



Robustness Improvement Against a Non-Geometrical Attacks of Lifting Scheme-Based Image Watermarking through Singular Value and Schur Decompositions

Aris Marjuni^{1*} Oky Dwi Nurhayati²

¹*Informatics Engineering, Faculty of Computer Science, Universitas Dian Nuswantoro, Indonesia*

²*Computer Engineering, Faculty of Engineering, Universitas Diponegoro, Indonesia*

* Corresponding author's Email: aris.marjuni@dsn.dinus.ac.id

Abstract: Image watermarking is a technique to embed a piece of information or data, which also can be an image, into the host or medium image for specific purposes, such as authentication, security, or copyright protection. A good image watermarking method should have a good perceptibility after adding the watermark. It should also robust against attacks and the watermark should be able to be recovered although the watermarked image has distorted. This paper presents the proposed lifting image watermarking scheme based on wavelet transform through singular value and Schur decomposition. This method is proposed to improve the perceptibility and the robustness of the image watermarking scheme, especially against non-geometrical attacks. The lifting scheme is performed by lifting wavelet transform to produce a better imperceptibility. The use of singular value decomposition is used to increase the intactness of the embedded watermark from the distorted image, which is performed on the triangular matrix of low-frequency sub-band produced by Schur decomposition. In perceptibility performance, experimental results show that the proposed method could produce a watermarked image very close to the host image with more than 99% of similarity level. In robustness performance, the proposed method can recover the embedded watermark even though the watermarked image has been distorted by non-geometrical attacks with more than 90% of normalized correlation average.

Keywords: Image watermarking, Lifting scheme, Singular value decomposition, Schur decomposition, Non-geometrical attacks.

1. Introduction and related works

Digital image watermarking is a useful technique to embed a unique identifier or watermark into the medium or host image such that it can be used to prove or authenticate the rightful ownership [1]. Technically, there are two image watermarking types, that are visible and invisible [1, 2]. In visible watermarking, the embedded watermark is perceptible visually by a human eye. Conversely, in invisible watermarking, the embedded watermark is invisible visually.

The most important aspects of the image watermarking quality are determined by imperceptibility and robustness parameters [3-4]. Imperceptibility, as the first aspect, refers to the visual quality of the host image upon embedded the watermark. A watermarked image should have a

similar quality close to the original image. The second aspect is robustness, which refers to the ability of the watermarking method to extract the embedded watermark from the watermarked image. This watermark extraction will be difficult when the watermarked image has been distorted or change by accidentally or intentionally image processing, which is called attacks, such as filtering, noising, compression, scaling, rotation, cropping, and more [5-7]. These attacks can be categorized into two types, non-geometrical and geometrical attacks. A non-geometrical attack might change the image pixel intensity but does not change the image pixel geometry, while a geometrical attack will change the image pixel geometry and its structure. Image compression, filtering, histogram equalization, blurring, sharpening, noising, and dithering are an example of non-geometrical attacks. While image

scaling, cropping, rotation, and translation are an example of geometrical attacks [8].

In the image watermarking development, there is a big challenge to cover the imperceptibility and robustness trade-off. The image watermarking should produce a watermarked image with good imperceptibility, and it should also be robust against any attacks so that the watermark could be extracted from the watermarked image [7, 9].

The image watermarking scheme can be developed on the spatial and transform or frequency domains [10]. In the spatial domain, a watermark is embedded by modifying the pixel values of the host image such that they remained unchanged. The least significant bit (LSB) is the most popular technique used in the spatial domain [11]. Many image watermarking methods are proposed to improve its performance in the spatial domain, such as using LSB with encryption and cryptography [12], using LSB with a random mapping function [13], modified LSB using SVD [14], combining LSB and an inverse bit [15], and many more improvements related to the use of LSB method. In the transform domain, a watermark is embedded through the transformed coefficients. The most popular techniques in this domain include discrete Fourier transform (DFT) [16, 17], discrete wavelet transform (DWT) [18-20], discrete cosine transform (DCT) [8, 21, 22], and singular value decomposition (SVD) [23-25].

Each method has advantages and disadvantages in the image watermarking scheme development. Some proposed image watermarking methods were carried out by combining one transform with the other transforms within or across domains. The image watermarking scheme based on DCT and DWT was proposed by Akter et al. [26], where the watermarking process is performed by two-level, three-level, and four-level DWT followed by respective DCT on the host image. Similar image watermarking schemes based on DCT and DWT are also proposed by Deb et al. [27] and Jiansheng et al. [28]. Razafindradina et al. [29] proposed the image watermarking using the hybrid of DCT and Schur decomposition by adding the triangular matrix of the watermark after the Schur decomposition to the DCT transform of the host image. Sverdllov et al. [30] proposed the image watermarking scheme based on the DCT and SVD. In their method, after applying the DCT to the cover image and ordering the DCT coefficients into four quadrants, they then apply the SVD to each quadrant. The other hybrid image watermarking schemes are also proposed using three transforms that are DCT, DWT, and SVD. Khan et al. [31] and also Satyanarayana and Kumar [32] chose the DWT high band to embed the watermark and also

apply the zig-zag method to map the DCT coefficient before applying the SVD on each quadrant. The image watermarking schemes based on wavelet transform and SVD are proposed by Alshoura et al. [1], Kala [33], and Talbi and Bouhlef [34], where the watermark is embedded on the singular values of the low-low sub-band of the original image. The different approach of the image watermarking scheme is also proposed by Liu et al. [35], where the watermark is embedded using SVD on the multi-level sub-bands by DWT, which their coefficients are changed by the HD. They also used an objective evaluation function (OEF) and fruit fly optimization algorithm (FOA) to obtain the optimal scaling factor and optimal factor of the contradiction or trade-off between imperceptibility and robustness. While the other image watermarking schemes based on the other domains are proposed using lifting wavelet transform (LWT) [2, 10, 36, 37], and Schur decomposition [38-41].

The image watermarking based on DWT is the most popular method as it has some advantages, such as multi-resolution, good energy compression, and imperceptible visual quality [35]. While the lifting wavelet transform is one of the best DWT implementations in the image watermarking method that gives better results in visual image quality [34, 36]. This means that the LWT could produce a good imperceptibility in image watermarking. It can also be used to reduce information loss of an embedded watermark from a distorted image. However, the trade-off between imperceptibility and robustness is always a challenge to propose a better image watermarking method.

In this paper, the lifting-based watermarking through singular value and Schur decomposition is proposed to achieve a better performance, especially robustness against non-geometrically attacks. The lifting scheme is selected as the watermarking scheme basis to reduce the wavelet computation by performing the decimation into wavelet filter factorization, which are lower and upper triangular matrices [42]. The use of SVD and Schur decomposition in this proposed method is motivated that SVD has quantitative information to represent the image through singular values and vectors. Thus, the modification of singular values will correspond to the singular vectors. This characteristic will be an important point to apply the image watermarking scheme through SVD [35]. This SVD is mainly used for embedding the watermark on its singular values. Whilst the Schur decomposition in this proposed method is especially used to increase the watermark embedding and extraction processes because this decomposition is an intermediary step for SVD,

which is performed just in the upper triangular matrix [43, 44]. Overall, the details of this proposed method are described in Section 3.

This paper is organized as follows. Section 2 presents the theoretical backgrounds that are mainly needed to describe the proposed image watermarking method, including singular value decomposition, Schur decomposition, and lifting scheme based on wavelet domain. Section 3 presents the details of the proposed method, including watermark embedding and extraction processes. Section 4 presents the experimental setup, including preparation, preprocessing, experimental design, and performance measurement. Section 5 presents the experimental results and discussion to explain the proposed method performance. Finally, Section 6 presents a conclusion of the proposed method.

2. Theoretical backgrounds

2.1 Singular value decomposition

Singular value decomposition (SVD) is a popular method to decompose a symmetric matrix into several component matrices, which are the singular value and singular vectors matrices [3, 6]. The singular value matrix consists of nonnegative real numbers or singular values listed in decreasing order on its diagonal. The singular vectors matrix consists of the singular vectors that correspond to the singular values. There are two singular vectors matrices, which are left and right singular vectors. Let A is a symmetric matrix with $N \times N$ of size, the $f(A)$ as SVD function of A is defined in Eq. (1) as follows.

$$f(A) = [U, S, V] \quad (1)$$

where S is a diagonal matrix of singular values s_1, s_2, \dots, s_N , U is an orthogonal matrix of left singular vectors, and V is an orthogonal matrix of right singular vectors.

In SVD, the singular values of singular matrix S are unique, while the singular matrices U and V are not unique. However, the singular vectors matrices have a unique characteristic, where $UU^T = I_N$ and $VV^T = I_N$ with I_N is an identity matrix of $N \times N$ in size. SVD also provides a reconstruction method to get the matrix A after decomposed it into U , S , and V matrices, which is defined in Eq. (2) as follows:

$$A = U \times S \times V^T \quad (2)$$

where S is a diagonal matrix of singular values s_1, s_2, \dots, s_N , U is an orthogonal matrix of left singular

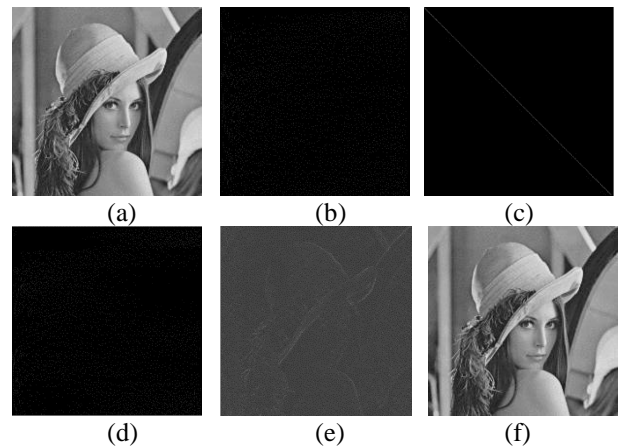


Figure. 1 Image and its SVD: (a) original Lena image, (b) U component, (c) S component, (d) V component, (e) UV^T component, and (f) reconstructed Lena image

vectors, and V^T is a transpose matrix of V .

In image processing, SVD is stable and effective to separate the image into linearly independent components, which are singular values and singular vectors. Singular values matrix S represents the luminance or energy as the main information of an image, while singular vectors U and V represent the image geometrical properties. An example of both image components is illustrated in Fig. 1 as follows.

The component matrices might not readable in the human visual system as the S component just represents the estimate of the image signal layer and both U and V components represent the noise layers. However, all components have numerical information which is very important in image computation. After decomposition, the original matrix A can be reconstructed by multiplication between U , S , and V^T , as computed in Eq. (2). SVD has been applied successfully in many image processing analyses, such as noise filtering, compression, forensic, and watermarking [24].

In image computation, SVD has the following advantages: Eq. (1) the singular value is relatively stable and there is no significant change due to small disturbances in the entire image, Eq. (2) singular values comes with an intrinsic image layer rather than visual, which makes it easier to perform any various algebraic operations in image analysis. However, SVD also has disadvantages, such as slow and computationally expensive, and also need more attention in handling the case of missing data.

2.2 Schur decomposition

Schur decomposition is a decomposition method of a square matrix into unitary and upper triangular matrices [38, 40-41]. Let I be a square matrix $N \times N$

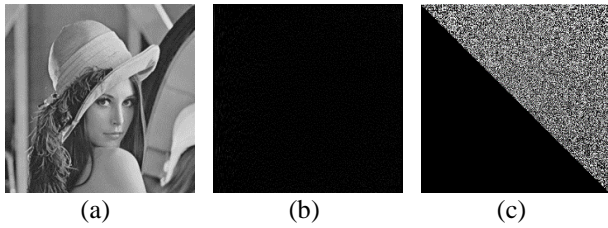


Figure. 2 Image and its Schur components: (a) original Lena image, (b) P component, (c) T component

of size. The $f(I)$ as Schur decomposition function of I is defined in Eq. (3), as follows:

$$f(I) = [P, T] \tag{3}$$

where P is a unitary matrix and T is an upper triangular matrix. The unitary matrix P represents the Schur vectors and satisfies the $UP = I_N$. The upper triangular matrix T is a diagonal matrix with eigenvalues on its diagonal. An example of this decomposition can be illustrated in Fig. 2 using the original Lena image. It shows that this decomposition produces the Schur vector form and upper-triangular form of eigenvalues as shown in Fig. 2(b) and Fig. 2(c), respectively.

In Schur decomposition, the original matrix can be reconstructed by multiplication between P , T , and P^T , as formulated in Eq. (4), as follows.

$$I = P \times T \times P^T \tag{4}$$

The Schur decomposition is similar to SVD, where both decompositions produce eigenvalues and eigenvectors components. Schur decomposition is also numerically stable like SVD in decomposing the matrix into sub-matrices components. In the case of a square matrix, the Schur decomposition has an advantage in computation speed as it uses an upper triangular matrix. The Schur decomposition produces just two sub-matrices components instead SVD produces three sub-matrices components. In term of complexity, Schur decomposition requires about $\frac{8}{3}N^3$ flops, instead SVD requires about $11N^3$, where N is iterators [41]. Hence, Schur decomposition has computationally efficient [43].

2.3 Lifting scheme

The lifting scheme, usually called second-generation wavelet transform and also lifting wavelet transform (LWT), was proposed initially by Wim Sweldens in 1995. This is a decomposition method to split a DWT into a finite sequence of filtering steps,

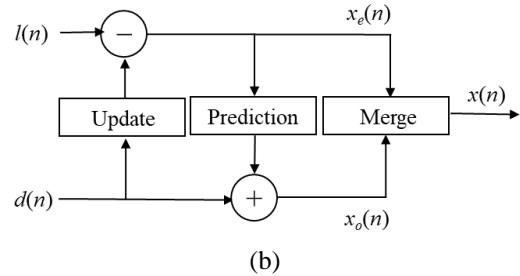
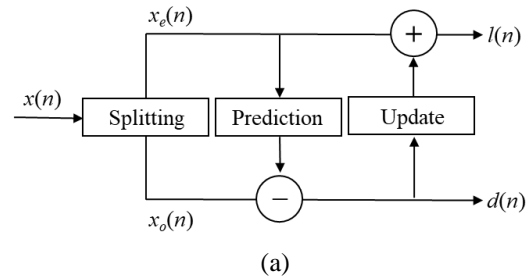


Figure. 3 Lifting scheme of signal $x(n)$: (a) decomposition, (b) reconstruction

which are called lifting steps [2, 36-37]. Comparing to the traditional DWT, the lifting scheme is faster and more effective to perform the wavelet with less data loss. The listing scheme also allows constructing an integer to integer wavelet transform that makes it more practical rather than a floating-point problem in the wavelet. These advantages make the Schur scheme became powerful in image analysis.

Practically, the lifting scheme corresponds to the polyphase matrix factorization of the wavelet filters into elementary matrices. Let given an original signal $x(n)$ The factorization $x(n)$ is performed in the three steps that are splitting, prediction, and update, as illustrated in Fig. 3a.

1. Splitting

The signal $x(n)$ is split into non-overlapping even signal $x_e(n)$ and odd signal $x_o(n)$, which are defined in Eq. (5).

$$x_e(n) = S_{2n} \text{ and } x_o(n) = S_{2n+1} \tag{5}$$

2. Prediction

If both even and odd signals are correlated then one signal can be the predictor of the other signal. Let $P(\cdot)$ is a predictor operator and assumed that the even signal $x_e(n)$ remains unchanged and the odd signal is predicted by the $P(x_e(n))$. The difference between the original and predicted signals, which is denoted as $d(n)$, is defined as a high-frequency component and formulated in Eq. (6).

$$d(n) = x_o(n) - P(x_e(n)) \tag{6}$$

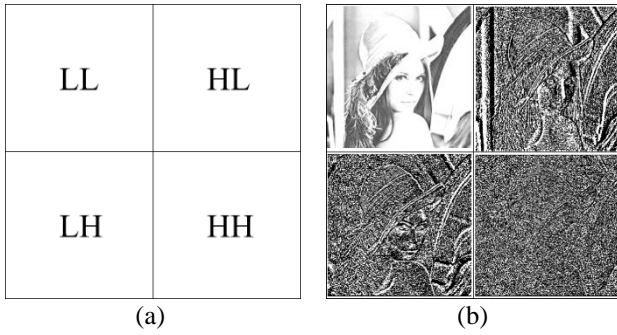


Figure. 4 Lifting wavelet decomposition:
(a) sub-bands regions, (b) sub-bands images

3. Update

This is a step to update the even signal using the $d(n)$ and the update operator $U(\cdot)$. This step produces low-frequency components $l(n)$, which is defined in Eq. (7).

$$l(n) = x_e(n) + U(d(n)) \quad (7)$$

The reconstruction scheme, as shown in Fig. 3(b), can be used to construct the original signal using similar terminology in signal decomposition, especially in prediction and update steps. In this reconstruction, the signal $l(n)$ and $d(n)$ are used as an input in both prediction and update processes to generate the even signal $x_e(n)$ and odd signal $x_o(n)$ using similar predictor and update operators. These generated signals are then merged to get the reconstructed original signal.

Applying LWT on an image will produce the four sub-bands that are a low-frequency sub-band (labeled as LL or Low-Low), horizontal feature sub-band (labeled as HL or High-Low), vertical feature sub-band (labeled as LH or Low-High), and diagonal feature sub-band (labeled as HH or High-High), as illustrated in Fig. 4. The LWT decomposition function lwt of image I with lifted wavelet W can be defined as:

$$lwt(I, W) = [LL, LH, HL, HH] \quad (8)$$

where LL , LH , HL , and HH are low-frequency, horizontal, vertical, and diagonal subbands, respectively. To reconstruct these decomposed images to the original image I can be performed using all sub-bands images by the inverse LWT (ILWT), which is formulated in Eq. (9):

$$ilwt(LL, LH, HL, HH, W) = I \quad (9)$$

The LL sub-band can be decomposed again up to a higher level to get more sub-bands for a specific purpose. For square images of size $N = 2^l$, for any

$l \in \mathbb{N}$, the LL sub-band can be decomposed at most $l = \log_2 N$ levels. The LL sub-band contains the most important of an image and it can be obtained using low pass filtering on both rows and columns, while the other sub-bands are classified as of lesser importance that can be obtained using high-pass filtering. Hence, any modification purpose of an image using wavelet transform is usually done in the LL sub-band, because it contains the most energy of the image. Besides, the changes of the coefficients in low frequency will not affect the energy on average and it will increase the robustness.

3. Proposed method

This study proposed an image watermarking scheme using a lifting wavelet transform (LWT) through singular value decomposition (SVD) and Schur decomposition (SD) with the watermark embedding and extraction processes are described as follows.

3.1 Watermark embedding process

The watermark embedding process mainly uses SVD because it has a good robustness characteristic as it has relatively stable and there is no significant change due to small disturbances in the entire image. In this proposed scheme, the disadvantages of SVD in computationally expensive will be covered through Schur decomposition. This means that the eigendecomposition will be performed on the upper-triangular matrix. The eigenvalues of this eigendecomposition are then used for embedding the watermark, which is embedded on the low-frequency sub-bands of the host image. This terminology is also used for watermark extraction, where the embedded watermark will be recovered using composite transforms in the watermark embedding process.

Given a square image I as a host image and a watermark W . The proposed watermark embedding process has summarized in Algorithm 1.

Algorithm 1: Watermark embedding

Input: Host image I , watermark image W , and embedding level parameter α .

Output: Watermarked image I' .

- Step 1. Perform lifting wavelet transform in Eq. (8) on the host image I to get the LL_I , LH_I , HL_I , and HH_I sub-bands.
- Step 2. Perform Schur decomposition in Eq. (3) on the LL_I sub-band to get the Schur vector P_I and upper-triangular matrix T_I .
- Step 3. Perform SVD in Eq. (1) on the T_I to get the singular values matrix S_I , and singular vectors matrices U_I and V_I .

- Step 4. Perform SVD in Eq. (1) on the watermark image W to get the singular values matrix S_W , and singular vectors matrices U_W and V_W .
- Step 5. Embed the watermark vector S_W into S_I using the embedding level parameter α , that is $S'_I = S_I + \alpha S_W$.
- Step 6. Obtain the new upper-triangular matrix T'_I using SVD in Eq. (2) by U_I , S'_I , and V_I^T matrices.
- Step 7. Obtain the new LL'_I sub-band of I using Schur transform in Eq. (4) by P_I and T'_I .
- Step 8. Construct the watermarked image I' using inverse lifting wavelet transform in Eq. (9) on LL'_I , LH_I , HL_I , and HH_I sub-bands.

The performance of the proposed watermark embedding algorithm is evaluated using the imperceptibility quality level by the peak signal to noise ratio (*PSNR*) and structural similarity index measure (*SSIM*) through comparing the original and watermarked images [34-35], which are formulated in Eq. (10) and Eq. (12), respectively.

3.2 Watermark extraction process

The image watermarking scheme should be able to recover the original watermark from the watermarked image, even when the watermarked image has been distorted by an attack. In this study, the watermark extraction process is proposed in Algorithm 2.

Algorithm 2: Watermark extraction

Input: Watermarked image I' , host image I , watermark image W , and embedding level parameter α .

Output: Recovered watermark image W' .

- Step 1. Perform lifting wavelet transform in Eq. (8) on the watermarked image I' to get the LL'_I , LH'_I , HL'_I , and HH'_I sub-bands.
- Step 2. Perform Schur decomposition in Eq. (3) on the LL'_I sub-band to get the Schur vector P'_I and upper-triangular matrix T'_I .
- Step 3. Perform SVD in Eq. (1) on the T'_I to get the singular values matrix S'_I , and singular vectors matrices U'_I and V'_I .
- Step 4. Perform lifting wavelet transform in Eq. (8) on the host image I to get the LL_I , LH_I , HL_I , and HH_I sub-bands.
- Step 5. Perform Schur decomposition in Eq. (3) on the LL_I sub-band to get the Schur vector P_I and upper-triangular matrix T_I .
- Step 6. Perform SVD in Eq. (1) on the T_I to get the singular values matrix S_I , and singular vectors matrices U_I and V_I .
- Step 7. Extract the watermark vector S'_W from S'_I by the embedding level parameter α , that is $S'_W = (S'_I - S_I)/\alpha$.

- Step 8. Perform lifting wavelet transform in Eq. (8) on the watermark image W to get the LL_W , LH_W , HL_W , and HH_W sub-bands.
- Step 9. Perform Schur decomposition in Eq. (3) on the LL_W sub-band to get the Schur vector P_W and upper-triangular matrix T_W .
- Step 10. Perform SVD in Eq. (1) on the T_W to get the singular values matrix S_W , and singular vectors matrices U_W and V_W .
- Step 11. Construct the recovered watermark W' using Eq. (2) by U_W , S'_W , and V_W^T .

Using the terminology in the watermark embedding process, the watermark extraction also uses the composite transforms that are LWT, Schur, and SVD. This proposed algorithm is a non-blind algorithm, where the extraction process requires support from both the original host. The input of Algorithm 2 can be a watermarked image with or without any attacks. Then, the recovered watermark as the output is compared to the original watermark to evaluate the robustness. This comparison is measured by the normalized correlation (*NC*) [34-35], which is formulated in Eq. (13).

4. Experimental setup

This section explains the experimental setup to evaluate the proposed method performance, including the dataset preparation, preprocessing, experiment design, and performance evaluation.

4.1 Dataset Preparation

This experiment uses the public standard test images, which are Lenna, Peppers, Mandril, House, Pirate, and Cameraman images, as shown in Fig. 5. Those images are commonly used in image

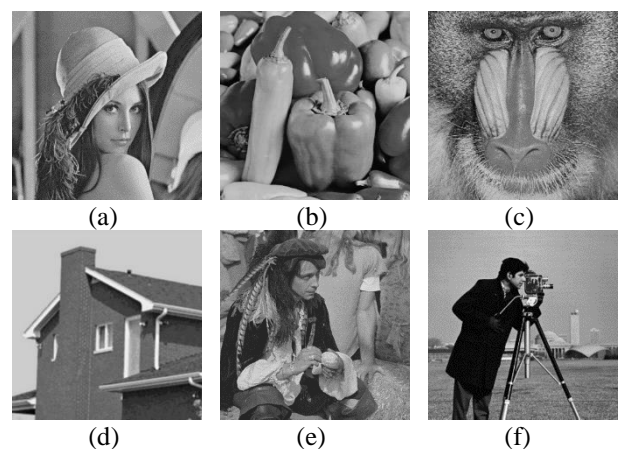


Figure 5 Standard test images of: (a) Lenna, (b) Peppers, (c) Mandril, (d) House, (e) Pirate, and (f) Cameraman

processing, analysis, and machine vision researches and are found frequently in some literature. As the proposed method is performed in the two-dimensional images domain, then various grayscale images are used for simulation and evaluation.

4.2 Image preprocessing

Image preprocessing is performed to ensure that all of the image datasets are in the same type and class to avoid any error in the digital image watermarking process. The image type in this experiment is a two-dimensional image or grayscale image, while the image class for the digital image watermarking process is a double-precision class.

4.3 Experimental design

The first experiment performs the watermark embedding process as described in Algorithm 1 with a watermarked image as an output. The performance of this process is measured by imperceptibility that is measured by the *PSNR* and *SSIM*, respectively. After embedding the watermark, the next experiment conducts the watermark extraction from the watermarked image, which is described in Algorithm 2. To evaluate the robustness, the correlation between the original and extracted watermark is measured through the *NC* value. The performances in both imperceptibility and robustness are then compared to the other similar schemes to evaluate the contribution in the image watermarking field research.

4.4 Performance evaluation

The performance of the proposed method is measured based on the output of the embedding and extraction processes. Imperceptibility is used to assess the watermark embedding performance. Theoretically, adding a watermark signal into the host image will reduce the visual quality. This performance is measured by the *PSNR* and *SSIM*. The *PSNR* of two images C and C^* with $M \times M$ of size is formulated by Eq. (10).

$$PSNR(C, C^*) = 10 \log \frac{C_{max}}{MSE} \quad (10)$$

$$MSE = \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M (C_{i,j} - C^*_{i,j})^2 \quad (11)$$

where C_{max} is the maximum pixel value of the image C and MSE is the mean square error between image C and C^* . While the *SSIM* of two images C and C^* is formulated by Eq. (12).

$$SSIM(C, C^*) = \frac{\mu_C \mu_{C^*} + d_1}{\mu_C^2 + \mu_{C^*}^2 + d_1} \cdot \frac{\mu_{CC^*} + d_2}{\sigma_C^2 + \sigma_{C^*}^2 + d_2} \quad (12)$$

where μ_C and μ_{C^*} are the average of C and C^* , μ_{CC^*} is a covariant between image C and C^* , σ_C and σ_{C^*} are the deviation standard of image C and C^* , while d_1 and d_2 are the denominator parameters. A higher *PSNR* and *SSIM* values between two images represent better imperceptibility.

For the extraction process, the performance of the image watermarking method is evaluated through a robustness level, which is measured by the *NC* value between the original and recovered watermark, as formulated in Eq. (13).

$$NC = \frac{\sum_{i=1}^N \sum_{j=1}^N w_{i,j} w^*_{i,j}}{\sqrt{\sum_{i=1}^N \sum_{j=1}^N w_{i,j}^2} \sqrt{\sum_{i=1}^N \sum_{j=1}^N w^{*2}_{i,j}}} \quad (13)$$

where $w_{i,j}$ and $w^*_{i,j}$ are the pixel values of image C and C^* . A higher *NC* value between two images represents better robustness.

5. Experimental results and discussion

This experiment is conducted to evaluate the proposed image watermarking method based on the lifting wavelet transform through Schur and singular values decomposition. Various host images, which are Lena, Peppers, Mandril, House, and Pirate, are used to evaluate the proposed method performance. While the Cameraman image is used as a watermark and simulated in various sizes that are 256×256 , 128×128 , and 64×64 .

5.1 Watermark embedding performance

The watermark embedding performance of the proposed method is evaluated through the visual quality or imperceptibility of the watermarked image after embedded the watermark. The samples of the

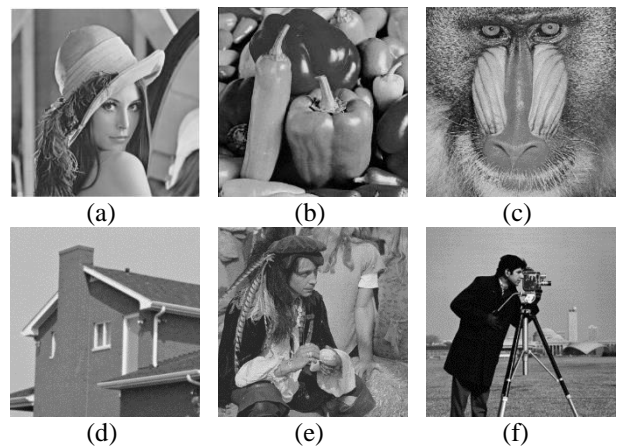


Figure. 6 Watermarked images of (a) Lena, (b) Peppers, (c) Mandril, (d) House, (e) Pirate, and watermark image: (f) Cameraman

Table 1. The visual quality of the watermarked images

Host	PSNR (dB)		
	256×256	128×128	64×64
Lena	37.6649	43.4147	49.2179
Peppers	37.6620	43.7783	49.2548
Mandril	37.6230	43.7168	49.0365
House	37.6124	44.0962	49.1369
Pirate	37.6270	43.6120	49.3619
Average	37.6379	43.7236	49.2016

Table 2. The similarity level of watermarked images

Host	SSIM (%)		
	256×256	128×128	64×64
Lena	99.86	99.94	99.98
Peppers	99.80	99.94	99.97
Mandril	99.90	99.96	99.99
House	99.14	99.84	99.96
Pirate	99.86	99.92	99.98
Average	99.72	99.92	99.98

visual quality of the watermarked images at an embedding level of 0.05 are illustrated in Fig. 6. The similarity between both images can be described by *SSIM*. The PSNR average of host images with various watermark sizes is presented in Table 1, which are 37.6379 dB, 43.7236 dB, and 49.2016 dB, respectively. Table 1 shows that embedding a smaller watermark will produce the watermarked image more perceptible.

The perceptibility of the watermarked images is also measured by *SSIM* to present the similarity between the watermarked and host images. The *SSIM* of the watermarked image is influenced by the watermark size, embedding a smaller watermark will increase the similarity between the original and watermarked images.

Table 2 shows the *SSIM* of various host images with various watermark sizes of 256×256 , 128×128 , and 64×64 gives an average of 99.72% (0.9972), 99.92% (0.9992), and 99.98% (0.9998), respectively. It can be concluded that the proposed watermark could produce a good similarity with more than 99% for all watermarked images.

5.2 Watermark extraction performance

The watermark extraction is performed to recover the embedded watermark, which is extracted using the proposed method from various watermarked images are presented in Fig. 7. The quality of the recovered watermark is measured by *NC* and represents the correlation between the original and recovered watermarks. Table 3 shows the *NC* of the extracted from various watermarked images with watermark sizes of 256×256 , 128×128 , and

64×64 gives an average of 0.99996, 0.99958, and 0.9981, respectively.

Table 3 also shows that the proposed watermark could extract recovered watermark well with high correlation. It can be concluded that embedding higher watermark capacity will be stronger recovered from the watermarked images than a small watermark.

5.3 Robustness against attacks

This experiment is performed on several non-geometrical attacks, which is presented in Table 4. The robustness evaluation is measured by *NC* values by comparing the extracted and original watermarks.

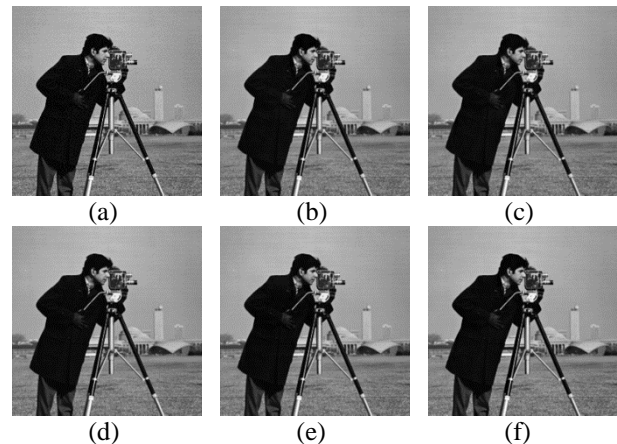


Figure 7 (a) Original watermark image of Cameraman, and recovered watermarks, which are extracted from (b) Lena, (c) Peppers, (d) Mandril, (e) House, (f) Pirate








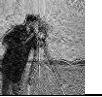











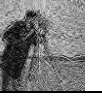
Table 3. The recovered watermark from various watermarked images with no attack

Host	NC		
	256×256	128×128	64×64
Lena	0.9999	0.9998	0.9974
Peppers	1.0000	0.9997	0.9981
Mandril	1.0000	0.9999	0.9983
House	0.9999	0.9992	0.9989
Pirate	1.0000	0.9993	0.9978
Average	0.99996	0.99958	0.9981

Table 4. Type of non-geometrical attacks for robustness evaluation

Type	Specification	Code
Filtering	Gaussian low-pass filter	A01
	Median filter	A02
	Average filter	A03
Noising	Gaussian noise (0; 0.001)	A04
	Salt & pepper noise (0.001)	A05
	Speckle noise (0.001)	A06
Compression	JPEG compression (QF=50)	A07
Histogram	Histogram equalization	A08
Blurring	Motion blur (7; 4)	A09
Sharpening	Sharpening (0.8)	A10

Table 5. The recovered watermark from various distorted watermarked images after filtering

Host	NC		
	Gaussian	Median	Average
Lena 	0.9919 	0.9853 	0.9470 
Peppers 	0.9930 	0.9898 	0.9531 
Mandrill 	0.9720 	0.9084 	0.8730 
House 	0.9986 	0.9983 	0.9863 
Pirate 	0.9868 	0.9716 	0.9256 
Average	0.9885	0.9707	0.9380

Filtering; The filtering attacks, such as Gaussian low-pass, median, and average filterings have been performed on various watermarked images. Table 5 presents the recovered watermark from the distorted watermarked images with its *NC* value.

This experiment shows that although the filtering degrades the watermarked image quality, however, the proposed method still could recover the watermark after Gaussian low-pass, median, and average filterings with the *NC* are 0.9885, 0.9707, and 0.9380, respectively.

Noise additional; The image watermarking should robust against noise interference that degrades the image quality. The common noise distributions, such as Gaussian, salt & peppers, and speckle noises are

used to evaluate how the proposed method could handle the watermark extraction from noised watermarked images. The noise is usually assumed as a random distribution. In this experiment, these noised are performed using a variance of 0.001. The *NC* values of the recovered watermarks from the noised watermarked images are presented in Table 6. It shows that the proposed method robust against Gaussian, salt & peppers, and speckle noises with *NC* average at 0.9888, 0.9985, and 0.9976, respectively. Almost all recovered watermarks have more than a 95% degree of similarity.

Compression; Image compression is an encoding method to reduce image data size for storing, handling, and transmitting content such that it takes less space than the original.

In this experiment, lossy JPEG compression is used to evaluate the robustness. Table 7 presents the *NC* values of the extracted watermark images from the compressed watermarked images. Using different quality factors of compression, the proposed method still robust against the JPEG compression with *NC* average of 0.99992, 0.99996, and 0.99996 for the various compression quality factor of 30, 50, and 70, respectively. A higher-quality factor of JPEG compression will increase the robustness.

Table 6. The recovered watermark from noised watermarked images








Host	NC		
	Gaussian	Salt&Peppers	Speckle
Lena 	0.9901 	0.9989 	0.9984 
Peppers 	0.9909 	0.9987 	0.9991 
Mandrill 	0.9970 	0.9997 	0.9995 
House 	0.9717 	0.9960 	0.9915 
Pirate 	0.9943 	0.9994 	0.9994 
Average	0.9888	0.9985	0.9976

Table 7. The recovered watermark from JPEG compressed watermarked images

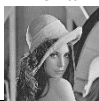



















Host	NC		
	QF=30	QF=50	QF=70
Lena 	0.9999 	1.0000 	1.0000 
Peppers 	0.9999 	1.0000 	1.0000 
Mandrill 	0.9999 	0.9999 	0.9999 
House 	0.9999 	0.9999 	0.9999 
Pirate 	1.0000 	1.0000 	1.0000 
Average	0.99992	0.99996	0.99996

Table 9. The recovered watermark from blurred watermarked images










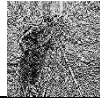






























Host	NC		
	256×256	128×128	64×64
Lena 	0.8557 	0.8576 	0.8784 
Peppers 	0.8784 	0.8676 	0.8612 
Mandrill 	0.7848 	0.8141 	0.8854 
House 	0.9716 	0.9465 	0.9511 
Pirate 	0.8356 	0.8549 	0.8852 
Average	0.8652	0.8681	0.8922

Table 8. The recovered watermark from watermarked images after histogram equalization

Host	NC		
	256×256	128×128	64×64
Lena 	0.9999 	0.9998 	0.9974 
Peppers 	1.0000 	0.9997 	0.9981 
Mandrill 	1.0000 	0.9999 	0.9983 
House 	0.9999 	0.9992 	0.9989 
Pirate 	1.0000 	0.9993 	0.9978 
Average	0.99996	0.9996	0.9981

Histogram equalization; This transformation is usually used to increase the global contrast of an image by adjusting the contrast through its histogram to produce a better intensities distribution.

To evaluate the robustness, histogram





















equalization has been performed on the various watermarked images with different watermark sizes. Table 8 shows that the proposed watermark method could recover the watermark after histogram equalization with high robustness represented by its correlations. It also shows that a smaller watermark will reduce the slight robustness of the proposed image watermarking scheme. The NC average of the recovered watermarks are 0.99996, 0.9996, and 0.9981 for various watermark sizes of 256×256, 128×128, and 64×64, respectively.

Motion blur; This is a type of low-pass filtering, and is usually called smoothing. The motion blur is used to add a motion effect on the image by moving noise and leaving the most object during a time and degree. This effect will produce an average out of changes in pixel intensity.

Thus, the blurred watermarked image will have different intensities to the original watermarked image. The robustness of the proposed method against image motion blur with a length of 7 and angle of 4 degrees is presented in Table 9. It shows that the proposed method still robust against the image blurring attack with NC average of 0.8652, 0.8681, and 0.8922 for various watermark sizes of 256 × 256, 128 × 128, and 64 × 64, respectively.

Sharpening; This is an image enhancement

Table 10. The recovered watermark from sharpened watermarked images

Host	NC		
	256×256	128×128	64×64
Lena 	0.9330 	0.9442 	0.9564 
Peppers 	0.9407 	0.9476 	0.9644 
Mandrill 	0.8758 	0.8975 	0.9450 
House 	0.9794 	0.9662 	0.9752 
Pirate 	0.9090 	0.9262 	0.9504 
Average	0.9276	0.9363	0.9583

technique by highlighting the edges and fine details in an image to produce the apparent sharpness by the local contrast. When the sharpening is performed on the watermarked image, the proposed image watermarking method should be able to extract the embedded watermark from the sharpened watermarked images. Table 10 presents the normalized correlation values of the extracted watermark images from various sharpened watermarked images with a 0.8 sharpening ratio. It shows that the proposed method robust against the image sharpening attack with *NC* average of 0.9276, 0.9363, and 0.9583 for various watermark sizes of 256×256, 128×128, and 64×64, respectively. It also shows that the proposed image watermarking scheme is more robust with a smaller watermark.

5.4 Performance comparison

To ensure the performance achievements, the proposed image watermarking method is compared to the other similar work in both perceptibility and robustness. This comparison is performed based on the most of related work, such as the use of similar images for watermark embedding, watermark sizes, type of attacks that are used for robustness analysis, and similar performance measurement units for perceptibility and robustness evaluation. Related to the proposed method, the most related and similar work, which is the image watermarking based on the

Table 11. The performance comparison of imperceptibility in PSNR and SSIM

Size*	Host	PSNR (dB)		SSIM (%)	
		Prop-osed	[35]	Prop-osed	[35]
256×256	Lena	37.6649	<i>38.1621</i>	99.86	99.92
	Peppers	37.6620	<i>38.1295</i>	99.80	99.89
128×128	Lena	<i>43.4147</i>	38.1625	<i>99.94</i>	99.92
	Peppers	<i>43.7783</i>	38.1539	<i>99.94</i>	99.90
64×64	Lena	<i>49.2479</i>	38.2477	<i>99.98</i>	99.91
	Peppers	<i>49.2548</i>	38.2178	<i>99.97</i>	99.89

Table 12. The performance comparison of robustness against some attacks

Attack Code	256×256		128×128	
	Prop-osed	[35]	Prop-osed	[35]
A01	<i>0.9919</i>	0.9749	0.9946	<i>0.9973</i>
A02	<i>0.9853</i>	0.9685	<i>0.9908</i>	0.9905
A03	<i>0.9470</i>	0.9294	<i>0.9554</i>	0.9539
A04	<i>0.9895</i>	0.9864	0.9978	<i>0.9987</i>
A05	<i>0.9989</i>	0.9985	<i>0.9996</i>	<i>0.9996</i>
A06	<i>0.9984</i>	0.9981	0.9994	<i>0.9998</i>
A07	<i>1.0000</i>	0.9998	<i>0.9999</i>	0.9999
A08	<i>0.9999</i>	0.9924	<i>0.9998</i>	0.9911
A09	<i>0.8557</i>	0.8322	<i>0.8576</i>	0.8569
A10	0.9330	<i>1.0000</i>	0.9442	<i>1.0000</i>

hybrid of DWT, HE, and SVD [35], is chosen as an observed method for performance comparison both in perceptibility and robustness.

For imperceptibility, Table 11 presents the performance comparison between the proposed and observed methods based on the *PSNR* and *SSIM* values (the italic font indicates better perceptibility). The proposed method outperforms the observed method based on the *PSNR* and *SSIM* in the 128×128 and 64×64 of the watermark sizes. However, the proposed method underperformed in 256×256 of the watermark size. Thus, it can be concluded that the proposed method could increase the perceptibility of the watermarked image, especially in the small portion of the watermark.

For robustness, Table 12 presents the performance comparison between the proposed and observed methods based on the *NC* values (the italic font indicates better robustness). Table 12 shows that the proposed method could increase the robustness at almost of non-geometrical attacks unless the sharpening attack. The attack codes in Table 12 refer to the attack code in Table 4.

Table 13. The performance comparison between the proposed method to the other various image watermarking domains

Methods	PSNR (dB)	NC
Proposed	43.42	0.99
LWT+Firefly [10]	37.98	0.99
DWT+DCT [26]	36.52	0.95
DCT+Schur [29]	30.00	0.70
DCT+DWT+SVD [31]	35.94	0.99
LWT+SVD [34]	39.68	0.96

The other performance comparison between the proposed method across the other various image watermarking domains is presented in Table 13. The proposed method is fairly comparable both in perceptibility and robustness performances.

6. Conclusion

The lifting scheme-based image watermarking through Schur and SVD has been presented in this paper. This method is mainly proposed to improve both the perceptibility and robustness of image watermarking. In this method, the watermark image is embedded on the low-frequency sub-band of the host image through the Schur decomposition and SVD. The Schur decomposition is proposed as the intermediary process before the watermark image is embedded through SVD.

Experimental results show that the proposed method achieves the perceptibility of the watermarked image more than 99% on average based on its similarities to the original image. The visual quality of the watermarked image also increases with more than 40dB on average. While the robustness of the proposed method is vary depending on the attack type, where it could recognize and recover the original watermark from almost all attack experiments with more than 90% of normalized correlation average. Hence, it can be concluded that the proposed method is feasible to be applied to watermark applications especially to overcome the non-geometrical attacks.

For future works, this experiment can be extended to the color image watermarking and also its resilience to a geometrical attack for further analysis of the proposed method performances. It is also suggested to conduct further experiments to evaluate the complexity of the proposed method.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

All authors contributed both the concepts and contents of this study. Conceptualization and methodology, A. Marjuni and O.D. Nurhayati; formal analysis and investigation, A. Marjuni and O. D. Nurhayati; resources and data curation, A. Marjuni and O. D. Nurhayati; writing—original draft preparation, A. Marjuni; writing—review and editing, A. Marjuni and O. D. Nurhayati; visualization, A. Marjuni; supervision, A. Marjuni and O. D. Nurhayati; project administration, A. Marjuni and O. D. Nurhayati. All authors have read and approved the final manuscript.

References

- [1] W. H. H. Alshoura, Z. Zainol, J. Sen Teh, and M. Alawida, "A New Chaotic Image Watermarking Scheme Based on SVD and IWT", *IEEE Access*, Vol. 8, pp. 43391-43406, 2020.
- [2] S. K. Shrivastava and S. K. Mahendran, "Digital Watermarking using Lifting Wavelet Transform of Crowd Sourced Images", *International Journal of Engineering & Technology*, Vol. 7, No. 2.20, p. 357-361, 2018.
- [3] S. B. B. Ahmadi, G. Zhang, S. Wei, and L. Boukela, "An Intelligent and Blind Image Watermarking Scheme based on Hybrid SVD Transforms using Human Visual System Characteristics", *The Visual Computer*, 37, pp. 385-409, 2021.
- [4] P. Khare and V. K. Srivastava, "A Novel Dual Image Watermarking Technique using Homomorphic Transform and DWT", *Journal of Intelligent Systems*, Vol. 30, No. 1, pp. 297-311, 2021.
- [5] Q. Su, L. Su, G. Wang, L. Li, and J. Ning, "A Novel Colour Image Watermarking Scheme Based on Schur Decomposition", *International Journal of Embedded System*, Vol. 12, No. 1, pp. 31-38, 2020.
- [6] A. K. Gupta and M. S. Raval, "A Robust And Secure Watermarking Scheme Based on Singular Values Replacement", *Sadhana*, Vol. 37, No. 4, pp. 425-440, 2012.
- [7] T. Khanam, P. K. Dhar, S. Kowsar, and J. M. Kim, "SVD-based Image Watermarking using the Fast Walsh-Hadamard Transform, Key Mapping, and Coefficient Ordering for Ownership Protection", *Symmetry*, Vol. 12, No. 1, pp. 1-20, 2020.
- [8] M. Barni, F. Bartolini, V. Cappellini, and A. Piva, "A DCT-domain System for Robust Image Watermarking", *Signal Processing*, Vol. 66, No. 3, pp. 357-372, 1998.

- [9] J. Kong and D. Zhou, "Digital Image Watermarking Based on Second Generation Wavelet Transform", *Applied Mechanics and Materials*, Vol. 241-244, pp. 3026-3029, 2013.
- [10] B. Kazemivash and M. E. Moghaddam, "A Robust Digital Image Watermarking Technique using Lifting Wavelet Transform and Firefly Algorithm", *Multimedia Tools and Applications*, Vol. 76, No. 20, pp. 20499–20524, 2017.
- [11] M. Gaaed and M. Tahar, "Digital Image Watermarking based on LSB Techniques: A Comparative Study", *International Journal of Computer Applications*, Vol. 181, No. 2630-36, 2018.
- [12] K. K. Singh and S. Dwivedi, "Digital Watermarking using Asymmetric Key Cryptography and Spatial Domain Technique", *International Journal of Advance Research in Computer Science and Management Studies*, Vol. 2, No. 8, pp. 65-72, 2014.
- [13] G. J. Lee, E. J. Yoon, and K. Y. Yoo, "A New LSB based Digital Watermarking Scheme with Random Mapping Function", In: *Proc. of 2008 International Symposium on Ubiquitous Multimedia Computing*, Australia, pp. 130-134, 2008.
- [14] S. Kumar and T. Singh, "Performance Improvement of Simple LSB watermarking using SVD", In: *Proc. of the Innovative Applications of Computational Intelligence on Power, Energy and Controls with their Impact on Humanity*, India, pp. 133-137, 2014.
- [15] A. Bamatraf, R. Ibrahim, and S. Najib, Mohd, "A New Digital Watermarking Algorithm Using Combination of Least Significant Bit (LSB) and Inverse Bit", *Journal of Computing*, Vol. 3, No. 4, pp. 1-8, 2011.
- [16] T. K. Tsui, X. P. Zhang, and D. Androutsos, "Color Image Watermarking Using Multidimensional Fourier Transforms", *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 1, pp. 16-28, 2008.
- [17] P. Tao and A. M. Eskicioglu, "An Adaptive Method for Image Recovery in the DFT Domain", *Journal of Multimedia*, Vol. 1, No. 6, pp. 36-45, 2006.
- [18] Y. Wang, J. F. Doherty, and R. E. Van Dyck, "A Wavelet-based Watermarking Algorithm for Ownership Verification of Digital Images", *IEEE Transactions on Image Processing*, Vol. 11, No. 2, pp. 77-88, 2002.
- [19] M. S. Hsieh, D. C. Tseng, and Y. H. Huang, "Hiding Digital Watermarks using Multiresolution Wavelet Transform", *IEEE Transactions on Industrial Electronics*, Vol. 48, No. 5, pp. 875-882, 2001.
- [20] Z. H. Wei, P. Qin, and Y. Q. Fu, "Perceptual Digital Watermark of Images using Wavelet Transform", *IEEE Transactions on Consumer Electronics*, Vol. 44, No. 4, pp. 1267-1272, 1998.
- [21] S. D. Lin and C. Chen, "A Robust DCT-based Watermarking for Copyright Protection", *IEEE Transactions on Consumer Electronics*, Vol. 46, No. 3, pp. 415-421, 2000.
- [22] J. C. Patra, J. E. Phua, and C. Bornand, "A Novel DCT Domain CRT-based Watermarking Scheme for Image Authentication Surviving JPEG Compression", *Digital Signal Processing: A Review Journal*, Vol. 20, No. 6, pp. 1597-1611, 2010.
- [23] A. K. Singh, "Improved Hybrid Algorithm for Robust and Imperceptible Multiple Watermarking using Digital Images", *Multimedia Tools and Applications*, Vol. 76, No. 6, pp. 8881-8900, 2017.
- [24] R. A. Sadek, "SVD based Image Processing Applications: State of the Art, Contributions and Research Challenges", *International Journal of Advanced Computer Science and Applications*, Vol. 3, No. 7, pp. 26-34, 2012.
- [25] N. Abbadi and A. Mohsin, "Blind Digital Images Tampering Detection Based on Singular Value Decomposition", *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 6, pp. 338-348, 2020.
- [26] A. Akter, N. E. Tajnina, and M. A. Ullah, "Digital Image Watermarking Based on DWT-DCT: Evaluate for a New Embedding Algorithm", In: *Proc. of the 2014 International Conference on Informatics, Electronics, and Vision*, Bangladesh, pp. 1-6, 2014.
- [27] K. Deb, M. S. Al-Seraj, M. M. Hoque, and M. I. H. Sarkar, "Combined DWT-DCT based Digital Image Watermarking Technique for Copyright Protection", In: *Proc. of the 2012 7th International Conference on Electrical and Computer Engineering*, Bangladesh, pp. 458-461, 2012.
- [28] M. Jiansheng, L. Sukang, and T. Xiaomei, "A Digital Watermarking Algorithm Based on DCT and DWT", In: *Proc. of the 2009 International Symposium on Web Information Systems and Applications*, China, pp. 104-107, 2009.
- [29] H. B. Razafindradina, N. R. Razafindrakoto, and P. A. Randriamitantsoa, "Improved Watermarking Scheme Using Discrete Cosine Transform and Schur Decomposition", *International Journal of Computer Science and Network*, Vol. 2, No. 4, pp. 25-31, 2013.
- [30] A. Sverdlov, S. Dexter, and A. M. Eskicioglu,

- “Robust DCT-SVD Domain Image Watermarking for Copyright Protection: Embedding Data in All Frequencies”, In: *Proc. of the 13th European Signal Processing Conference, EUSIPCO 2005*, pp. 29-32, 2005.
- [31] M. I. Khan, M. M. Rahman, and M. I. H. Sarker, “Digital Watermarking for Image Authentication Based on Combined DCT, DWT and SVD Transformation”, *International Journal of Computer Science*, Vol. 10, Issue 3, pp. 223-230, 2013.
- [32] M. P. Satyanarayana and P. R. Kumar, “A Robust Digital Image Watermarking Scheme using Hybrid DWT-DCT-SVD Technique”, *International Journal of Computer Science and Network Security*, Vol. 10, No. 10, pp. 185-192, 2010.
- [33] A. Kala, “Robust Lossless Image Watermarking in Integer Wavelet Domain using SVD”, *International Journal of Computer Science Engineering*, Vol. 2, No. 02, pp. 30-35, 2013.
- [34] M. Talbi and M. S. Bouhlel, “Lifting Wavelet Transform and Singular Values Decomposition for Secure Image Watermarking”, *International Journal of Computer and Information Engineering*, Vol. 11, No. 7, pp. 860-855, 2017.
- [35] J. Liu, J. Huang, Y. Luo, L. Cao, S. Yang, D. Wei, and R. Zhou, “An Optimized Image Watermarking Method Based on HD and SVD in DWT Domain”, *IEEE Access*, Vol. 7, pp. 80849-80860, 2019.
- [36] S. Ben Ftima, M. Talbi, and T. Ezzedine, “Lifting Wavelet Transform and Singular Values Decomposition for Secure Image Watermarking”, *International Journal of Computer and Information Engineering*, Vol. 11, No. 7, pp. 860-865, 2017.
- [37] D. C. Akua, S. M. Sani, B. G. Bajoga, and A. D. Usman, “An Improved Approach to Robust Digital Image Watermarking Algorithm Using Lifting Wavelet Transform Technique for Copyright Protection”, *Journal of Science Technology and Education*, Vol. 8, No. 1, pp. 146-153, 2020.
- [38] V. Pomponiu, D. Cavagnino, and M. Botta, “On the Security of the Schur-based Watermarking Schemes”, In: *Proc. of the 2015 IEEE International Conference on Digital Signal Processing (DSP)*, Singapore, pp. 215-218, 2015.
- [39] L. Y. Hsu and H. T. Hu, “A Reinforced Blind Color Image Watermarking Scheme Based on Schur Decomposition”, *IEEE Access*, Vol. 7, pp. 107438-107452, 2019.
- [40] A. A. Mohammad, “A New Digital Image Watermarking Scheme Based on Schur Decomposition”, *Multimedia Tools and Applications*, Vol. 59, No. 3, pp. 851-883, 2012.
- [41] Q. Su and B. Chen, “An Improved Color Image Watermarking Scheme Based on Schur Decomposition”, *Multimedia Tools and Applications*, Vol. 76, No. 22, pp. 24221-24249, 2017.
- [42] A. Guntoro and M. Glesner, “A Lifting-based Discrete Wavelet Transform and Discrete Wavelet Packet Processor with Support for Higher-Order Wavelet Filters”, In: *Proc. of the 19th IFIP WG 10.5/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, Greece, pp. 154-173, 2008.
- [43] E. W. K. Full, S. Metkar, and H. C. Kamble, “Image Watermarking by Schur Decomposition”, *International Journal of Information & Computation Technology*, Vol. 4, No. 12, pp. 1155-1159, 2014.
- [44] H. Shi, F. Lv, and Y. Cao, “A Dual Color Image Watermarking Scheme Based on non-overlapping Blocks with Circulation”, *Journal of Computer*, Vol. 9, No. 8, pp. 1871-1879, 2014.