
- [2] Lynch G, Shih FY, Liao HY. An efficient expanding block algorithm for image copy-move forgery detection. *Information Sciences*. 2013; 239:253-65.
- [3] Tralic D, Rosin PL, Sun X, Grgic S. Copy-move forgery detection using cellular automata. In *cellular automata in image processing and geometry 2014* (pp. 105-25). Springer, Cham.
- [4] Sunil K, Jagan D, Shaktidev M. DCT-PCA based method for copy-move forgery detection. In *ICT and critical infrastructure: proceedings of the annual convention of computer society of india-Vol II 2014* (pp. 577-83). Springer, Cham.
- [5] Ahmadi M, Ulyanov D, Semenov S, Trofimov M, Giacinto G. Novel feature extraction, selection and fusion for effective malware family classification. In *the conference on data and application security and privacy 2016* (pp. 183-94). ACM.
- [6] Ramya RS, Venugopal KR, Iyengar SS, Patnaik LM. Feature extraction and duplicate detection for text

- mining: a survey. *Global Journal of Computer Science and Technology*. 2017.
- [7] Lei B, Xu G, Feng M, Van der Heijden F, Zou Y, De Ridder D, et al. Classification, parameter estimation and state estimation: an engineering approach using MATLAB. John Wiley & Sons; 2017.
- [8] Shah FP, Patel V. A review on feature selection and feature extraction for text classification. In *international conference on wireless communications, signal processing and networking 2016* (pp. 2264-8). IEEE.
- [9] Ponti M, Nazaré TS, Thumé GS. Image quantization as a dimensionality reduction procedure in color and texture feature extraction. *Neurocomputing*. 2016; 173:385-96.
- [10] Bin YA, Xingming SU, Xianyi C, Zhang J, Xu LI. An efficient forensic method for copy-move forgery detection based on DWT-FWHT. *Radioengineering*. 2013; 22(4): 1098-105.
- [11] Zhao J, Guo J. Passive forensics for copy-move image forgery using a method based on DCT and SVD. *Forensic Science International*. 2013; 233(1-3):158-66.
- [12] Huang H, Guo W, Zhang Y. Detection of copy-move forgery in digital images using SIFT algorithm. In *pacific-Asia workshop on computational intelligence and industrial application 2008* (pp. 272-6). IEEE.
- [13] Pan X, Lyu S. Detecting image region duplication using SIFT features. In *international conference on acoustics, speech and signal processing 2010* (pp. 1706-9). IEEE.
- [14] Pan X, Lyu S. Region duplication detection using image feature matching. *Transactions on Information Forensics and Security*. 2010; 5(4):857-67.
- [15] Bo X, Junwen W, Guangjie L, Yuewei D. Image copy-move forgery detection based on SURF. In *international conference on multimedia information networking and security 2010* (pp. 889-92). IEEE.
- [16] Shukla KK, Tiwari AK. Efficient algorithms for discrete wavelet transform: with applications to denoising and fuzzy inference systems. Springer Science & Business Media; 2013.
- [17] Kociotek M, Materka A, Strzelecki M, Szczypiński P. Discrete wavelet transform-derived features for digital image texture analysis. In *international conference on signals and electronic systems 2001* (pp. 99-104).
- [18] Chui CK. *An introduction to wavelets*. Elsevier; 2016.
- [19] Kanika DN, Sharma K. Comparative analysis of discrete wavelet transform and fast wavelet transform on image compression. *International Journal of Engineering Research & Technology*. 2012; 1(5):1-7.
- [20] Chowdhury MM, Khatun A. Image compression using discrete wavelet transform. *International Journal of Computer Science Issues*. 2012; 9(4):327-30.
- [21] Ahmed N, Natarajan T, Rao KR. Discrete cosine transform. *IEEE Transactions on Computers*. 1974; 100(1):90-3.
- [22] Cook GW, Kalker T. The sparse discrete cosine transform with application to image compression. In *picture coding symposium (PCS) 2013* (pp. 9-12). IEEE.
- [23] Kekre HB, Sarode T, Natu PJ. Color image compression using hybrid Haar-DCT wavelet in different color spaces. *Advances in Image and Video Processing*. 2014; 2(4):1-1.
- [24] Salomon D. *A concise introduction to data compression*. Springer Science & Business Media; 2007.
- [25] Andrews H, Patterson CL. Singular value decomposition (SVD) image coding. *IEEE Transactions on Communications*. 1976; 24(4):425-32.
- [26] Gallier J. Singular value decomposition (SVD) and polar form. In *geometric methods and applications 2011* (pp. 367-85). Springer, New York, NY.
- [27] Lee N, Cichocki A. Big data matrix singular value decomposition based on low-rank tensor train decomposition. In *international symposium on neural networks 2014* (pp. 121-30). Springer, Cham.
- [28] Ientilucci EJ. *Using the singular value decomposition*. Rochester Institute of Technology, Rochester, New York, United States, Technical Report. 2003.
- [29] Balaji GN, Subashini TS, Chidambaram N. Cardiac view classification using speed Up robust. *Indian Journal of Science and Technology*. 2015; 8(S7):1-5.
- [30] Bay H, Ess A, Tuytelaars T, Van Gool L. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*. 2008; 110(3):346-59.
- [31] Guo F, Luo X, Liu Y. Research on feature extraction and match method based on the surf algorithm for mobile augmented reality system. In *international industrial informatics and computer engineering conference 2015*. Atlantis Press.
- [32] Alamro L, Yusoff N. Copy-move forgery detection using integrated DWT and SURF. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*. 2017; 9(1-2):67-71.



Nooraini Yusoff is currently an Associate Professor from the Data Science Lab Research group at Universiti Utara Malaysia, Malaysia. She obtained her Ph.D from University of Surrey, United Kingdom, in the field of Computational Neuroscience. She has 15+ years of experience in Artificial Intelligence research. Nooraini is actively involved in Soft Computing and Neural Network Researches, particularly in developing Learning Algorithms for Complex Systems, Brain-Inspired Applications, and Cognitive Modelling. She has published a number of national and international journals and conference proceedings. She is also the reviewer of some reputable journals in the related fields.
Email: nooraini@uum.edu.my



Loai Alamro received the Assistant Lecturer degree in 2016, from Al-Turath University College, Iraq and the MSc. degree in Information Technology, in 2016, from the Universiti Utara Malaysia, Malaysia. He was the director of the Internet Center at Al-Turath University College

2008-2013. In addition, he was the secretary of the Al-Turath Scientific Journal 2010-2013 and the secretary of the Committee of Scientific Promotion at the same university 2011-2013. His research interests involve Computer Vision, Image Processing, Multimedia Forensics, Matching Techniques, Pattern Recognition, Object Detection, and Machine Learning.

Email: loaialamro@gmail.com