IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 13, 2020, accepted February 28, 2020, date of publication March 4, 2020, date of current version March 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978186

A New Chaotic Image Watermarking Scheme Based on SVD and IWT

WAFA' HAMDAN ALSHOURA, ZURINAHNI ZAINOL, JE SEN TEH, AND MOATSUM ALAWIDA

School of Computer Sciences, Universiti Sains Malaysia (USM), George Town 11800, Malaysia

Corresponding authors: Wafa' Hamdan Alshoura (wafahamdan@student.usm.my) and Je Sen Teh (jesen_teh@usm.my)

This work was supported in part by the Ministry of Education Malaysia through the Fundamental Research Grant Scheme (FRGS) under Project FRGS/1/2019/ICT05/USM/02/1.

ABSTRACT Image watermarking schemes based on singular value decomposition (SVD) have become popular due to a good trade-off between robustness and imperceptibility. However, the false positive problem (FPP) is the main drawback of SVD-based watermarking schemes. The singular value is the main cause of FPP issues because it a fixed value that does not hold structural information of an image. In this paper, a new SVD-based image watermarking scheme that uses a chaotic map is proposed to overcome this issue. The secret key is first extracted from both the host and watermark image. This key is used to generate a new chaotic matrix and chaotic multiple scaling factors (CMSF) to increase the sensitivity of the proposed scheme. The watermark image is then transformed based on the chaotic matrix before being directly embedded into the singular value of the host image by using the CMSF. The extracted secret key is unique to the host and the watermark images, which improves security and overcomes FPP issues. Experimental results show that the proposed scheme fulfils all watermarking requirements in terms of robustness, imperceptibility, security, and payload. Furthermore, it achieves high robustness with different scaling factors, and outperforms several existing schemes.

INDEX TERMS Chaotic map, image watermark, integer wavelet transform, IWT, singular value decomposition, SVD.

I. INTRODUCTION

With the widespread growth of digital applications and improved network technology services, the demand for enhanced data protection methods have greatly increased due to illegal copying, editing, distribution, and integrity problems. Watermarking technologies have been introduced to provide additional protection on top of existing cryptographic technologies. Digital watermarking uses embedding or hiding methods to provide copyright protection for multimedia data. Embedding and extraction are the basic processes of digital watermarking schemes, whereby the embedding process hides watermark information in another piece of digital data such as images, whereas the extraction process involves retrieving the embedded information. In other words, digital watermarking can be referred to as an embedding method for secret information. Digital watermarking does not only provide copyright protection but can also be used for other purposes such as content identification and authentication, digital forensics, tamper detection, broadcast monitoring, fingerprinting, and media file archiving [1].

Digital image watermarking can be classified into two categories: visible and invisible. Visible watermarks are used to insert logos or labels into a host image as proof of content ownership. Visible watermarks are easy to recognize but are easily attacked and removed by adversaries [2]. In contrast, invisible watermarks are more commonly used because it is difficult to be perceived by the human visual system (HVS). Invisible watermarks are based on the notion of embedding watermark information into unknown parts of a host image. All existing invisible watermarking schemes are subject to three main requirements: robustness, imperceptibility and security. A robust watermarking scheme is one which ensures that the extracted watermark remains recognizable even after the watermarked image has been subjected to geometrical and non-geometrical attacks. An imperceptible watermarking scheme is one which will not lead to a perceptible or visible difference between the host and watermarked image. As such, it is difficult to find the embedded watermark or identify patterns that are caused by the embedding process [3]. Lastly,

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed Farouk.

a secure watermarking scheme is one that is secure against various attacks [4], [5].

Watermark information can be embedded based on either the spatial or transform domains. Spatial-based watermarking techniques involve changing the least significant bits (LSB) of the grayscale pixel of the host image. Although the spatial-based technique has low computational complexity [6], it is easy to destroy an embedded watermark by changing just one bit of the pixel value. In the transform domain, a watermark can be embedded after transforming a digital image from the spatial domain into frequency coefficients based on methods such as discrete cosine transform (DCT) [7], discrete Fourier transform (DFT) [8], discrete wavelet transform (DWT) [9]–[12], integer wavelet transform (IWT) [13], [14] and singular value decomposition (SVD) [13], [15]–[20]. In general, transform domain schemes are robust against well-known attacks and but perform poorly in terms of imperceptibility. To achieve both robustness and imperceptibility, a combination of transformation techniques have been used in image watermarking schemes [21]–[24]. However, the data capacity of the embedded watermark information is limited because frequency domain watermarks significantly degrade when embedding larger watermarks. Hybrid transforms have also been used to develop new watermarking techniques such as DWT+SVD [25], [26], DWT+DCT [27], DCT+SVD [28], [29], DWT+DCT+SVD [30], [31], and IWT+SVD [13].

In SVD-based image watermarking schemes, a watermark image is decomposed into three matrices, one of which will be selected for the embedding process. Most SVD watermarking schemes perform the embedding process based on one of two techniques: The first technique decomposes a watermark, W via SVD into three matrices, two of which (U_W and V_W) describe the geometric properties of W whereas the third (S_W) describes its luminescence. S_W consists of non-negative singular values arranged in descending order. The host image, H is also decomposed into three matrices, U_H, S_H and V_H using SVD. The singular values of S_W are then embedded into the singular values of S_H of the host image [17], [32]. U_W and V_W are used as "keys" which are used in the extraction process. In the second technique, a watermark is directly embedded into the S_H matrix of the host image [33]. SVD is again performed on S_H and the resulting U and V matrices of the second SVD process is used as "keys" for extraction. These keys can be considered as side information in terms of watermarking [34]. Unfortunately, both of these techniques suffer from the false positive problem (FPP) in which an adversary can extract corrupt watermarks by exploiting this side information in different attacks, then claim rightful ownership to those illegally watermarked images. The three main attacks in FPP are as follows: (1) Assuming the same host image has been used in two separate SVD watermarking processes, A and B, the side information from A can be used to extract a corrupted watermark from B, and vice versa. (2) An adversary can embed a fake watermark into an already watermarked image then extract the fake watermark to claim ownership of the image. (3) By using side information from watermarking an image A, an adversary can extract the same watermark from an arbitrary image, B, thus claiming ownership of B.

To overcome the FPP and other security issues, several solutions have been proposed such as the use of hashing [35], encryption [32], digital signature [36], principle component embedding [37], [38] and singular vector embedding [13]. In the hashing methodology [35], the side information, U_W and V_W are hashed by using a one-way hash function and stored during the embedding process. During extraction, the side information used in the extraction will be hashed and compared against the stored hash values. If they match, the side information is successfully authenticated and the extraction is completed. Otherwise, extraction is aborted. The use of encryption involves encrypting the watermark prior to the embedding process [32]. During extraction, successful decryption must be performed in order to recover the original watermark. Otherwise, an invalid, arbitrary image will be produced. The digital signature-based scheme [36] is similar to the other methods, whereby the watermark is directly embedded into S_H. In addition, the digital signature of the side information is also embedded into the host image. During extraction, the digital signature can be used to authenticate the extracted watermark. All of these methods have demonstrated a good trade-off between robustness and imperceptibility. However, these methods still rely on using side information as the extraction key, and may still be susceptible to different variants of FPP. Also, the side information is not sensitive to small changes, which leads to easy estimation of the watermark. Furthermore, extra authentication processes are required during the extraction process which incurs computational overhead.

Due to the drawbacks of the aforementioned schemes, new schemes specifically addressing FPP have been proposed. One of these methods embeds the principal component (PC) instead of the singular matrix, S_W or the watermark itself [37], [38]. PC includes both U and S matrices, whereas remaining matrix, V is used as the extraction key. The principle component of the watermark, PC_W is generated using SVD and embedded into S_H of the host image. Without V_W, an adversary cannot extract the embedded watermark, thus circumventing FPP. However, this method is vulnerable to geometrical and non-geometrical attacks because the U_W holds the structure of the image and has high sensitivity to small change. Another method aimed to overcome FPP embeds U_W instead of S_W [13]. The side information, V_W and S_W are used as extraction keys. This idea achieves high imperceptibility but is still not robust against different well-known attacks.

The scale factor plays an important role in image watermarking because it determines the level of embedding. A single scaling factor (SSF) can lead to a stable level of embedding but it cannot fulfil the desired balance between robustness and imperceptibility. For example, a small SSF value can lead to high imperceptibility but lower robustness against common attacks. On the other hand, a large SSF value improves robustness while sacrificing imperceptibility. Thus, researchers have used multiple scaling factors (MSF) instead of SSF to achieve the desired goals of robustness and imperceptibility. Optimal MSF values can be determined by using optimization algorithms such as genetic algorithms (GA) [39], particle swarm optimization (PSO) [15], multi-objective ant colony optimization (MOACO) [13] and differential evolution (DE) [18], [19], [28]. MSF-based optimization techniques achieve all requirements of a good image watermarking scheme. However, SVD based on optimization techniques are computationally expensive and have a limited range of suitable MSF values for each watermarking scheme.

In this paper, a new chaos-based SVD image watermarking scheme in the frequency coefficient domain is proposed. The proposed scheme uses a new embedding strategy whereby the entire transformed watermark image is embedded into the S_{LL} matrix of the decomposed host image. The novelty of this proposed scheme is the use of a chaotic watermark which is embedded into the host image based on a secret key is generated from both the host and watermark images. In addition, MSF values are generated by using a chaotic map, which we denote as CMSF. The host image first undergoes IWT transform, to produce an approximate image LL which is then decomposed using SVD to obtain S_{LL}. The watermark image is transformed into a chaotic matrix by using chaotic maps, and directly embedded into S_{LL}. A secret key is extracted from the host image and the watermark image to generate initial conditions and control parameters for the chaotic maps. The secret key is also used for authentication during the extraction process to address FPP. Moreover, using chaotic maps to generate CMSF values negates the need for optimization algorithms, leading to higher efficiency. The proposed scheme achieves the desired requirements of robustness, imperceptibility, high capacity, and security of an image watermarking scheme. The proposed scheme is resistant to various well-known attacks, and can support a wide range of CMSF values. The proposed scheme also has high sensitivity to the secret key.

The rest of this paper is organized as follows: Section II, briefly introduces SVD, IWT, and chaotic maps. The proposed scheme is then detailed in Section III, followed by the experimental results in Section IV that include imperceptibility and robustness tests, FPP analysis, secret key sensitivity as well as comparative analysis with others schemes. Finally, the paper is concluded in Section V.

II. PRELIMINARIES

A. SINGULAR VALUE DECOMPOSITION (SVD)

SVD is a numerical tool that decomposes any matrix into three matrices. An image can be considered as a matrix, I that consists of 8-bit numbers with a variety dimensions depending on the type of image. For example, a grayscale image has a dimension of $1 \times M \times N$ whereas a color image has a dimension of $3 \times M \times N$, where M and N are height (number of rows) and width (number of columns) of the matrix, respectively. An image can also be denoted as a matrix I of real numbers $\mathbb{R}^2$. SVD can be applied on I and the results are three matrices, U, S, and V. Anyone can recover the original matrix with knowledge of these three matrices. SVD can be defined as

$$SVD(I) = U_I S_I V_I = \sum U_I * S_I * {V_I}^T, \tag{1}$$

where U_I and V_I are orthogonal matrices of $\mathbb{R}^2$. These matrices, which we henceforth refer to as the left and right singular vectors, are highly sensitivity to changes in the original matrix, I. S_I is a diagonal matrix of $\mathbb{R}^2$ that consists of positive singular values in descending order. Researchers leverage upon the following properties of SVD in designing image watermarking schemes:

- The diagonal values in S are highly stable. When there are small changes to these singular values, there will be barely any effect on the resulting image pixels. Thus, watermark information can be embedded without affecting the visual perception of the host image.
- Due to how the singular values in S are in descending order, the smaller values are located towards the end of the matrix. Adding or updating these smaller values during the recovery stage has minimal effect on image quality. In addition, adding new small values in all positions in S also has minimal effect on image quality.

B. INTEGER WAVELET TRANSFORM (IWT)

Lifting wavelet transform (LWT) is a signal processing tool used in the many applications such as image processing and compressing. One of the important properties of LWT is that it supports floating point numbers or integers, unlike classical transforms that deal with floating point values. The use of floating point representation in image processing may result in loss of information due to round-off operations. As such, LWT is suitable for image processing because 8-bit integers are used to represent pixels.

IWT is a lifting transform that maps input data to integers without quantization errors and it is also reversible. It consists of three processes which are split, predict, and update. Although it has similar processes to LWT, IWT is more computationally efficient. Figure 1 shows one IWT level of the Lena image, where four sub-bands are generated.

C. CHAOTIC MAP

A chaotic map is a nonlinear dynamical system which can produce random trajectories in different dimensions [40], [41]. Chaotic trajectories have unique properties such as unpredictability, dense orbits, random-like behavior, and nonlinearity [42]. Each chaotic map has at least one chaotic state and system parameter. A high dimensional chaotic map may have more than one state or parameter. Regardless of the number of dimensions, all chaotic maps are iterated functions that generate a sequence of chaotic states starting from an initial chaotic state, under the control of the

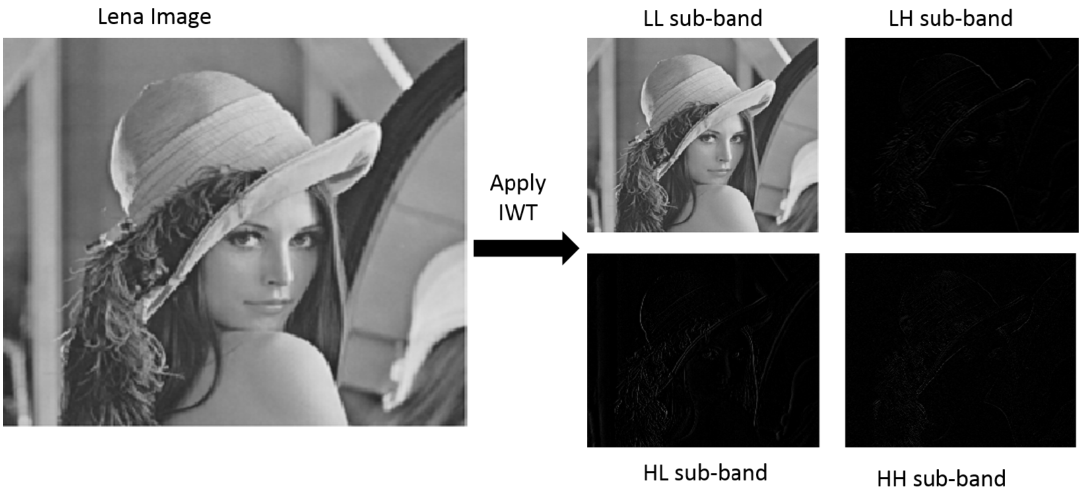


FIGURE 1. IWT sub-bands of Lena.

system parameter. The initial chaotic state and system parameter values are known as the initial conditions. Chaotic maps depict a high sensitivity to these initial conditions, making them suitable for designing cryptographic algorithms [43]. In this paper, we use one-dimensional chaotic maps due to their simplicity and low computational requirements.

The logistic and sine maps are one dimensional unimodal chaotic maps that are widely used areas such as image encryption, hash functions and watermarking. However, their chaotic behaviors are easy to predict and they suffer from a limited chaotic parameter range. In the effort to overcome these drawbacks, Alawida et al. proposed a hybridization method to enhance chaotic behaviors of the logistic and sine maps based on linear and nonlinear functions [44]. The resulting chaotic systems depict enhanced complexity and a larger chaotic parameter range. These maps can be rewritten as

$$x_{n+1} = (r_1^2 \times x_n \times (1 - r \times x_n) + \frac{r_1}{x_n})\ mod\ 1 \qquad (2)$$

$$y_{n+1} = (r_2 \times sin(\pi \times r_2 \times y_n) + \frac{r_2}{y_n})\ mod\ 1 \qquad (3)$$

where x_n and y_n are chaotic state within interval (0, 1), and r_1 and r_2 are system parameters within range $(0, \infty)$. We refer to the modified logistic and sine maps as logistic-G and sine-G respectively.

Figures 2 and 3 show the bifurcation diagrams of the logistic-G and sine-G maps whereas Figures 4 and 5 depict their Lyapunov exponent and fuzzy entropy, respectively. From the bifurcation diagrams, it can be seen that the modified maps have a large chaotic parameter range. The Lyapunov exponent for both maps indicate the existence of chaos over the entire range of system parameter values. In Figure 4, the Lyapunov exponent has large positive values that implies high chaotic sensitivity and fast divergence between chaotic states. However for the sine-G map, there exists small windows of periodicity as indicated by the inverse spikes in the Lyapunov exponent plot as well as the unshaded regions in the bifurcation diagram. Fuzzy entropy is the complexity measure of a chaotic map, whereby a large value corresponds to high complexity. Higher complexity is vital for resisting initial condition or system parameter estimation attacks. Figure 5 shows that the fuzzy entropy for both maps is high for the entirety of the control parameter range. Based on these properties, we have employed the enhanced logistic and sine maps in the proposed watermarking scheme.

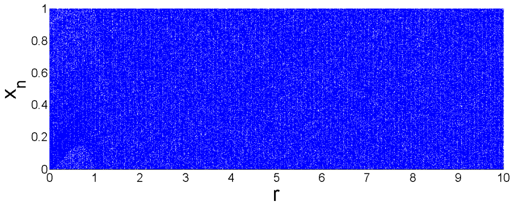


FIGURE 2. Bifurcation diagram of the logistic-G map.

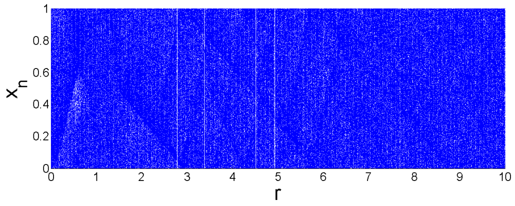


FIGURE 3. Bifurcation diagram of the sine-G map.

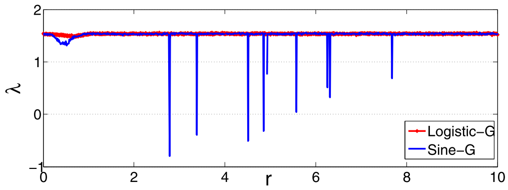


FIGURE 4. Lyapunov exponent of the logistic-G and sine-G maps.

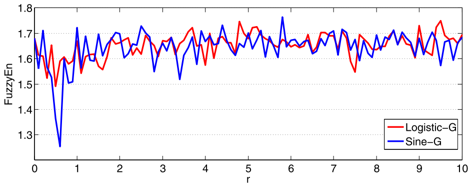


FIGURE 5. Fuzzy entropy of the logistic-G and sine-G maps.

III. PROPOSED SCHEME

The proposed scheme consists of three phases: key generation, embedding, and extraction. The key generation phase involves extracting information from the host and watermark images, then hashing them to generate the secret key of the proposed scheme. This secret key is used to generate initial conditions and system parameters of the chaotic maps, which are then iterated to produce a matrix that will be used to transform the watermark prior to the embedding process. The use of the transformed watermark protects the original watermark against attacks. Figures 6 and 7 visually summarizes the embedding and extraction schemes respectively, which will be discussed in detail in the following subsections.

A. KEY GENERATION

Secret keys are commonly used in symmetric encryption schemes to ensure the confidentiality of information. Generally, a secret key should be at least 128 bits long to withstand brute force attacks. Any algorithm using a secret key should be highly sensitive to any slight changes to its key bits [45]. In the proposed scheme, the secret key is generated from the host and watermark images. The key bits are then used to generate initial conditions and system parameters of the enhanced chaotic maps. The maps are then iterated to produce a matrix that will be used to transform the watermark. The key generation process is as follows:

1) The host image is transformed using IWT into four sub-bands, the first of which is an approximate image whereas the remaining three sub-bands contain other details, as shown in Figure 1.
2) The secret key is calculated based on the mean value of the three *detail* sub-bands and the transformed mean value of the watermark's histogram. This is calculated as

$$Key_{numbers} = \Big(mean(abs(LH)) + mean(abs(HL)) + mean(abs(HH)) + BinTrans(W)\Big) \times 10^{14} \quad (4)$$

where LH, HL and HH are *detail* sub-bands, W is the watermark and $abs(\cdot)$ is a function that returns positive values. $BinTrans(W)$, a transform function applied on the watermark, is defined mathematically as

$$BinTrans(W) = mean((Hist_{p_W} || p_W) \times 2^{14}) \quad \forall\, p_W \in [0, 255] \quad (5)$$

where $Hist_{p_W}$ is the frequency of a pixel, p_W in W. For example, if $Hist_{p_W} = 100$ and $p_W = 0$, their corresponding binary values are 1100100 and 0 respectively, which are then concatenated to become 11001000 (equivalent to 200 in decimal). Then, the resulting value is multiplied by 2^{14}. This process is repeated for all possible greyscale pixel values (ranging from 0 to 255), and the mean of all the resulting values is the output of $BinTrans$. The purpose of $BinTrans$ is to ensure that each pixel of the watermark will affect the resulting secret key. In other words, the pixels are indirectly diffused throughout the secret key. The factor of 10^{14} (which is the approximate limit of the IEEE 754 double floating point representation) was selected to magnify the values and increase the overall sensitivity of the scheme to small changes in the watermark and host images.
3) The resulting intermediary key, $Key_{numbers}$ is hashed by the hash function MD5 and the obtained result is 128-bits (Key_{bits}).
4) The initial conditions x_0 and y_0, and system parameters r_1 and r_2 are calculated based on Key_{bits} as

$$x_0 = \sum_{i=1}^{52} \frac{key_{bits}(i)}{2^i}, \quad (6)$$

$$y_0 = (\sum_{i=53}^{104} \frac{key_{bits}(i)}{2^{i-52}} + x_0)\ mod\ 1, \quad (7)$$

$$r_1 = (\sum_{i=1}^{8} key_{bits}(i) \times 2^i + \sum_{i=25}^{76} \frac{key_{bits}(i)}{2^{i-24}} + y_0)\ mod\ 10 + 10, \quad (8)$$

$$r_2 = (\sum_{i=121}^{128} key_{bits}(i) \times 2^{129-i} + \sum_{i=77}^{128} \frac{key_{bits}(i)}{2^{i-76}} r_1)\ mod\ 10 + 10, \quad (9)$$

$$x_0 = (x_0 + r_2)\ mod\ 1. \quad (10)$$

5) The initial values $\{x_0, r_1\}$ and $\{y_0, r_2\}$ are used for the enhanced logistic and sine map, respectively. The maps are iterated $M \times 256$ times, whereby the chaotic trajectories are stored as matrices X and Y of size $M \times 256$. Based on these intermediary matrices, a final chaotic matrix A is then calculated as

$$A = \sum_{i=1}^{M} \sum_{j=1}^{256} \Big((x(i,j) + y(i,j))\ mod\ 0.5\Big), \quad (11)$$

where $x(i,j)$ and $y(i,j)$ are the elements of the X and Y matrices respectively, while i and j denote the row and column of A.

The resulting matrix A consists of random numbers within the phase space of $[0, 0.5]$. Any changes to the secret

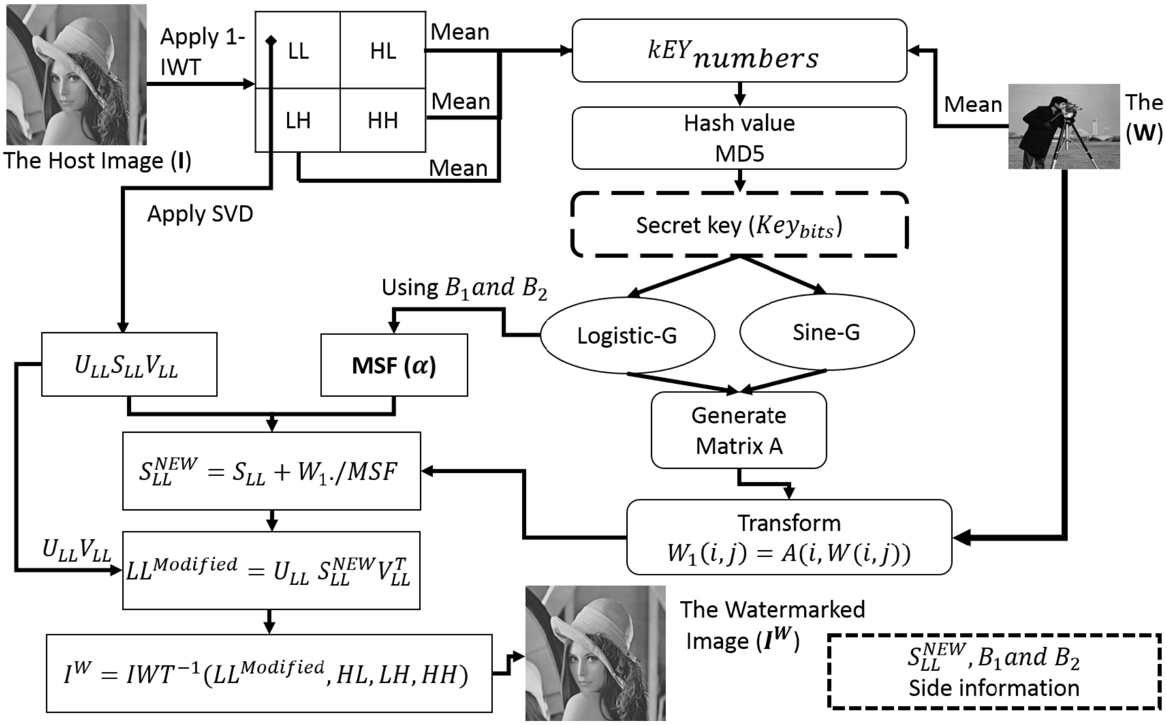

Apply 1-IWT
LL
HL
LH
HH
Mean
Mean
Mean
$kEY_{numbers}$
Hash value MD5
Secret key (Key_{bits})
Mean
The Watermark (**W**)
The Host Image (**I**)
Apply SVD
Using $B_1 and\ B_2$
Logistic-G
Sine-G
$U_{LL}S_{LL}V_{LL}$
MSF (α)
Generate Matrix A
$S_{LL}^{NEW} = S_{LL} + W_1./MSF$
Transform $W_1(i,j) = A(i, W(i,j))$
$U_{LL}V_{LL}$
$LL^{Modified} = U_{LL}\ S_{LL}^{NEW} V_{LL}^T$
$I^W = IWT^{-1}(LL^{Modified}, HL, LH, HH)$
The Watermarked Image (I^W)
$S_{LL}^{NEW}, B_1 and\ B_2$ Side information

FIGURE 6. The proposed scheme embedding process.

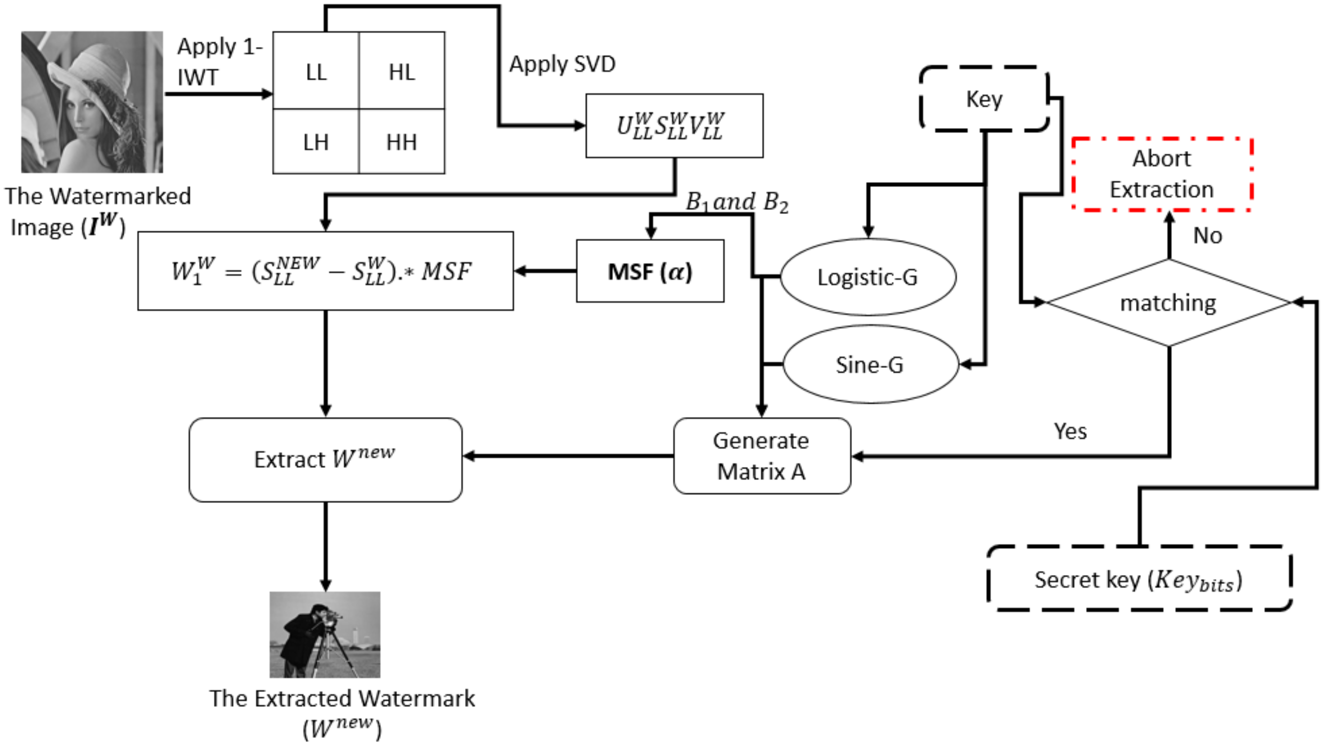

Apply 1-IWT
LL
HL
LH
HH
The Watermarked Image (I^W)
Apply SVD
$U_{LL}^W S_{LL}^W V_{LL}^W$
Key
Abort Extraction
$B_1 and\ B_2$
$W_1^W = (S_{LL}^{NEW} - S_{LL}^W).* MSF$
MSF (α)
Logistic-G
Sine-G
No
matching
Yes
Extract W^{new}
Generate Matrix A
Secret key (Key_{bits})
The Extracted Watermark (W^{new})

FIGURE 7. The proposed scheme extraction process.

key will lead to an entirely different matrix due to the initial condition sensitivity of the enhanced chaotic maps. The secret key itself is unique to each watermark and host image, whereby a small change in the watermark or the host image will generate an entirely new secret key. Thus, the secret key should be saved in a secure database by a trusted third party that deals with ownership protection.

B. WATERMARK EMBEDDING

The steps of the proposed embedding process are as follows:

1) Select the LL sub-band of host image and apply SVD to it

$$SVD(LL) = U_{LL} * S_{LL} * V_{LL}^{T} \tag{12}$$

2) Transform the watermark, W into a new matrix W_1 using the chaotic matrix, A. The transformation process is defined as

$$W_1(i,j) = A(i, W(i,j)), \quad i = 1, 2, 3 \ldots M \quad and\ j = 1, 2, 3, \ldots N. \tag{13}$$

3) The values from the matrix W_1 is embedded into S_{LL}. The process can be mathematically defined as

$$S_{LL}^{new} = S_{LL} + (W_1) \cdot \alpha \tag{14}$$

where $\cdot$ is the dot product operation and α is the scaling factor. As mentioned in Section I, there are two types of scaling factors, SSF and MSF. In SSF, α is the only value used in watermarking scheme. When α is small, the imperceptibility between the watermarked image and the host image will be high but the scheme will lack robustness. On the other hand, MSF has multiple scaling values which can provide a good balance between imperceptibility and robust. Most existing schemes use optimization algorithms to select optimal MSFs, which can achieve good trade-offs between these key metrics [13], [15], [25]. In this paper, the logistic-G map is used to generate new chaotic scaling factors without the use of optimization algorithms. Chaotic points that are generated from the logistic-G map are employed to generate α of size W. The CMSF consists of scaling factor elements that are suitable for the proposed scheme, and requires less computational effort to generate as compared to optimization algorithms. It is calculated as

$$\alpha = fix((x(i,j) \cdot 10^{10})\ mod\ \beta_1 + \beta_2) \tag{15}$$

$$\alpha = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,N} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{M,1} & \alpha_{M,2} & \cdots & \alpha_{M,N} \end{bmatrix} \tag{16}$$

where α may be one value or multiple values between the bounded interval of $[\beta_1, \beta_2]$, and x represents the chaotic points of the enhanced logistic map. Function *fix* returns an integer value. For example, if $\beta_1 = 1$ and $\beta_2 = 5$, then α is a SSF equal to 5. If $\beta_1 = 50$ and $\beta_2 = 50$, then α represents CMSF ranging from [50, 100]. Thus, CMSF is generated randomly based on chaotic points and the bounded interval. The chaotic points are essentially random numbers which can lead to inconsistent interval ranges if left unbounded. Thus, the bounded interval is imposed to control the CMSF values. This allows a more consistent evaluation of the proposed scheme's performance. By allowing the bounding interval variables β_1, β_2 to be specified, the CMSF values are more flexible and random.

4) The modified LL sub-band is obtained through inverse SVD as

$$LL^{modified} = U_{LL} * S_{LL}^{new} * U_{LL}^{T} \tag{17}$$

5) The inverse IWT is applied by using $LL^{modified}$ and the remaining *detail* sub-bands (LH, HL, HH) to obtain the watermarked image, I^W.

6) β_1,β_2, and S_{LL}^{new} are used as the side information extraction in addition to key_{bits} as the secret key.

C. WATERMARK EXTRACTION

The extraction process starts off by extracting W_1' from the watermarked image. Then, W_1' is transformed into the watermark, W. The extraction steps are as follows:

1) Apply one-level of IWT on the watermarked image I^W (possibly distorted due to an attack) to obtain the four sub-bands, LL^{I^W}, LH, HL and HH.

2) Further decompose LL^{I^W} by SVD

$$SVD(LL^{I^W}) = U_{LL^{I^W}} * S_{LL^{I^W}} * V_{LL^{I^W}} \tag{18}$$

3) Obtain W_1^W by computing

$$W_1^W = (S_{LL}^{new} - S_{LL^{I^W}}) \cdot \alpha \tag{19}$$

where α is computed based on the secret key, Key_{bits} and β values from Eq. 15, and S_{LL}^{new} is obtained from the side information.

Before transforming W_1^W into the watermark, W^{new}, the secret key of the claimant (individual claiming ownership to the host image) will be compared against the secret key stored by the trusted third party. If the two keys match, the transformation process will be allowed to continue. Otherwise, the process will be halted. Thus, adversaries with forged secret keys will not be able to successfully prove ownership of the host image. If the matching process is successful, the transformation process proceeds as follows:

1) Regenerate the matrix A by using Eq.11.

2) Extract the watermark W^{new} by computing the absolute difference between each row of the two matrices W_1^W and A as

$$W^{new}(i,j) = \begin{cases} k-1 & if\ abs\Big(W_1^W(i,j) - A(i,k)\Big) = 0, \\ W^{new}(i-1,j) & Otherwise, \end{cases} \tag{20}$$

where $i = \{1, 2, \ldots, M\}$ and $j = k = 1, 2, .., N$. W^{new} is the extracted watermark image which accounts for any distortions to the watermarked host image. For i and j values, k is iterated from 1 to N. When the condition is fulfilled, we select $k - 1$. If the condition is not fulfilled, select the previous value from the same

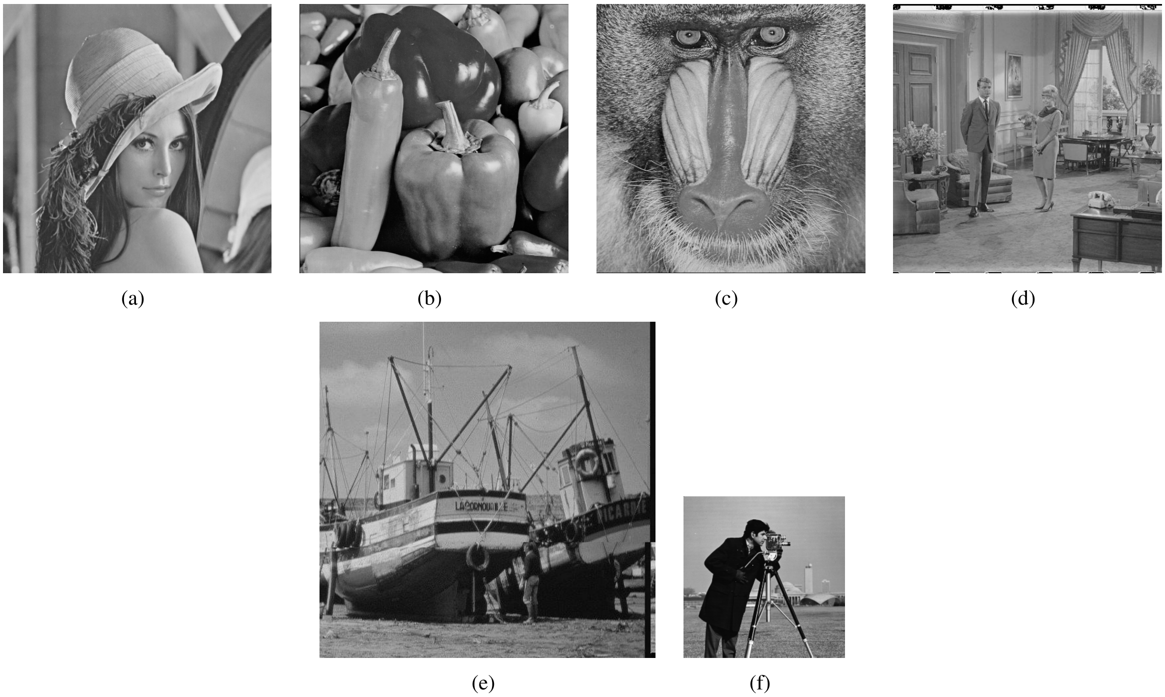

(a) (b) (c) (d)
(e) (f)

FIGURE 8. Host images (a) Lena, (b) Peppers, (c) Baboon, (d) Couple, (e) Boat, (f) the watermark image, Cameraman.

watermark. If and only if i not equal 1, select a value of zero because zero is not within the width of image. So when there is any change in the key, the extracted watermark will consist of all zeroes (black image).

D. KEY MANAGEMENT

For copyright verification, the secret key must be registered and saved with a third party known as a certified authority (CA) [22], [46]–[49]. This will circumvent attacks whereby adversaries embed their own watermark and generate their respective secret keys. Without the CA, any adversary can claim ownership of the watermarked image. To solve this issue, the CA will play a role in linking each image to its side information, secret key or its ownership share in zero-watermarking schemes [22], [47].

E. DISCUSSION

In the proposed scheme, IWT, SVD, and chaotic maps are employed to achieve high imperceptibility and robustness. The proposed scheme has a number of advantages which includes the following:

- The secret key, Key_{bits} is generated from the host and watermark images to increase security of the proposed scheme. Thus, it is impossible to extract the watermark without the secret key.
- To protect the watermark image from distortions and attacks, the watermark image is not embedded directly into the host image. Instead, it is transformed using a chaotic matrix prior to embedding. The chaotic matrix consists of elements that are randomized.
- The chaotic scaling factor α is generated by using the chaotic trajectory of the enhanced logistic map and the bounded interval variables. α can achieve a trade-off between imperceptibility and robustness without the need for optimization algorithms.
- The matrix W_1 is employed in the embedding process to increase the payload capacity of the proposed scheme.
- FFP issues are addressed by the secret key matching process during the extraction phase.
- The proposed scheme can embed a watermark image with a dimension of $(\frac{M}{4} \times \frac{N}{4})$. In the case of smaller watermarks, the watermark image pixels can be duplicated prior to the embedding process and be embedded into other sub-bands for increased robustness and capacity.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

For all experiments, five grayscale images Lena, Peppers, Baboon, Couple and Boat of size 512 are used as the host images, whereas the 256 × 256 Cameraman image is used as the watermark image. These images are as shown in Figure 8. The proposed scheme is simulated using MATLAB R2012b on a 32-bit on processor and 4 GB RAM. The proposed scheme uses the IWT 'Haar' wavelet to transform the host image into four sub-bands. Although computing the 'Haar' wavelet is computationally expensive, it leads to improved image resolution and adopts integers without round-off errors. The transformed matrix of the watermark image is directly embedded into the singular values of the approximation sub-band only after carrying out the IWT

TABLE 1. Imperceptibility (PSNR) results for different CMSF values.

Test image	CMSF				
	$[1-10^2]$	$[1-10^3]$	$[1-10^4]$	$[1-10^5]$	$[1-10^6]$
Lena	52.22	52.51	52.58	52.22	52.21
Peppers	52.34	52.51	52.22	52.25	52.37
Baboon	52.36	52.21	52.31	52.36	52.42
Couple	52.44	52.42	52.51	52.53	52.48
Boat	52.42	52.42	52.43	52.40	52.43

TABLE 2. Robustness (NC) results for various CMSF without attacks.

Test image	CMSF				
	$[1-10^2]$	$[1-10^3]$	$[1-10^4]$	$[1-10^5]$	$[1-10^6]$
Lena	0.99521	0.99484	0.99469	0.99424	0.99413
Peppers	0.99455	0.99542	0.99412	0.99388	0.99401
Baboon	0.99544	0.99399	0.99402	0.99320	0.99384
Couple	0.99517	0.99515	0.99442	0.99377	0.99412

transform on the host image. As previously mentioned, the CMSFs are computed based on β_1,β_2, and the enhanced logistic map (Eq. (15)).

A. IMPERCEPTIBILITY AND ROBUSTNESS ANALYSIS

Peak-signal-to-noise ratio (PSNR) and normalized correlation (NC) are the two main tests to assess imperceptibility and robustness of a watermarking scheme, respectively. PSNR can be defined as

$$PSNR = 10\log_{10}\left[\frac{max(I(i,j))^2}{MSE}\right] \tag{21}$$

where $max(I(i,j))$ is the largest pixel value in the host image, while the mean square error (MSE) between the host image I and the watermarked image I^W can be calculated as

$$MSE = \frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}[I(i,j)-I^W(i,j)]^2 \tag{22}$$

where M and N are the number of rows and columns of an image. A high PSNR value is desired as it indicates that there is minimal difference between the host image, I and watermarked image, I^W. On the other hand, NC is a measure of the difference between an extracted watermark W^{new} and the original watermark W. NC is calculated as

$$NC(W,W^{new}) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[W(i,j)-\mu_1][W^{new}(i,j)-\mu_2]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[W(i,j)-\mu_1]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[W^{new}(i,j)-\mu_2]^2}} \tag{23}$$

where μ_1 and μ_2 denote the mean values of W and W^{new}, respectively. When the original and extracted watermark image closely resemble one another, $NC \approx 1$. In case where $NC = 1$, the original and extracted watermarks are identical.

For the proposed scheme, the PSNR results for the five host images based on various CMSF ranges are tabulated in Table 1. In other existing schemes, small SSF can achieve high PSNR values but are subpar in terms of robustness against geometrical and non-geometrical attacks. In the proposed scheme, this problem is circumvented as shown in Table 1, whereby both small and large scaling factors are able to achieve high PSNR. Next, Table 2 presents the NC results of the proposed scheme using different CMSF ranges. In these experiments, five CMSF intervals are used to study the performance of the proposed scheme under various intervals, thus depicting its flexibility. The proposed scheme can successfully extract the watermark regardless of whether the CMSF values are small or large, with NC results of approximately 1.

Table 3 and 4 show the NC values of the proposed scheme for the Lena and Peppers images in comparison with another existing scheme, using various the CMSF ranges. 15 different geometrical and non-geometrical modifications were applied on the watermarked image, then the watermarks are extracted. The proposed scheme can successfully extract the watermark in all situations regardless of the CMSF range. The robustness of the proposed scheme is near-ideal. When the CMSF range is small, the proposed scheme achieves high imperceptibility and high robustness. On the other hand when the CMSF range is large, the proposed scheme also has high imperceptibility and high robustness against the several well-known attacks. In addition to achieving better results in PSNR and NC, the proposed method to generate CMSF values is also more efficient than optimization algorithms. Figure 9 shows the watermarked and the extracted watermark images under the various attacks, whereby the results indicate that the proposed scheme can extract the watermark with minimal distortions.

The PSNR imperceptibility values of the proposed scheme and other exiting watermarking schemes for different host images are compared in Table 5. One can observe that the proposed scheme achieves high imperceptibility as compared to the other existing schemes for all of images [13], [39]. Furthermore, the NC robustness measure of the proposed scheme and other existing schemes [18], [19], [50], [51] under several well-known attacks are listed in Table 6. Results indicate that the proposed scheme outperforms the other schemes, and can successfully extract the watermark without distortions. This is due to the second portion of the extraction process which can remove distortions to recover the correct pixels.

B. FPP ANALYSIS

The proposed scheme is suitable for copyright ownership protection of digital images because it overcomes FPP issues by performing secret key matching to confirm owner legitimacy. In addition to side information such as β_1, β_2 and S_{LL}^{new}, secret key Key_{bits} is also used to embed and extract the watermark. Most existing image watermarking schemes only rely on side information generated from the embedding process as the secret key, which can be easily attacked and modified.

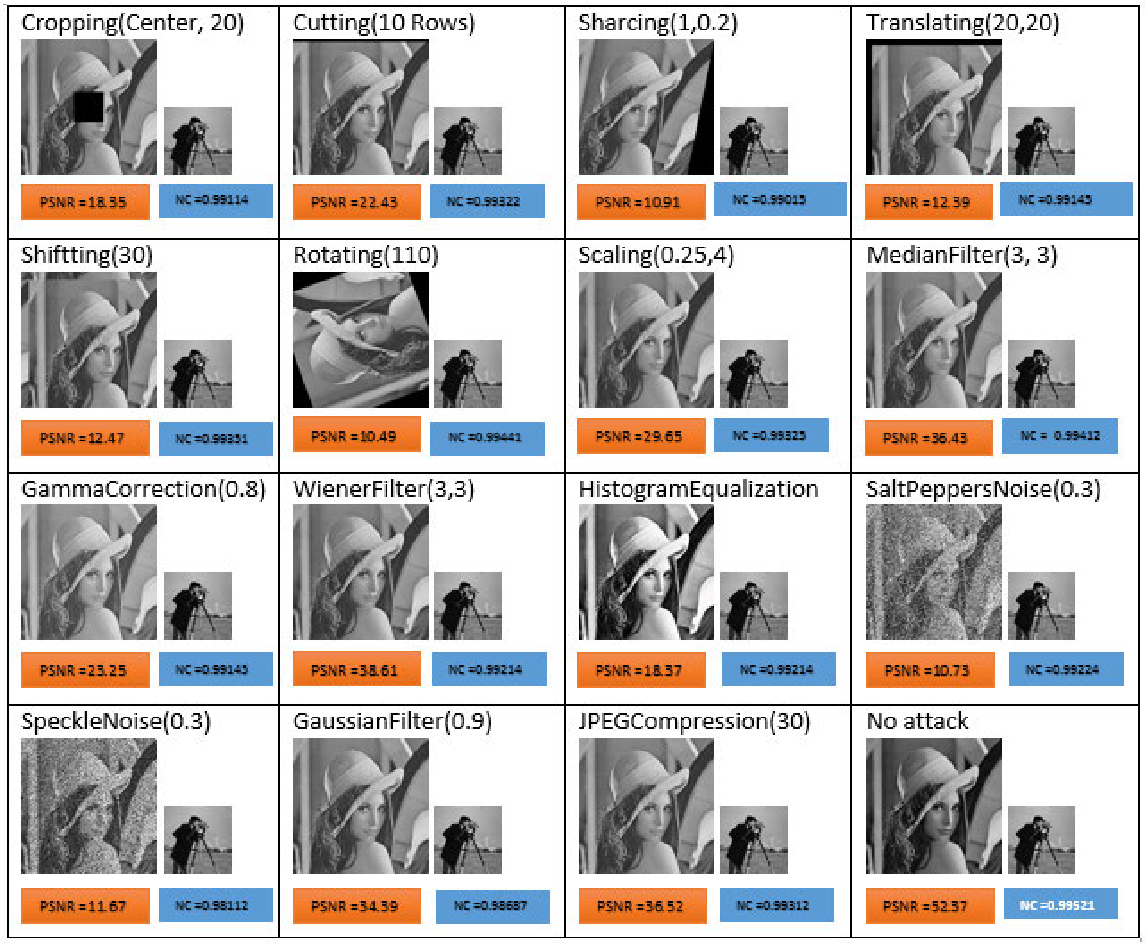


FIGURE 9. Watermarked image and extracted watermark against different attacks.

TABLE 3. Robustness (NC) results for different CMSF under different attacks on the Lena image.

Attacks	CMSF					Ref. [13]
	$[1-10^2]$	$[1-10^3]$	$[1-10^4]$	$[1-10^5]$	$[1-10^6]$	
Cropping (Center, 20)	0.99114	0.98241	0.98245	0.99112	0.99412	0.9200
Cutting (10 rows)	0.99322	0.99345	0.99152	0.99451	0.99014	0.9822
Shareing (1, 0.2)	0.99015	0.99176	0.99122	0.99202	0.99245	0.9018
Translating (20, 20)	0.99145	0.99045	0.99214	0.99451	0.99174	0.9380
Shiftting (30)	0.99351	0.99245	0.99331	0.99074	0.99151	0.9907
Rotating (110)	0.99441	0.99342	0.99145	0.99099	0.99435	0.9479
Scaling (0.25, 4)	0.99325	0.99188	0.99254	0.99425	0.99421	0.9680
Median filter (3, 3)	0.99412	0.99425	0.99421	0.99324	0.99225	0.9974
Gamma Correction (0.8)	0.99145	0.99421	0.99224	0.99422	0.99124	0.9939
Wiener Filter (3, 3)	0.99214	0.99214	0.99421	0.99124	0.99114	0.9901
Histogram Equalization	0.99214	0.99112	0.99324	0.99244	0.99422	0.9311
Salt Peppers Noise (0.3)	0.99224	0.99352	0.99289	0.99441	0.99388	0.9353
Speckle Noise (0.3)	0.98112	0.98214	0.99101	0.98658	0.98881	0.9152
Gaussian Filter (0.3)	0.98687	0.98578	0.98789	0.98875	0.98584	0.9003
JPEG Compression (30)	0.99312	0.99545	0.99254	0.99321	0.99124	0.9930

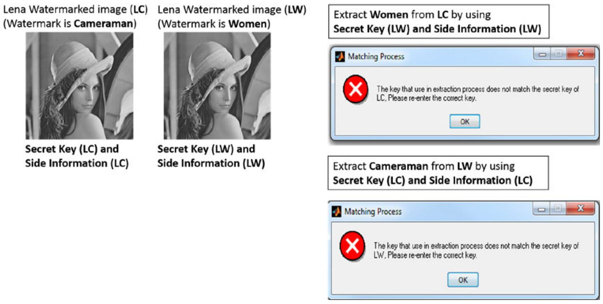
Furthermore, the side information is not sensitive to slight changes, a property which can be exploited by adversaries when falsely claiming ownership of the host image. In the proposed scheme, the secret key is extracted from the host and watermark images. Therefore, changes to either one of them will lead to a large change in the secret key. We now analyze the robustness of the proposed scheme based on the three FPP scenarios:

TABLE 4. Robustness (NC) results for different CMSF values under different attacks on the Peppers image.

Attacks	CMSF					Ref. [13]
	$[1-10^2]$	$[1-10^3]$	$[1-10^4]$	$[1-10^5]$	$[1-10^6]$	
Cropping (Center, 20)	0.99021	0.99124	0.99101	0.98104	0.99087	0.9340
Cutting (10 rows)	0.99325	0.99265	0.99245	0.99348	0.99149	0.9757
Shareing (1, 0.2)	0.99398	0.99365	0.99185	0.99165	0.99011	0.9130
Translating (20, 20)	0.99214	0.99224	0.99511	0.99231	0.99285	0.9470
Shiftting (30)	0.99222	0.99332	0.99342	0.99112	0.99101	0.9910
Rotating (110)	0.99354	0.99212	0.99340	0.99265	0.99254	0.9507
Scaling (0.25, 4)	0.99123	0.99254	0.99284	0.99254	0.99131	0.9720
Median filter (3, 3)	0.99254	0.99218	0.99241	0.99153	0.99241	0.9969
Gamma Correction (0.8)	0.99211	0.99315	0.99335	0.99121	0.99245	0.9956
Wiener Filter (3, 3)	0.99384	0.99254	0.99221	0.99245	0.99224	0.9892
Histogram Equalization	0.99021	0.99121	0.99312	0.99124	0.99412	0.9405
Salt Peppers Noise (0.3)	0.99211	0.99211	0.98122	0.98123	0.98258	0.9433
Speckle Noise (0.3)	0.98012	0.98212	0.98128	0.98136	0.98246	0.9277
Gaussian Filter (0.3)	0.98214	0.98298	0.98754	0.98949	0.98942	0.9878
Jpeg Compression (30)	0.99128	0.99255	0.99258	0.99354	0.99393	0.9965

TABLE 5. Imperceptibility comparison between various schemes.

Test Image	Proposed scheme	Ref. [13]	Ref. [39]
Lena	**52.21**	42.9245	39.56
Peppers	**52.37**	42.9477	39.98
Baboon	**52.42**	42.9159	39.31
Couple	**52.48**	42.9322	39.63
Boat	**52.43**	42.9381	39.31



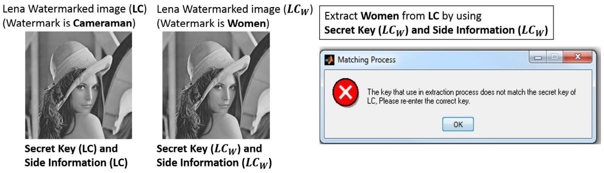
Extract **Women** from **LC** by using
Secret Key (LW) and Side Information (LW)

The key that use in extraction process does not match the secret key of LC, Please re-enter the correct key.

Extract **Cameraman** from **LW** by using
Secret Key (LC) and Side Information (LC)

The key that use in extraction process does not match the secret key of LW, Please re-enter the correct key.

FIGURE 10. Resistance against FPP attack 1.


Lena Watermarked image (LC)
(Watermark is **Cameraman**)
Lena Watermarked image (LC_W)
(Watermark is **Women**)
Secret Key (LC) and
Side Information (LC)
Secret Key (LC_W) and
Side Information (LC_W)
Matching Process
OK

Extract **Women** from **LC** by using
Secret Key (LC_W) and Side Information (LC_W)

The key that use in extraction process does not match the secret key of LC, Please re-enter the correct key.

FIGURE 11. Resistance against FPP attack 2.

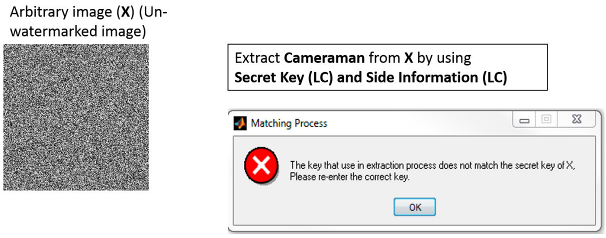
- **FPP Scenario 1:** Let the Lena image (denoted as L) be the host image and the Cameraman (C) and Woman (W) images be watermarks. The Cameraman watermark is embedded into the Lena image (version 1 of Lena) to produce the watermarked image, LC. Next, the Woman watermark is embedded in another Lena image (version 2 of Lena) separately, resulting in another watermarked image, LW. The secret key for each watermarking processes are generated, along with the corresponding side information. When the secret key and side information of LW are used to extract the W of LC, or vice versa, the secret key mismatch will halt the extraction process. as shown in Fig. 10.
- **FPP Scenario 2:** Let L be the host image whereas C is the legitimate owner's watermark image. The owner embeds C into L to obtain the watermarked image LC, the secret key and side information. On the attacker's end, LC is used as the host image to embed another watermark W to produce a new watermarked image LC_W, another secret key and a new set of side information. The attacker then tries to extract the forged watermark, W from LC using his secret key and the side information from LC_W. Because the attacker's secret key does not match the key that corresponds to LC, the extraction process is halted.
- **FPP Scenario 3:** Let L be the host image whereas C is the legitimate owner's watermark image. The owner embeds C into L to obtain the watermarked image LC, the secret key and side information. In this scenario, an attacker uses an arbitrary image X as the host image and attempts to extract C from X by using the side information of LC. If the attacker succeeds, he or she can claim ownership of X without embedding any watermark into X. However without the secret key, the attacker will not be able to extract C even with the side information of LC.

C. SECRET KEY SENSITIVITY

To overcome FPP issues and withstand attacks, the proposed scheme needs to be highly sensitive to slight changes in the secret key. A small change in the secret key should lead to an entirely different watermarked image. In addition, an incorrect secret key should not be able to extract the original watermark. To analyze the proposed scheme's sensitivity to

TABLE 6. Robustness (NC) comparison under different attacks between various schemes.

Attacks	Ref. [50]	Ref. [18]	Ref. [19]	Ref. [51]	Proposed sechme
Histogram Equalization	0.9934	0.9849	0.9664	**0.9982**	0.99214
Gaussian Filter	0.9849	0.9244	0.9358	0.9567	**0.98687**
JPEG Compression	**0.9991**	0.9954	0.9875	0.9942	0.99312
Gamma Correction	**0.9981**	0.9952	0.9585	0.9948	0.99145
Median filter	0.9932	0.9894	0.9458	0.9903	**0.99412**
Cropping	0.9907	0.9592	0.5789	0.9878	**0.99114**
Shiftting	0.9934	0.9899	0.5435	0.9847	**0.99351**



Extract **Cameraman** from **X** by using
Secret Key (LC) and Side Information (LC)

Matching Process

The key that use in extraction process does not match the secret key of X, Please re-enter the correct key.

FIGURE 12. Resistance against FPP attack 3.

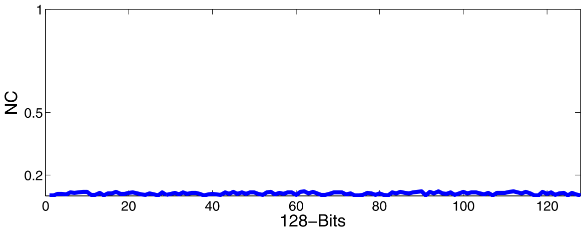


FIGURE 13. Secret key sensitivity of the proposed scheme.

FIGURE 14. Extracted black watermark due to incorrect secret key.

the secret key, we ignore the secret key matching portion of the proposed scheme. Let K_W be the secret key generated when embedding a watermark, W into a host image I to produce a watermarked image I_W. We toggle one bit of K_W to obtain a forged key, $\hat{K_W}$. We use K_W and $\hat{K_W}$ to extract W from I_W. The NC values are calculated for the resulting watermarks. We repeat the experiment 128 times, whereby we toggle each of the 128 bit positions of K_W. The resulting NC values are as shown in Figure13, whereas Figure 14 shows a watermarked image during the extraction process when one bit of the secret key has been changed. Results show that even a one-bit change will generate black images with low NC values.

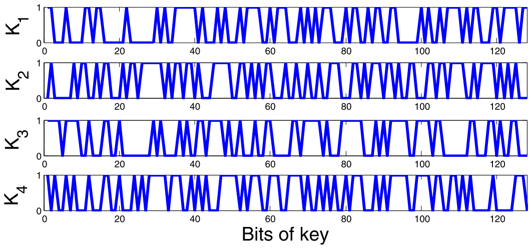


FIGURE 15. Key bits after changing one bit of the watermark image (four cases).

Next, we analyze how a small change in the watermark pixels lead to entirely new secret keys. The MD5 hash function is highly sensitive to tiny changes to its input message. Moreover, the enhanced chaotic maps are very sensitive to small change to their chaotic variables as shown in Figure 4. Hence, the enhanced chaotic maps will generate new overall chaotic points when small change to chaotic variables. We toggle a single bit of the Cameraman watermark image in different positions, then its corresponding secret key (hash value of MD5) is generated. The secret keys can be represented as binary streams to visually compare differences with other keys. We denote the watermark images as W_i whereas their corresponding secret keys are K_i, where i is number of changes that have been performed. Figure 15 shows the binary differences between between the secret keys. We can see that a difference of a single bit leads to an overall change to the secret key.

D. NPCR AND UACI TESTS

In this section, we evaluate the randomness of the watermark image that has been converted to a random matrix (essentially encrypted) by using chaotic maps and the secret key. To evaluate its randomness by using NPCR and UACI, the random matrix is converted to unsigned 8-bit integers. The 8-bit conversion is performed by computing

$$W_B = (W_1 \times 2^{14})\ mod\ 256 \tag{24}$$

where W_1 denotes the random matrix. w_B is an image consisting of 8-bit grayscale pixels. We change one-bit of the original watermark W to generate a new W_B, then NPCR and UACI tests can be estimated as

$$NPCR({W_1}^1, {W_1}^2) = \frac{\sum_{i,j}^{M,N} D(i,j)}{MN} \times 100\% \tag{25}$$

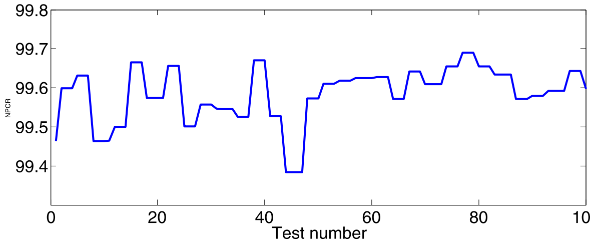


FIGURE 16. **NPCR results for 100 modified Cameraman watermarks.**

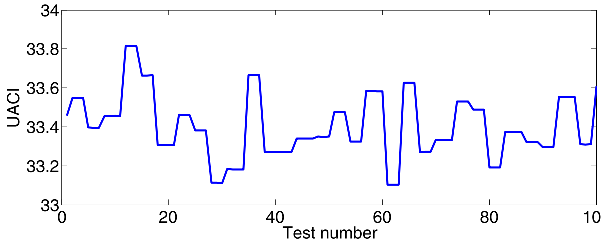


FIGURE 17. **UACI results for 100 modified Cameraman watermarks.**

TABLE 7. **PSNR and NC values of different size of the host images and watermark images.**

Host Image	Watermark Image			

128 × 128	32 × 32
	PSNR= 52.33
	NC=0.99411
		65 × 32
		PSNR= 50.45
		NC=0.99221
			32 × 65
			PSNR= 49.31
			NC=0.99512
				65 × 65
				PSNR= 47.74
				NC=0.99391

320 × 320	32 × 32
	PSNR= 50.12
	NC=0.99101
		90 × 60
		PSNR= 49.28
		NC=0.99201
			72 × 90
			PSNR= 48.78
			NC=0.99513
				120 × 120
				PSNR= 46.85
				NC=0.99372

and

$$UACI(W_1{}^1, W_1{}^2)=\frac{1}{MN}\sum_{i,j}^{M,N}\frac{|W_1{}^1(i,j)-W_1{}^2(i,j)|}{L}\times 100\% \tag{26}$$

respectively, where MN is the total number of pixels in a watermark, $L = 255$ is the maximum gray level value for an 8-bit pixel, and

$$D(i,j)=\begin{cases}1, & \text{if } W_1{}^1(i,j)\neq W_1{}^2(i,j)\\ 0, & \text{if } W_1{}^1(i,j)=W_1{}^2(i,j),\end{cases} \tag{27}$$

Figure 16 and 17 show the NPCR and UACI tests respectively, for 100 randomly modified Cameraman watermark images. We can observe that the proposed scheme can generate entirely different random images and the test results are close to ideal ($NPCR \geq 99.5693$ and $UACI \in [33.2824, 33.6447]$ [41], [52]). This shows that the proposed scheme is highly sensitive to small changes to the watermark image.

E. FLEXIBILITY

This section depicts the flexibility of the proposed scheme when embedding watermark images of varying sizes into different sub-bands. In this experiment, we use host and watermark images of different sizes. We embed into all sub-bands to maximize capacity. For situations whereby a watermark is of the same size as the host image, it is divided into four subsections and each subsection is embedded into one sub-band. For situations where the watermark is smaller than the host image and is not equal to the size of the sub-bands, some row and column pixels are duplicated so that the watermark can be divided into subsections that are of the same size as the sub-bands. Algorithm 22 describes the watermark division process.

Algorithm 1 Watermark Division

```
Data: Input watermark $W$, where $M_W$ and $N_W$ are
      height and width. Input host image $I$, where $M_S$
      and $N_S$ are height and width of sub-band
Result: $W_k$, where $k$ denotes number of subsections
if $M_W == M_S$ $N_W == N_S$ then
    Embed the watermark $W$ into sub-band $LL$;
else
    if $M_W == M_S$ $N_W < N_S$ then
        Duplicate the last pixel columns $(N_S - N_W)$ ;
        Embed the watermark $W$ into sub-band $LL$;
    else
        if $M_W == M_S$ $N_W < 2N_S$ then
            Duplicate the last pixel columns $(2N_S - N_W)$
            ;
            Embed the watermark $W$ into sub-bands $LL$
            and $LH$;
        else
            if $M_W < M_S$ $N_W == N_S$ then
                Duplicate the last pixel rows $(M_S - M_W)$
                ;
                Embed the watermark $W$ into sub-bands
                $LL$;
            else
                if $M_W < 2M_S$ $N_W == N_S$ then
                    Duplicate the last pixel rows
                    $(2M_S - M_W)$ ;
                    Embed the watermark $W$ into
                    sub-bands $LL$ and $HL$;
                else
                    Duplicate the last pixel rows
                    $(2M_S - M_W)$ ;
                    Duplicate the last pixel columns
                    $(2N_S - N_W)$ ;
                    Embed the watermark $W$ into all
                    sub-bands ;
```

Table 7 shows the PSNR and NC values of the different host images and watermark images. We calculate the NC values after removing duplicated pixels to obtain accurate results. The proposed scheme is able to embed watermark images of different sizes into the host image in one or more sub-bands. The results indicate that the proposed scheme

TABLE 8. Comparative analysis of existing SVD-based watermark schemes.

Reference	Embedding Sub-Bands	Type of Trasforms	Watermark Size	Scalling Factor	Solve FPP	Embedding Procedure
Proposed Scheme	LL or All	IWT+SVD	Equal to or smaller than host image	MSF (Chaotic)	Yes	Embed the watermark image after transformed into another chaotic matrix into S_{LL} of host and generating a secret key that is unique to extract.
Ref. [17]	All	RT+DWT+SVD	33×33	SSF	No	S_W was embedded into S_H
Ref. [51]	ALL	DWT+SVD	256×256	MSF (PSO)	Yes	PCs of the watermark were embedded into the host image
Ref. [54]	LH_3	DWT+SVD	32×32	MSF (MOACO)	Yes	S_W was embedded into S_H of the host image, U_W and V_W were hashed and stored as private key
Ref. [36]	HH	DWT+SVD	256×256	-	Yes	S_W was replaced of S_H and U_W and V_W were authenticated before extraction
Ref. [18]	LL_3 and HH_3	DWT+SVD+HVS	64×64	(MSF) DE	Yes	The gray-scale watermark was embedded into S_H of LL_3 and HH_3. A Secret key was generated by Xoring two binary images extracted of LL_3 and watermark
Ref. [55]	All	DWT+SVD	256×256	(MSF) SDE	Yes	Extract PCs of sub-bands of watermark after applied DWT and SVD of each sub-band, and alter S_H of host image after applied DWT and SVD
Ref. [53]	All	RSWT +SVD	512×512	SSF	No	Modify S_H of all sub-bands by S_W of watermark image

has high imperceptibility and high robustness even when embedding watermarks of the same size as the host image. Thus, the proposed scheme has the flexibility to accommodate different image sizes as well as the capability to embed into other sub-bands.

F. COMPARATIVE ANALYSIS

SVD-based watermarking schemes have two ways to embed a watermark image into the host image after it has been decomposed into the three matrices, U, V and S. The first method directly embeds watermark information into the singular values of S based on the scaling factors. This method is susceptible to FPP because the singular values hold minimal structural information of the host image. A large change in these values has minimal effect on the host image. The second method decomposes the watermark using SVD then embeds the singular value of the watermark image into the singular value of the host image based on the scaling factors. Although the proposed scheme is based on the first method, it overcomes FPP through the use of chaotic maps and the secret key, leading to a high imperceptibility and robustness.

SVD-based image watermarking schemes are able to embed the watermark information with high capacity [53]. However, schemes based on embedding one of the singular vectors or principal component are not robust against some well-known attacks such as image manipulations and geometric attacks [13], [37], [38]. This is due to the singular vectors which contain a large amount of structural information of watermark, whereby a small change to the left or right of the singular vectors will have a significant effect on the resulting watermark being extracted.

Table 8 shows a comparison of the proposed scheme against several existing SVD-based image watermarking schemes. The proposed scheme shows higher robustness as compared to its peers. The proposed scheme is able to circumvent FPP and achieve high NC values. It also has a larger embedding capacity than some of the other schemes. Due to the MSF generated by the chaotic points, the proposed scheme achieves a good trade-off between PSNR and NC values.

V. CONCLUSION

A new chaos-based SVD image watermarking scheme in the frequency domain was proposed in this paper. The new scheme introduced the use of a secret key, generated from the host and watermark images, as a central component in the embedding, extraction, and ownership verification processes. Initial conditions and control parameters of two chaotic maps were generated based on the secret key. The chaotic map trajectories were then used to transform the watermark into a chaotic matrix which is then embedded into the host image. In addition, MSF values were generated by the chaotic maps and secret key to achieve high robustness and imperceptibility. The host image was first transformed by a one-level IWT then the LL sub-bands were selected. Next, SVD was applied on LL, and S_{LL} was altered by the CMSF and chaotic matrix during the embedding process. The extraction process involves an additional key matching phase, whereby if the secret key is valid, the extraction process will continue to produce an extracted watermark that is identical to the original watermark. Otherwise, a black image will be produced. Thus, the proposed scheme is able to circumvent FPP, leading to improved security. In addition, the proposed scheme can accommodate different watermark sizes by embedding them into different sub-bands for higher flexible and larger capacity. However, the proposed scheme generates side information during the embedding phase which is used in the extraction process. This side information along with the secret key must be kept confidential. Our future work is to overcome this drawback, whereby the embedding phase will only rely on the secret key without side information. Other techniques such as cellular automation, blockchain or deep learning can also be explored to achieve blind or more secure watermarking systems.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] C. Kumar, A. K. Singh, and P. Kumar, "A recent survey on image watermarking techniques and its application in e-governance," *Multimedia Tools Appl.*, vol. 77, no. 3, pp. 3597–3622, Sep. 2017.

[2] J.-S. Pan, H.-C. Huang, and L. C. Jain, *Intelligent watermarking Techniques*, vol. 7. Singapore: World Scientific, 2004.

[3] H.-C. Huang, *Information Hiding Application*. Berlin, Germany: Springer, 2009, vol. 227.

[4] S. N. Bal, M. R. Nayak, and S. K. Sarkar, "On the implementation of a secured watermarking mechanism based on cryptography and bit pairs matching," *J. King Saud Univ.-Comput. Inf. Sci.*, to be published.

[5] F. Ernawan and M. N. Kabir, "An improved watermarking technique for copyright protection based on tchebichef moments," *IEEE Access*, vol. 7, pp. 151985–152003, 2019.

[6] 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.

[7] J. C. Patra, J. E. Phua, and C. Bornand, "A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression," *Digit. Signal Process.*, vol. 20, no. 6, pp. 1597–1611, Dec. 2010.

[8] P. Premaratne and C. Ko, "A novel watermark embedding and detection scheme for images in DFT domain," in *Proc. Image Process. Its Appl., 7th Int. Conf.*, Jul. 1999, pp. 780–783.

[9] E. Najafi, "A robust embedding and blind extraction of image watermarking based on discrete wavelet transform," *Math. Sci.*, vol. 11, no. 4, pp. 307–318, Aug. 2017.

[10] N. M. Makbol and B. E. Khoo, "Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition," *AEU Int. J. Electron. Commun.*, vol. 67, no. 2, pp. 102–112, Feb. 2013.

[11] A. Anand and A. K. Singh, "An improved DWT-SVD domain watermarking for medical information security," *Comput. Commun.*, vol. 152, pp. 72–80, Feb. 2020.

[12] N. A. Loan, N. N. Hurrah, S. A. Parah, J. W. Lee, J. A. Sheikh, and G. M. Bhat, "Secure and robust digital image watermarking using coefficient differencing and chaotic encryption," *IEEE Access*, vol. 6, pp. 19876–19897, 2018.

[13] N. M. Makbol, B. E. Khoo, T. H. Rassem, and K. Loukhaoukha, "A new reliable optimized image watermarking scheme based on the integer wavelet transform and singular value decomposition for copyright protection," *Inf. Sci.*, vol. 417, pp. 381–400, Nov. 2017.

[14] Priyanka and S. Maheshkar, "Region-based hybrid medical image watermarking for secure telemedicine applications," *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 3617–3647, Sep. 2016, doi: 10.1007/s11042-016-3913-1.

[15] N. R. Zhou, A. W. Luo, and W. P. Zou, "Secure and robust watermark scheme based on multiple transforms and particle swarm optimization algorithm," *Multimedia Tools Appl.*, vol. 78, no. 2, pp. 2507–2523, Jul. 2018.

[16] A. K. Singh, B. Kumar, S. K. Singh, S. P. Ghrera, and A. Mohan, "Multiple watermarking technique for securing online social network contents using back propagation neural network," *Future Gener. Comput. Syst.*, vol. 86, pp. 926–939, Sep. 2018.

[17] S. Rastegar, F. Namazi, K. Yaghmaie, and A. Aliabadian, "Hybrid watermarking algorithm based on singular value decomposition and radon transform," *AEU Int. J. Electron. Commun.*, vol. 65, no. 7, pp. 658–663, Jul. 2011.

[18] M. Ali, C. W. Ahn, and P. Siarry, "Differential evolution algorithm for the selection of optimal scaling factors in image watermarking," *Eng. Appl. Artif. Intell.*, vol. 31, pp. 15–26, May 2014.

[19] V. Aslantas, "An optimal robust digital image watermarking based on SVD using differential evolution algorithm," *Opt. Commun.*, vol. 282, no. 5, pp. 769–777, Mar. 2009.

[20] C. Kumar, A. K. Singh, and P. Kumar, "Dual watermarking: An approach for securing digital documents," *Multimedia Tools Appl.*, pp. 1–16, Dec. 2019.

[21] A. K. Abdulrahman and S. Ozturk, "A novel hybrid DCT and DWT based robust watermarking algorithm for color images," *Multimedia Tools Appl.*, vol. 78, no. 12, pp. 17027–17049, Jan. 2019.

[22] M. Ali, C. W. Ahn, and M. Pant, "An efficient lossless robust watermarking scheme by integrating redistributed invariant wavelet and fractional Fourier transforms," *Multimedia Tools Appl.*, vol. 77, no. 10, pp. 11751–11773, May 2017.

[23] X.-B. Kang, F. Zhao, G.-F. Lin, and Y.-J. Chen, "A novel hybrid of DCT and SVD in DWT domain for robust and invisible blind image watermarking with optimal embedding strength," *Multimedia Tools Appl.*, vol. 77, no. 11, pp. 13197–13224, Jul. 2017.

[24] E. Najafi and K. Loukhaoukha, "Hybrid secure and robust image watermarking scheme based on SVD and sharp frequency localized contourlet transform," *J. Inf. Secur. Appl.*, vol. 44, pp. 144–156, Feb. 2019.

[25] M. Ali, C. W. Ahn, M. Pant, and P. Siarry, "An image watermarking scheme in wavelet domain with optimized compensation of singular value decomposition via artificial bee colony," *Inf. Sci.*, vol. 301, pp. 44–60, Apr. 2015.

[26] N. M. Makbol, B. E. Khoo, and T. H. Rassem, "Block-based discrete wavelet transform-singular value decomposition image watermarking scheme using human visual system characteristics," *IET Image Process.*, vol. 10, no. 1, pp. 34–52, Jan. 2016.

[27] S. Roy and A. K. Pal, "A robust blind hybrid image watermarking scheme in RDWT-DCT domain using arnold scrambling," *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 3577–3616, Sep. 2016.

[28] M. Ali, C. W. Ahn, and M. Pant, "A robust image watermarking technique using SVD and differential evolution in DCT domain," *Optik*, vol. 125, no. 1, pp. 428–434, Jan. 2014.

[29] H.-C. Ling, R. C.-W. Phan, and S.-H. Heng, "On the security of a hybrid SVD-DCT watermarking method based on lpsnr," in *Proc. Pacific-Rim Symp. Image Video Technol.* Berlin, Germany: Springer, 2011, pp. 257–266.

[30] D. Singh and S. K. Singh, "DWT-SVD and DCT based robust and blind watermarking scheme for copyright protection," *Multimedia Tools Appl.*, vol. 76, no. 11, pp. 13001–13024, Jul. 2016.

[31] A. K. Singh, "Improved hybrid algorithm for robust and imperceptible multiple watermarking using digital images," *Multimedia Tools Appl.*, vol. 76, no. 6, pp. 8881–8900, Apr. 2016.

[32] K. Loukhaoukha, J.-Y. Chouinard, and M. H. Taieb, "Optimal image watermarking algorithm based on LWT-SVD via multi-objective ant colony optimization," *J. Inf. Hiding Multimedia Signal Process.*, vol. 2, no. 4, pp. 303–319, 2011.

[33] N. M. Makbol and B. E. Khoo, "A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition," *Digit. Signal Process.*, vol. 33, pp. 134–147, Oct. 2014.

[34] I. J. Cox, M. L. Miller, and A. L. McKellips, "Watermarking as communications with side information," *Proc. IEEE*, vol. 87, no. 7, pp. 1127–1141, Jul. 1999.

[35] K. Loukhaoukha and J.-Y. Chouinard, "Hybrid watermarking algorithm based on SVD and lifting wavelet transform for ownership verification," in *Proc. 11th Can. Workshop Inf. Theory*, May 2009, pp. 177–182.

[36] 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, Sep. 2012.

[37] C. Jain, S. Arora, and P. K. Panigrahi, "A reliable SVD based watermarking schem," 2008, *arXiv:0808.0309*. [Online]. Available: http://arxiv.org/abs/0808.0309

[38] R. Liu and T. Tan, "An SVD-based watermarking scheme for protecting rightful ownership," *IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 121–128, Mar. 2002.

[39] J. Wang, H. Peng, and P. Shi, "An optimal image watermarking approach based on a multi-objective genetic algorithm," *Inf. Sci.*, vol. 181, no. 24, pp. 5501–5514, Dec. 2011.

[40] M. Alawida, A. Samsudin, J. S. Teh, and R. S. Alkhawaldeh, "A new hybrid digital chaotic system with applications in image encryption," *Signal Process.*, vol. 160, pp. 45–58, Jul. 2019.

[41] M. Alawida, J. S. Teh, A. Samsudin, and W. H. Alshoura, "An image encryption scheme based on hybridizing digital chaos and finite state machine," *Signal Process.*, vol. 164, pp. 249–266, Nov. 2019.

[42] M. Alawida, A. Samsudin, and J. S. Teh, "Enhanced digital chaotic maps based on bit reversal with applications in random bit generators," *Inf. Sci.*, vol. 512, pp. 1155–1169, Feb. 2020.

[43] M. Alawida, A. Samsudin, J. S. Teh, and W. H. Alshoura, "Digital cosine chaotic map for cryptographic applications," *IEEE Access*, vol. 7, pp. 150609–150622, 2019.

[44] M. Alawida, A. Samsudin, and J. S. Teh, "Enhancing unimodal digital chaotic maps through hybridisation," *Nonlinear Dyn.*, vol. 96, no. 1, pp. 601–613, Feb. 2019.

[45] L. Chen, J. Chen, G. Zhao, and S. Wang, "Cryptanalysis and improvement of a chaos-based watermarking scheme," *IEEE Access*, vol. 7, pp. 97549–97565, 2019.

[46] X. Wu and W. Sun, "Robust copyright protection scheme for digital images using overlapping DCT and SVD," *Appl. Soft Comput.*, vol. 13, no. 2, pp. 1170–1182, Feb. 2013.

[47] S. Rawat and B. Raman, "A blind watermarking algorithm based on fractional Fourier transform and visual cryptography," *Signal Process.*, vol. 92, no. 6, pp. 1480–1491, Jun. 2012.

[48] A. Rani and B. Raman, "An image copyright protection scheme by encrypting secret data with the host image," *Multimedia Tools Appl.*, vol. 75, no. 2, pp. 1027–1042, Nov. 2014.

[49] T.-Y. Fan, H.-C. Chao, and B.-C. Chieu, "Medical image watermarking based on visual secret sharing and cellular automata transform for copyright protection," *KSII Trans. Internet & Inf. Syst.*, vol. 12, no. 12, pp. 6177–6200, Dec. 2018.

[50] C.-C. Lai, "A digital watermarking scheme based on singular value decomposition and tiny genetic algorithm," *Digit. Signal Process.*, vol. 21, no. 4, pp. 522–527, Jul. 2011.

[51] R.-S. Run, S.-J. Horng, J.-L. Lai, T.-W. Kao, and R.-J. Chen, "An improved SVD-based watermarking technique for copyright protection," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 673–689, 2012.

[52] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Cyber J., Multidisciplinary J. Sci. Technol., J. Select. Areas Telecommu.*, vol. 1, pp. 31–38, Apr. 2011.

[53] S. Lagzian, M. Soryani, and M. Fathy, "Robust watermarking scheme based on RDWT-SVD: Embedding data in all subbands," in *Proc. Int. Symp. Artif. Intell. Signal Process. (AISP)*, Jun. 2011, pp. 48–52.

[54] K. Loukhaoukha, "Image watermarking algorithm based on multiobjective ant colony optimization and singular value decomposition in wavelet domain," *J. Optim.*, vol. 2013, pp. 1–10, 2013.

[55] M. Ali and C. W. Ahn, "An optimized watermarking technique based on self-adaptive DE in DWT–SVD transform domain," *Signal Process.*, vol. 94, pp. 545–556, Jan. 2014.

WAFA' HAMDAN ALSHOURA received the B.Sc. and M.Sc. degrees in computer science from Al-Zaytoonah University, Jordan, in 2012 and 2017, respectively. She is currently pursuing the Ph.D. degree with the School of Computer Sciences, University Sains Malaysia. Her research interests include digital watermarking and hash function.

ZURINAHNI ZAINOL received the bachelor's degree in computer sciences from Universiti Kebangsaan Malaysia (UKM) and Universiti Teknologi Mara (UITM), the master's degree in artificial intelligence from Universiti Sains Malaysia (USM), and the Ph.D. degree in computer science from the University of Hull, U.K., in 2012. She is currently an Associate Professor with the School of Computer Sciences, Universiti Sains Malaysia. She is also the Deputy Dean for Academic, Career, and International. She has published articles in international journals, conferences, and book chapters. Her research interests include data modeling, XML database schema, optimization algorithm, and information retrieval in multidisciplinary domain, such as medical, health, biological, and education data.

JE SEN TEH received the B.Eng. degree (Hons.) in electronics from Multimedia University, Malaysia, in 2011, the M.Sc. degree in computer science from Universiti Sains Malaysia, in 2013, and the Ph.D. degree from the School of Computer Sciences, Universiti Sains Malaysia, in 2017. He is currently working as a Senior Lecturer. His research interests include cryptography, cryptanalysis, random number generation, machine learning, and chaos theory.

MOATSUM ALAWIDA received the B.Sc. degree from Mutah University Jordan, in 2005, and the M.Sc. degree in information systems from the University of Jordan, in 2010. He is currently pursuing the Ph.D. degree from the School of Computer Sciences, University Sains Malaysia. His research interests include chaotic systems, chaos-based applications, and cryptography.