Multimedia Tools and Applications (2023) 82:15553–15571
https://doi.org/10.1007/s11042-022-13907-8

High payload watermarking based on enhanced image saliency detection

Ahmed Khan[1] · **KokSheik Wong**[2]

Received: 15 March 2022 / Revised: 19 July 2022 / Accepted: 12 September 2022 /
Published online: 8 October 2022

Abstract

Nowadays, images are circulated rapidly over the internet and they are subject to some risk of misuses. To address this issue, various watermarking methods are proposed in the literature. However, most conventional methods achieve a certain trade-off among imperceptibility and high capacity payload, and they are not able to improve these criteria simultaneously. Therefore, in this paper, a robust saliency-based image watermarking method is proposed to achieve high payload and high quality watermarked image. First, an enhanced salient object model is proposed to produce a saliency map, followed by a binary mask to segments the foreground/background region of a host image. The same mask is then consulted to decompose the watermark image. Next, the RGB channels of the watermark are encrypted by using Arnold, 3-DES and multi-flipping permutation encoding (MFPE). Furthermore, the principal key used for encryption is embedded in the singular matrix of the *blue* channel. Moreover, the *blue* channel is encrypted by using the Okamoto-Uchiyama homomorphic encryption (OUHE) method. Finally, these encrypted watermark channels are diffused and embedded into the host channels. When the need arises, more watermarks can be embedded into the host at the expense of the quality of the embedded watermarks. Our method can embed watermark of the same dimension as the host image, which is the first of its kind. Experimental results suggest that the proposed method maintains robustness while achieving high image quality and high payload. It also outperforms the state-of-the-art (SOTA) methods.

Keywords High payload · Image saliency · Robust watermarking · Multiple watermarks

✉ KokSheik Wong
wong.koksheik@monash.edu

Ahmed Khan
ahmed.khan1@monash.edu

[1] School of Information Technology, Monash University Malaysia, Sunway, Malaysia

[2] Advanced Engineering Platform, Monash University Malaysia, Sunway, Malaysia

1 Introduction

Image watermarking (IW) methods have been thoroughly researched by the signal processing community [10, 17, 34, 44]. IW aims to insert a piece of data into the image of interest (host) so that the inserted data can be retrieved later from the potentially modified host to serve a specific purpose. IW has contributed in a variety of applications including copyright protection, image authentication, tampering detection and localization, to name a few [9, 19, 31]. Recently, IW has also been adopted to combat the poisoning of neural network by using malicious training images [37, 39]. More importantly, nowadays anyone with a smart device, which is very affordable, can capture/copy, modify, and broadcast images at one's fingertip. Images can be forged to defame certain individual or to gain an upper hand over certain parties. Therefore, it is crucial to be able verify whether an image is genuinely coming from the claimed source [11, 29].

While various IW techniques were proposed [12, 53], these schemes focus on improving the image quality and/or robustness but they lack in capacity. Among the innovative IW proposals such as watermarking in the encrypted domain and strategic embedding in different frequency transformation domains [3, 8, 27, 52], saliency based IW methods [4, 23, 55] are particularly interesting. It is because the saliency of an image is arguably the most eye-catching or interesting region, hence it is likely that the attacker will retain that region of the image. Therefore, researchers consider to embed the watermark in the salient regions of the image. However, such schemes lacks in terms of payload due to its limited embedding capacity or they focus on only improving one of the three aspect, i.e., image quality, payload, or robustness [1, 2, 13, 20, 50].

Therefore, this paper proposes a unique blend of saliency based robust IW and a homomorphic encryption to simultaneously improve both aspects of IW, namely, quality of the watermarked image and watermark capacity, which depart from the conventional IW methods. First, we extend Bhowmik et al.'s IW technique by (a) performing less DWT decompositions (i.e., only 2-level decomposition), and (b) exploiting the local and global minima (that represents the image background) for more accurate saliency detection purposes. Next, multiple morphological processes are applied to extract the background of the host image, which is subsequently used in extracting the foreground. Our method can embed a watermark that has the same dimension and bit-depth as the host image, which is the first of its kind. Our method also produces high quality watermarked image. Furthermore, each channel of the watermark is encrypted with Arnold, Triple DES (TDES), Multi Flipping Permutation Encoding (MFPE) and the keys are secured with OUHE [41, 49]. Experiment results suggest that our proposed method outperforms the current state-of-the-art (SOTA) methods.

The proposed method can be deployed for copyright protection purposes. Furthermore, in case of multiple watermarks embedding, the proposed method can be enhanced for dual-authentication to realize hierarchical-integrity checking. For example, a well-established content creator/company can protect his/her digital properties (i.e., images) by using his/her watermark W_1. A new small-scale startup can then purchase these images from the established company and further watermarked them with W_2. In case of any malicious usage/attack, W_2 might be destroyed/effected but W_1 remains intact. This is when the hierarchical-copyright protection feature can contribute. This paper makes the following contributions: 1) embed an image-based watermark which is of the same dimension and bit-depth as the host image; 2) secure keys for decryptions by using OUHE and MFPE; 3) achieve high imperceptibility and robustness; 4) improve saliency detection, and; 5) enhancement to embed two or more watermarks (but of lower bit-depth) into one host image.

The rest of this paper is organized as follows: Section 2 reviews the related work, Section 3 details our proposed method, Section 4 presents the experiment results and finally, Section 5 concludes this work.

2 Related work

This section reviews two classes of image watermarking methods, namely, 1) transformation based IW methods, and 2) bitplane based IW methods.

2.1 Transformation based image watermarking methods

Our literature survey suggests that DWT is one of the most commonly adopted techniques to realize watermarking. For example, Qi et al. [38] proposed an IW method to achieve both robustness and quality. First, the host image is divided into 4×4 blocks and the resulting blocks are processed by applying a one-way hash function. Next, 1-level DWT transform is applied on each block to obtain the subbands LL, LH, HL, and HH. The watermark image is processed by using Mersenne Twister algorithm (MTA) and the output is subsequently embedded into the LL subbands. In addition, Chen et al. [7] proposed an IW method to embed a sequence of pseudo-randomly generated bits as the watermark into the DWT-LL subband of the host image. Similarly, Liu et al. [22] embed a watermark of size 64×64 bits into the DWT-LL subband of the host image.

Taking a different approach, Liu et al. [23] propose to divide the host image into $N \times N$ blocks, but instead, only the high intensity blocks (i.e., salient image region) are selected for watermarking purposes. Here, DWT is applied on each block to obtain the subbands. The watermark is then embedded into LL subband of the host image. Similarly, Bhowmik et al. [4] proposed a saliency detection based image watermarking method. First, the host image is converted into the YUV color space and it is then transformed by performing DWT. Upon performing the subbands intensity centering operation, a saliency map is produced, which is used in segmenting the foreground and background regions of the host image. Next, the background of the watermark (i.e., secret image) is embedded into the background of the host image with low strength. In contrast, the foreground of the watermark image is embedded in the foreground of the host image with high strength. Likewise, Zhang et al.'s method [55] also relies on saliency map to embed watermark. Here, the saliency map is produced from the host image by performing logarithmic quantization. Instead of using DWT, Contourlet Transformation (CT) is applied to the host image to get the approximated subbands, which are in-turn serving as the venues to host the watermark image. Another work based on CT is proposed by Najafi et al's. [32]. First, a 2-level Sharp Frequency Localized Contourlet Transformation (SFLCT) is applied to the host image to obtain the approximation and detailed subbands. 1-level SFLCT is also applied to the watermark image. Next, Singular Value Decomposition (SVD) is performed to the detailed subbands of the host and the watermark images. Finally, the S matrix of the subbands of the host image is replaced by the S matrix of the subbands of the watermark by using α-strength diffusion. There are also watermarking methods that are based solely on a single transformation using SVD [24, 26]. However, this class of watermarking methods suffers from limited embedding capacity (i.e., restricted to the number diagonal entries in the S matrix) and low image quality when the embedding rate is high.

It should be noted that other transformations such as Discrete Cosine Transformation (DCT) [16] and Discrete Fourier Transformation (DFT) [33] have also been adopted to embed watermark. Interested readers may refer to these references.

2.2 Bitplane based IW methods

Bitplane based IW methods are usually innovated to serve specific purposes such as authentication and tampering detection. These enhancements are achieved in a few common ways, including (a) spreading the watermark across the host image by means of logistic mapping or permutation, (b) increasing capacity by reserving more bitplanes for watermark embedding purposes, and (c) embedding far less than 1 bit per pixel to achieve high watermarked image quality.

Specifically, Liu et al. [25] applied the logistic map to permute the (binary) watermark, then embed the processed randomized watermark via least significant bit (LSB) embedding. To improve the quality of the watermarked image, Lin et al. [21] and Chang et al. [5] proposed to use absolute moment block truncation coding (AMBTC). Both methods embed the watermark into two LSBs of the host image. To enhance the embedding capacity, Yu et al. [54] reserve 2 bitplanes to embed the watermark.

To realize the application of authentication, Molina et al. [30] and Shehab et al. [42] proposed to embed two types of watermark bits, namely, 1) recovery bits (RB), and 2) authentication bits (AB). The RB's are embedded into the LSBs of the halftoned luminance component of the host image, followed by the embedding of AB's into the LSB of the host. Similar to Molina et al. [30], Tohidi et al. [51] and Chen et al. [6] are also embedding two types of watermark bits. However, they both require auxiliary data to operate, which reduces the embedding efficiency, i.e., number of changes performed per embedded watermark bit.

In addition, there are other innovative ways of using the bitplanes. For example, Singh et al. [45] proposed a fragile watermarking method to improve the quality of the watermarked image. First, the 4th to 8th bitplanes are divided into non-overlapping blocks to generate the ABs. Next, the host image is transformed by using DCT and the ABs are embedded into the LSBs of the selected AC components of the DCT-transformed host image. Furthermore, Sidiropoulos et al. [43] proposed an LSB based embedding method where the randomly generated watermark bits are embedded into the bitplanes of the host image. Similarly, Su et al. [48] scramble the watermark by using Arnold mapping and the bitplanes of the processed watermark are then embedded into the LSB bitplane of the host image. Prasad et al.'s method [36] also follows the same approach put forward by Su et al. [48] but they utilise Logistic map.

Although researchers have put forward various proposals for image watermarking purposes, based on our literature survey, it is noticed that the embedding capacity remains low due to the dependency between capacity and watermarked image quality. Therefore, in this work, we aim to improve both capacity and quality simultaneously.

3 Proposed methodology

Given host image, the salient object in the host color image I is first determined. Segmentation is then performed to obtain its foreground and background. Next, chaotic symmetric crypto-models and homomorphic encryption are applied to the RGB-channels of the watermark W and the outputs are embedded into I as shown in Fig. 1.

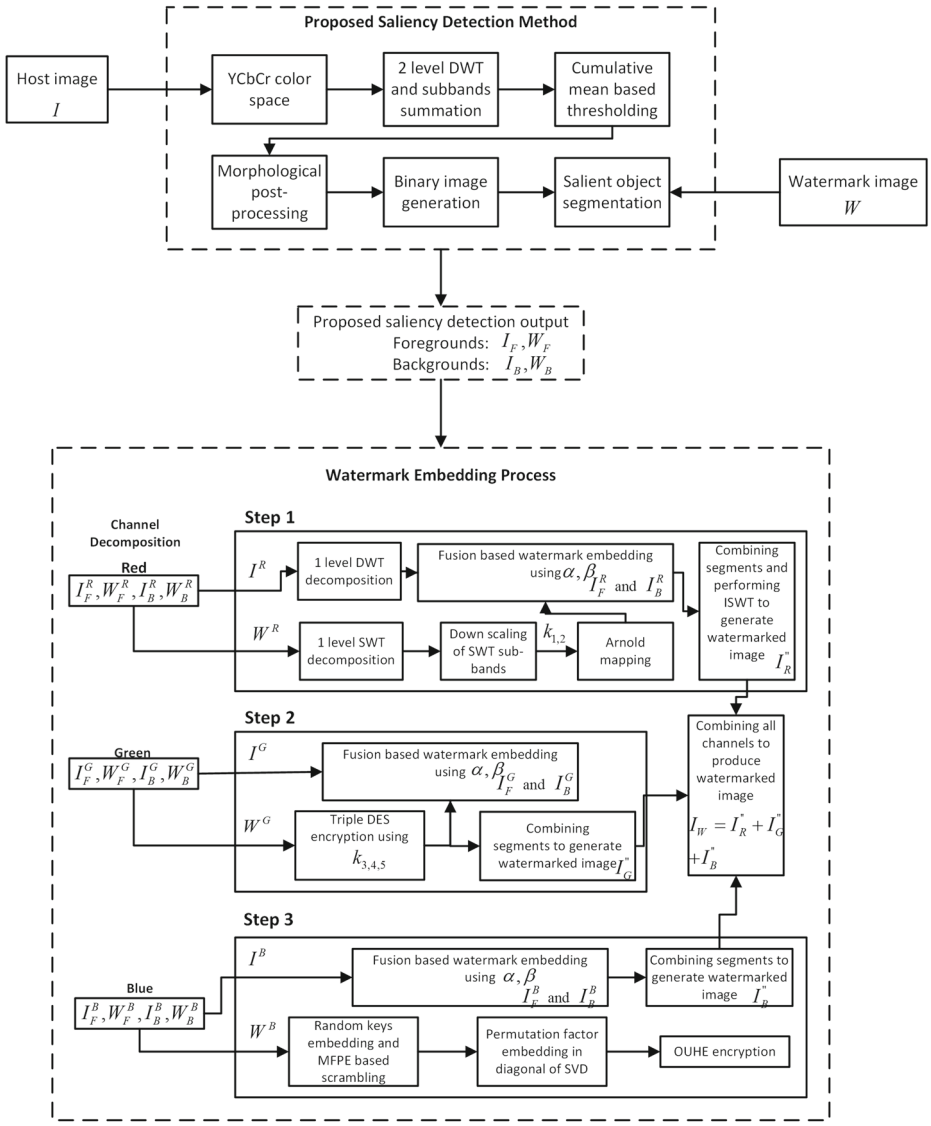


Fig. 1 Proposed saliency model and watermark embedding method

3.1 Salient object detection

We extend the work by Bhomwik et al. [4] by using less number of DWT (as opposed to ≥ 3 level) and considering only the subtraction of local maxima. Furthermore, in our work, a 2-level DWT is applied and the local maxima is subtracted from the image minima to achieve better foreground segmentation. Moreover, our method applies threshold-based segmentation to the average saliency maps generated from the subbands (except LL subbands), which

produces a more accurate binary mask to segment the foreground from the background. Specifically, I is first converted to the YC_bC_r color space. A 2-level DWT is then applied to the luminance (Y) and chrominance channels (C_b, C_r) by using the *Haar* kernel filter

$$\varphi_{(j,k)} = 2^{\frac{j}{2}} \varphi.\psi(2^j x - k), \tag{1}$$

where j and k are the momentous point on the signal and x denotes the magnitude. This operation generates different levels of sub-bands, including LL_{2_φ} and $(HL, LH, HH)_{2_\psi}$. An up-scaling operation is then performed on the HL_{2_ψ}, LH_{2_ψ} and HH_{2_ψ} sub-bands to match the size of LL_{1_φ}. Let $HL^{\uparrow}_{2_\psi}$, $LH^{\uparrow}_{2_\psi}$ and $HH^{\uparrow}_{2_\psi}$ respectively denote these up-scaled subbands. Next, a saliency map S is computed by taking the summation of the obtained sub-bands of both levels of DWTs:

$$S = \sum_u \sum_v (HL_{1_\psi}(u,v) + LH_{1_\psi}(u,v) + HH_{1_\psi}(u,v) + HL^{\uparrow}_{2_\psi}(u,v) + LH^{\uparrow}_{2_\psi}(u,v) + HH^{\uparrow}_{2_\psi}(u,v)). \tag{2}$$

In order to generate a binary mask M_s, a simple thresholding process is applied to the saliency map S, i.e.,

$$M_s(u,v) = \begin{cases} 1 & S(u,v) \geq \mu; \\ 0 & otherwise, \end{cases} \tag{3}$$

where $\mu = S/N$ is the cumulative mean of map S. For further refinement on M_s, morphological erosion, holes filling and black-white open-area filtering are performed to remove small objects from the binary image. A closed operation (dilation-then-erosion) is then applied by using a $disk$ structuring element (SE) with 19 neighborhoods (viz., diameter, $d = 10$). Subsequently, the resulting binary image M'_s segments the salient (viz., foreground) objects in I by using

$$I_F = I * M'_s, \tag{4}$$

where $*$ denotes the *element-wise* multiplication operation, and the background is produced by subtracting the foreground from the image, i.e.,

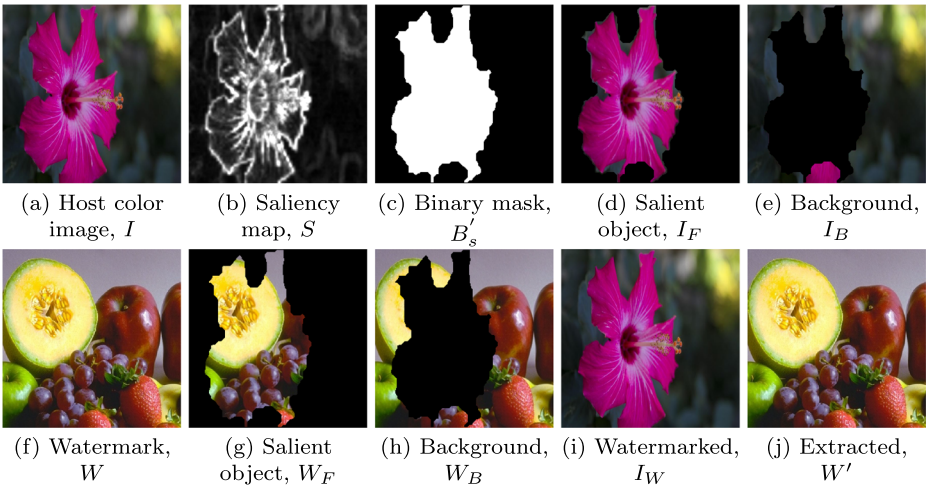
$$I_B = I * (1 - M'_s) \tag{5}$$

Finally, the mask M'_s is applied on the watermark image W, i.e., $W_F = W * M'_s$ and $W_B = W * (1 - M'_s)$. The output of the intermediate steps are shown in Fig. 2.

3.2 Encryption and embedding of watermark

Each color channel of the host image I will be utilized to host one channel of the watermark W. Specifically, the chaotic symmetric crypto-model and the homomorphic encryption proposed by Okamoto-Uchiyama (denoted by OUHE) are applied to all RGB-channels of the foreground W_F and background W_B regions of the watermark W. The outputs are then embedded correspondingly into I_F and I_B. The embedding process is as follows:

Step 1: The *red* channel from the foreground of the watermark image W^R_F is embedded into the *red* channel from the foreground of the host image I^R_F by using sub-band up-scaling and *Haar* wavelet based interpolated DWT-SWT approach. First, 1-level DWT is applied to I^R_F and 1-level SWT is applied to W^R_F. The outputs are correspondingly denoted by $\{LL, HL, LH, HH\}_{I^R_F}$ and $\{LL, HL, LH, HH\}_{W^R_F}$. Down-scaling is then applied on all sub-bands $\{LL, HL, LH, HH\}_{W^R_F}$ to attain the size of the sub-bands of I^R_F. A variant of the Arnold map [15] is then applied to shuffle the



(a) Host color image, I (b) Saliency map, S (c) Binary mask, $B_s^{'}$ (d) Salient object, I_F (e) Background, I_B

(f) Watermark, W (g) Salient object, W_F (h) Background, W_B (i) Watermarked, I_W (j) Extracted, W'

Fig. 2 The intermediate images produced by the proposed watermarking method

pixels in $\{LL, HL, LH, HH\}^{\downarrow}_{W_F^R}$ by using the positive integer key k_1 to generate $\{LL', HL', LH', HH'\}^{\downarrow}_{W_F^R}$ where v is the value at position (x, y):

$$\begin{bmatrix} x' \\ y' \\ v' \end{bmatrix} = \begin{bmatrix} 1 & c_1 & c_2 \\ c_3 & 1 + c_1c_3 & c_2c_3 \\ c_4 & c_1c_2c_3c_4 & 1 + c_2c_4 \end{bmatrix} \begin{bmatrix} x \\ y \\ v \end{bmatrix} \bmod M. \tag{6}$$

Here, we set $c_i = k_1$, and $M = 256$ so that values are clipped to $[0, 255]$. This scrambles the value at position (x, y) to position (x', y') and changes the value from v to v'. The aforementioned processes are repeated for the *red* channel of the background of the watermark and host images, and the positive integer key k_2 is used in generating $\{LL', HL', LH', HH'\}_{W_B^R}$. Finally, α and β are introduced to control the strength in embedding the shuffled sub-bands of the watermark by computing

$$LL_{I_F^{R''}} = LL_{I_F^R} + \alpha * LL^{\downarrow}_{W_F^{R'}} \tag{7}$$

and

$$LL_{I_B^{R''}} = LL_{I_B^R} + \beta * LL^{\downarrow}_{W_B^{R'}}. \tag{8}$$

The HL, HL and HH subbands are computed in the same manner. Note that, in general, $\alpha \geq \beta$ leads to higher imperceptibility. Furthermore, SWT^{-1} is applied on the LL, LH, HL, and HH subbands of $I_F^{R''}$ and $I_B^{R''}$, which form the watermarked *red* channel $I_R'' = I_F^{R''} + I_B^{R''}$. Subsequently, $\{k_1, k_2\}$ are randomly embedded into the *blue* channel of I at locations $\{\vec{x_1}, \vec{x_2}\}$. In addition, as a form of enhancement, more than one watermark can be embedded into the host image. However, the watermark is of lower bit-depth. Specifically, to embed 2, 4, and 8 watermarks, $M = 128, 64$ can be adopted, respectively. However, when embedding more watermarks, the proposed method can only take watermark of lower quality (i.e., lower bit-depth) as the input.

Step 2: The *green* channel from the watermark partitions $\{W_F^G, W_B^G\}$ are encrypted by using TDES [28, 40] crypto model to achieve confidentiality. According to NIST [28], TDES has a security level between DES and AES. Three 56-bit keys, i.e., k_3, k_4 and

k_5 are utilized to encrypt $\{W_F^G, W_B^G\}$ with modulo $M = 256$ (as mentioned in (6)) to produce $\{W_F^{G'}, W_B^{G'}\}$. The ciphertext watermark segments are then embedded into the respective segments in the *green* channel of the host, i.e., $\{I_F^G, I_B^G\}$. Note that $k_3 \neq k_4 \neq k_5$, and they are randomly embedded in the *blue* channel of I at locations $\{\vec{x_3}, \vec{x_4}, \vec{x_5}\}$. Finally, the same parameters α and β (introduced in Step 1) are utilized to control the embedding strength, i.e., $I_F^{G''} = I_F^G + \alpha * W_F^{G'}$ and $I_B^{G''} = I_B^G + \beta * W_B^{G'}$ to produce $\{I_F^G$", I_B^G"$\}$, which are subsequently combined to form the *green* channel of the watermarked image $I_G'' = I_F^{G''} + I_B^{G''}$.

Step 3: The *blue* channel of the watermark partitions $\{W_F^B, W_B^B\}$ are shuffled by MFPE [18]. The resulting encrypted partitions are embedded into the respective *blue* channel partitions of the host, i.e., $\{I_F^B, I_B^B\}$. Specifically, in MFPE, every $2nd$ row and $2nd$ column of the partitions $\{W_F^B, W_B^B\}$ are flipped horizontally (i.e., left to right) and vertically (i.e., up to down), respectively. Furthermore, a random permutation ϕ (viz., a PRNG series that shuffle the position of the pixels) is applied to $\{W_F^B, W_B^B\}$ to produce $\{W_F^{B'}, W_B^{B'}\}$. Furthermore, ϕ is stored as diagonal entries of the singular matrix S (having same dimensions of the original image) of a SVD decomposition. Next, OUHE is applied to diffuse the processed $\{W_F^{B'}, W_B^{B'}\}$, which are later embedded into $\{I_F^B, I_B^B\}$. First, we compute the following to generate public and private keys:

$$n = p^2 q, \tag{9}$$

where p and q are two large prime numbers. We then select $g \in \{2, ..., n-1\}$ that satisfies

$$g^{p-1} \not\equiv 1 \bmod p^2 \tag{10}$$

and finally compute

$$h \equiv g^n \bmod n. \tag{11}$$

Hence, the generated public and private keys are $\{h, n, g\}$ and $\{p, q\}$, respectively. Next, the pixel values are encrypted by using $\{h, n, g\}$ by computing

$$c \equiv g^m h^r \bmod M, \tag{12}$$

where M is the modulo operation from Step 1, $m \in W_F^{B'} \bigcup W_B^{B'}$, $m < p$ and $c \in W_F^{B"} \bigcup W_B^{B"}$. Further diffusion is performed by using $I_F^{B''} = I_F^B + \alpha * W_F^{B''}$ and $I_B^{B''} = I_B^B + \beta * W_B^{B'}$ to jointly form $I_B'' = I_F^{B''} + I_B^{B''}$. Subsequently, the final watermarked image is produced by computing $I_W = I_R'' + f_G'' + I_B''$. A sample of watermarked image is shown in Fig. 2.

3.3 Decryption and extraction of watermark

First, the proposed saliency detection method is applied to segment the watermarked image I_W and the host image I into foreground and background partitions, denoted by $\{I_{W_F}, I_{W_B}\}$ and $\{I_F, I_B\}$, respectively. The encrypted foreground and background of the image are extracted by computing

$$W_F^{B'} = \frac{I_{W_F}^B - I_F^B}{\alpha}, \tag{13}$$

and

$$W_B^{B'} = \frac{I_{W_B}^B - I_B^B}{\beta}. \tag{14}$$

Next, $W_F^{B'}$ is decrypted by using p and q by calculating

$$A = \frac{((W_F^{B'})^{ew(p-1)} \mod p^2) - 1}{p} \tag{15}$$

and

$$B = \frac{(g^{p-1} \mod p^2) - 1}{p}, \tag{16}$$

followed by

$$B' = \delta(B) \mod p, \tag{17}$$

where $\delta(B)$ refers to the matrix containing the reciprocal of all elements in B.[1] Note that in (15), $X^{ew(p)}$ refers to the output of taking each element in X and raised it to the power of p (i.e., element-wise exponential function). Finally, the decrypted watermark $M = W_F^{B''}$ is computed by using the corresponding decryption function for OUHE:

$$M = (A \otimes \delta(B)) \mod p, \tag{18}$$

where $\otimes$ denotes the element-wise multiplication. In a similar manner, $W_B^{B''}$ is obtained by replacing $W_F^{B''}$ by $W_B^{B''}$ in (15).

Furthermore, the SVD decomposition is applied to extract the random permutation order ϕ from the diagonal values of the singular matrix. Multi Flipping Permutation Decoding (MFPD) is then applied on both $W_F^{B''}$ and $W_B^{B''}$ to produce the decoded *blue* partitions W_F^B and $W_B^B\}$, which are then put together to form the *blue* watermarked channel $W^B = W_F^B + W_B^B$. Subsequently, $W_F^{G'}$ and $W_B^{G'}$ are extracted from the *green* watermarked partitions where

$$W_F^{G'} = \frac{I_{W_F}^G - I_F^G}{\alpha} \tag{19}$$

and

$$W_B^{G'} = \frac{I_{W_B}^G - I_B^G}{\beta}. \tag{20}$$

Moreover, the keys $\{k_3, k_4, k_5\}$ are extracted from locations $\{\mathbf{x_3}, \mathbf{x_4}, \mathbf{x_5}\}$ in W_F^B and W_B^B. The TDES decryption modulo 256 operation is performed on $W_F^{G'}$ and $W_B^{G'}$ to produce the decoded *green* partitions W_F^G and W_B^G, which jointly form W^G. Subsequently, the keys $\{k_1, k_2\}$ are extracted from W_F^B and W_B^B at locations $\{\mathbf{x_1}, \mathbf{x_2}\}$. Then,

$$W_F^{R'} = \frac{I_{W_F}^R - I_F^R}{\alpha} \tag{21}$$

and

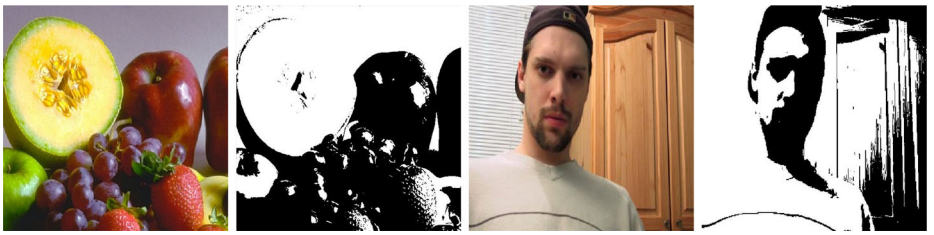
$$W_B^{R'} = \frac{I_{W_B}^R - I_B^R}{\beta} \tag{22}$$

are computed. Next, the inverse Arnold map is computed for $W_F^{R'}$ and $W_B^{R'}$ by using $\{k_1, k_2\}$ to produce the decoded *red* partitions W_F^G and W_B^G, which jointly form the *red* component of watermark $W^R = W_F^{R'} + W_B^{R'}$. Finally, the extracted watermarked image W' is formed by computing

$$W' = W^R + W^G + W^B. \tag{23}$$

An example of W' is shown in Fig. 2(j).

[1] If $B = [2, 3; 4, 5]$, then $\delta(B) = [0.5, 0.333; 0.25, 0.2]$.



(a) A 24-bit Watermark W (b) Binary Watermark W_b (c) A 24-bit Watermark W (d) Binary Watermark W_b

Fig. 3 The watermark images used in the experiments. Each binary watermark is the most significant bitplane of the grayscale image of its color image counterpart

4 Experiments

The proposed watermarking method is implemented in MATLAB 2020 running on a Core i7-7th Gen 7500u 2.9GHz processor with 16GB ram. The standard test images from the SIPI and the MSRA dataset (10K images) [4] are considered for evaluation purposes. Here, the MSRA images and watermarks are resized to $512 \times 512 \times 3$ using the MATLAB function *imresize*. For all experiments conducted, $\alpha = 0.04$ and $\beta = 0.02$ are set, which makes the background slightly blurry while the salient object remains completely imperceptible. We utilize the 24-bit image shown in Fig. 3 as the watermark and embed it into the host image. For evaluation purposes, we consider the following scenarios: (a) PSNR and aSSIM [4] of the watermarked image after embedding a watermark, i.e., 24-bit color image W or a binary image W_b, (b) aSSIM of the extracted W and, (c) normalized coefficient (NC) of the extracted W_b. Here, aSSIM refers to the average of the SSIM values for each of the RGB channels. In addition, for the case of binary watermark W_b, we embed the same binary watermark into all three color channels of the host image.

4.1 Saliency detection

First, the performance of the proposed salient object detection (SOD) method is evaluated. Some representative results are shown in Fig. 4. By visual inspection, it is verified that the proposed method is able to identify the salient objects (regions). It produces boundary that confines the salient object with a wider dynamic range. For example, for the bird image (3rd column), our method can detect the feather and beak completely and the detected objects are brighter than that of [4]. To quantify the results, the Mean Absolute Error (MAE), F1-score, Area Under Receiver Operating Characteristic (AUROC) are recorded in Table 1.[2] Specifically, the metrics are computed as follows:

$$MAE = \frac{1}{M \times N} \sum_{x}^{M} \sum_{y}^{N} |S(x, y) - G(x, y)|, \tag{24}$$

[2]The result for [4] is only available in the form of average AUROC, which is 0.887 and it is lower than those achieved by the proposed method.

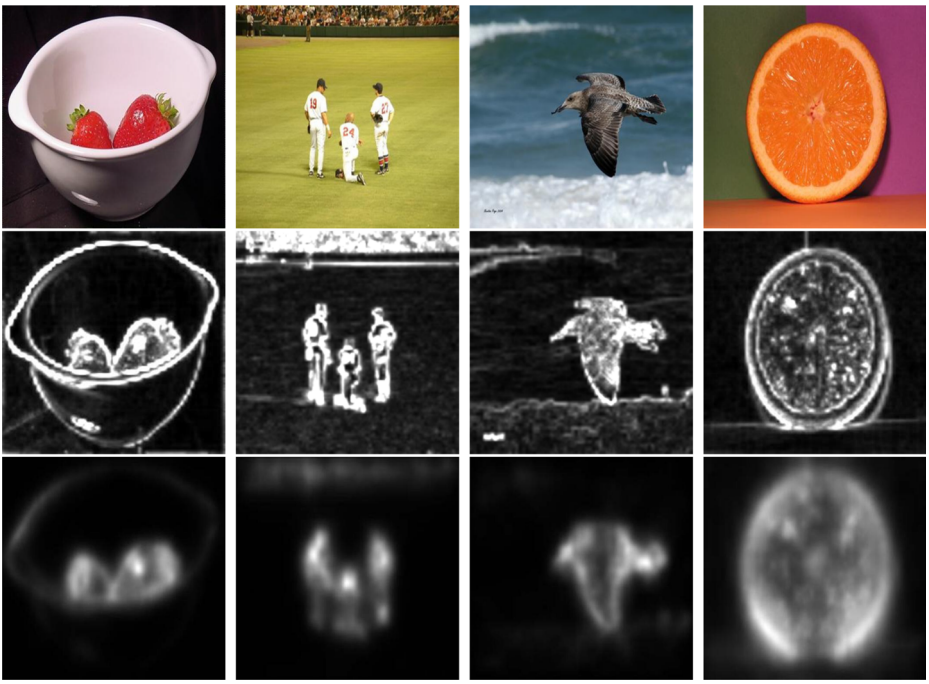


Fig. 4 Comparison of saliency map produced by our proposed method and Bhowmik et al.'s method [4] . The original images, saliency map by our proposed method and [4] are shown in the 1st, 2nd and 3rd row, respectively

where S and G are the saliency map and the ground truth, respectively. On the other hand, F1 is defined as:

$$F1 = \frac{TP}{TP + 0.5 \times (FP + FN)}, \tag{25}$$

where TP, FN and FP are the true positive, false negative, and false positive, respectively. Likewise, specificity is defined as:

$$FPR = 1 - \frac{TN}{TN + FP}, \tag{26}$$

where TN refer to the true negative. On average, 0.035, 0.790 and 0.900 are attained for MAE, F1 and AUROC, respectively. Here, we also consider the results for Peng et al.'s

Table 1 Performance of the generated saliency maps. Results are presented in the format of "proposed"/"Peng et al. [35]"/"Singh et al. [46]" method for MSRA dataset

Image	MAE(↑)	F1(↑)	AUROC(↑)
Strawberry	**0.039**/0.104/0.120	**0.800**/0.704/0.710	**0.910**/0.847/0.800
Players	**0.038**/0.104/0.120	**0.790**/0.704/0.710	**0.900**/0.847/0.820
Bird	**0.032**/0.104/0.130	**0.780**/0.704/0.710	**0.890**/0.847/0.810
Orange	**0.033**/0.104/0.130	**0.790**/0.704/0.720	**0.890**/0.847/0.800

method [35] and Singh et al.'s method [46]. Results suggest that our proposed method outperforms these SOTA SOD methods by at least MAE$\geq$ 66%, F1 $\geq$ 60%, and AUROC$\geq$ 53%. Therefore, we conclude that our salient object detection method outperforms [4, 35] and [46] .

4.2 Quality of watermarked image

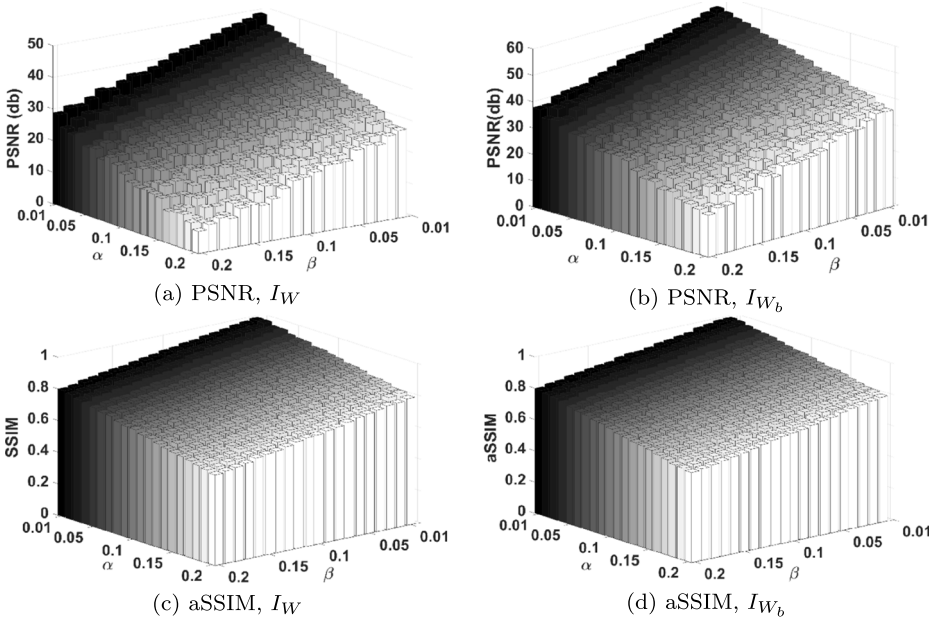
The quality of the watermarked image after embedding a 24-bit color watermark W and binary watermark W_b is evaluated in terms of PSNR and aSSIM. The results are recorded in Table 2. The PSNR value of the watermarked image after embedding W and W_b are consistently $\geq$ 43dB and $\geq$ 54dB, respectively. It is noticed that, in comparison to embedding W_b, the quality of the watermarked image is significantly lower after embedding W. This is due to the fact that embedding W will affect all bitplanes in all channels of the host image, but in the case of embedding W_b, only one bitplane in each channel of the host image is modified. Besides embedding in all bitplanes of the RGB channels of the host image, the difference is also due to the precision error in floating point representation and its related operations. Similar observation is made for the case of aSSIM for embedding W and W_b. The mean aSSIM of the watermarked image for W is $\geq$ 0.9986, whereas aSSIM for W_b is $\geq$ 0.9999.

To investigate how the quality of the watermarked image changes when using different parameter settings, results are collected by varying α and β within the range of [0.01, 0.2]. The results are summarized in Fig. 5. In case of embedding W, the PSNR value ranges from 5 to 48dB, while the aSSIM range is 0.5612 to 0.9995. However, for embedding W_b, the observed PSNR value ranges from 15 to 56dB, and the aSSIM value ranges from 0.5789 to 0.9999. It is apparent that the quality of the watermarked image decreases when either α or β increases, and vice versa (Fig. 6).

When compared to the conventional SOTA methods, we focus on the results collected for embedding W_b because the conventional methods embed a binary image as the watermark. Results in Table 2 suggest that the PSNR attained by our proposed method is the highest, i.e., +18dB for the case of [4], and +13 for both [23, 55]. On the other hand, the aSSIM value shows mixed performances. As compare to [23, 55], our proposed method performs better by 8%, but our proposed method is on par with [4]. It is noteworthy that our

Table 2 Quality analysis of the watermarked image after embedding the color watermark W and the binary watermark W_b (format: PSNR/aSSIM) for the proposed method and [4, 23, 55]

Images	Proposed (W)	Proposed (W_b)	[23][1], [55][2], [4][3]
Lena	43.90/0.9989	**55.65/0.9999**	42.00/0.9100[1], 42.00/0.9989[2]
Barbara	44.10/0.9991	**54.31/0.9999**	42.00/0.9100[1], 42.00/0.9991[2]
Strawberry	43.72/0.9988	**54.62/0.9999**	36.07/0.9999[3]
Players	44.31/0.9987	**54.19/0.9999**	36.42/0.9999 [3]
Bird	44.19/0.9986	**55.10/0.9999**	36.18/0.9999 [3]
Orange	43.45/0.9987	**55.53/0.9999**	36.42/0.9999 [3]
MSRA	43.60/0.9988	**54.95/0.9999**	36.30/0.9999[3]



(a) PSNR, I_W (b) PSNR, I_{W_b}

(c) aSSIM, I_W (d) aSSIM, I_{W_b}

Fig. 5 PSNR and aSSIM graphs of the watermarked image with various α and β

proposed method still outperforms [23, 55] when embedding W, i.e., a significantly larger payload (watermark) size. In addition, our proposed method (embedding W) is marginally inferior in comparison to [4] (embedding W_b). These results suggest that, despite embedding highly diffused watermark images of the same dimension and bit-depth as the host image, the output watermarked image produced by our proposed method is of high quality. Hence, the proposed method produces high quality watermarked images as compared to the conventional saliency-based IW methods [4, 23, 55]. In addition, for the SOTA methods considered in this work [5, 21, 47], the binary watermark image is embedded into the LSB bitplane of the host images, which makes them vulnerable to malicious attacks hence less robust.

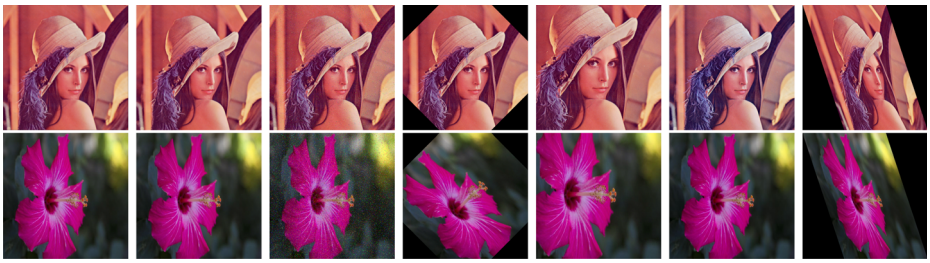


Fig. 6 Attacks on watermarked Leena (from SIPI) and Flower image (from MSRA dataset)

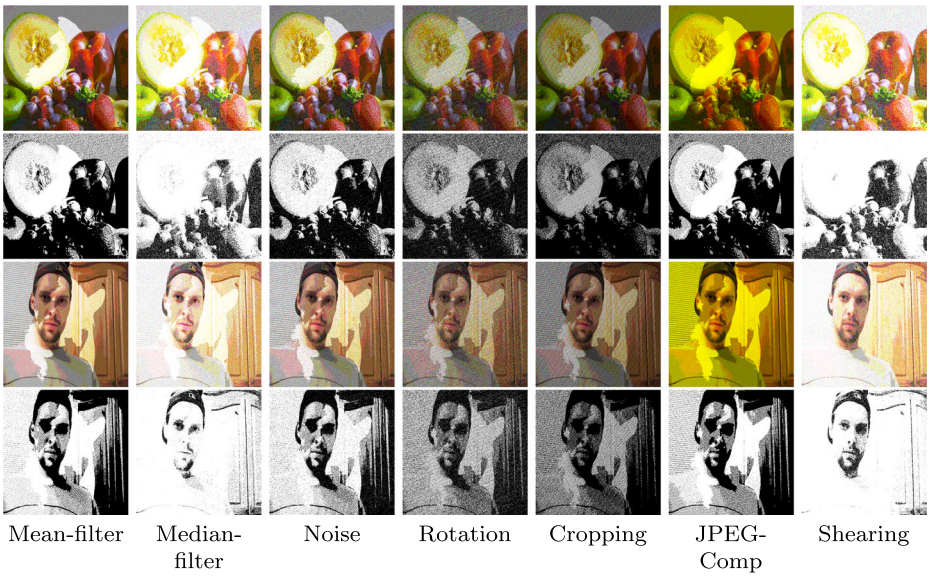


Fig. 7 The first and third rows show the extracted color watermark W from the watermarked image Leena. The second and fourth rows show the extracted binary watermark W_b

4.3 Robustness of the embedded watermark

This section reports the robustness of the embedded watermark by calculating aSSIM and Normalized Correlation (NC). Specifically, NC is computed as follows:

$$NC = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(W_{(i,j)} - \mu_W)(W'_{(i,j)} - \mu_{W'})}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(W_{(i,j)} - \mu_W)^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(W'_{(i,j)} - \mu_{W'})^2}}, \tag{27}$$

where μ_W and $\mu_{W'}$ are the mean of the original watermark W and the extracted watermark W', respectively. Specifically, the proposed method is evaluated by embedding watermark in two different ways, namely, a) embedding a 24-bit watermark W and, b) a binary watermark W_b. Various attacks including mean-filtering, median-filtering, shearing, noise, rotate, cropping and JPEG compression attacks are performed on the watermarked images for both scenarios and the results are recorded. Table 3 records the aSSIM value for the case of 24-bit watermark W, and Table 4 records the NC value for the case of binary watermark W_b.

First, we consider scenario a) where a 24-bit watermark is embedded. When there are no attacks applied on the watermarked image, the aSSIM value is 0.9999, which suggests that the extracted watermarks are of high quality. When any form of attack is applied, the aSSIM value drops, but the average aSSIM for the extracted W remains high at 0.9300. The proposed method appears to be particularly robust against mean-filtering and noise attack. As expected, the aSSIM value for cropping is particularly low because certain image information is completely removed from the watermarked image. Since the proposed method is the only known method that embeds a 24-color image as the watermark, there are no existing methods for comparison purposes.

Next, we consider scenario b), where a binary watermark (Fruits and Male as shown in Fig. 3(b) and (d), respectively) is embedded, and the results are recorded in Table 4. Here,

Table 3 The aSSIM values of the extracted 24-bit color watermark

Images	No Attack	Mean	Median	Noise	Rotation	Cropping	Compression	Shear
Leena	0.9999	0.9512	0.9445	0.9511	0.9530	0.9100	0.9319	0.9488
Barbara	0.9999	0.9711	0.9488	0.9509	0.9455	0.9262	0.9201	0.9429
Baboon	0.9999	0.9702	0.9512	0.9521	0.9467	0.9199	0.9272	0.9412
Peppers	0.9999	0.9721	0.9482	0.9514	0.9431	0.9278	0.9212	0.9411
MSRA	0.9999	0.9820	0.9499	0.9542	0.9462	0.9107	0.9222	0.9401
Leena	0.9999	0.9572	0.9491	0.9535	0.9569	0.9211	0.9362	0.9448
Barbara	0.9999	0.9737	0.9439	0.9416	0.9489	0.9194	0.9217	0.9442
Baboon	0.9999	0.9733	0.9555	0.9549	0.9380	0.9202	0.9291	0.9438
Peppers	0.9999	0.9661	0.9509	0.9545	0.9449	0.9192	0.9229	0.9511
MSRA	0.9999	0.9711	0.9502	0.9575	0.9483	0.9211	0.9251	0.9418

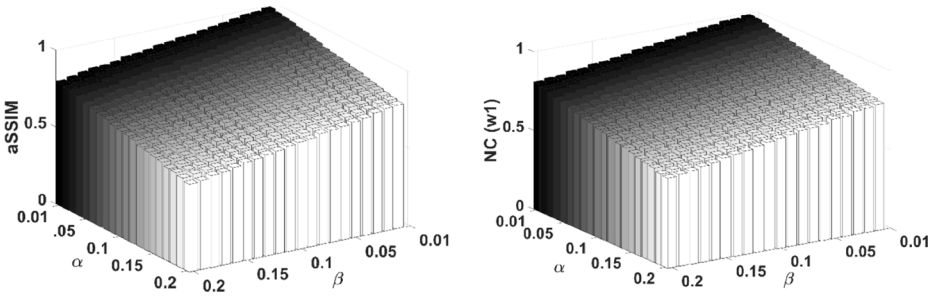
Rows 2-6 and 7-11 are the results for Fruits and Male from Fig. 3, respectively

the majority vote strategy is adopted because three copies of the same watermark (one from each of the RGB channels) are available. In addition, the discussion here is based on the Leena (as watermarked) image because it is the only image that is commonly evaluated by all the existing methods considered for the comparison. Among the seven types of attacks, results suggest that Jiang et al.'s method [14] is the most robust IW method against both mean-filter and median-filter attacks, which is, on average, $\geq$ 6% higher than the rest. On the other hand, Zhang et al.'s method [55] outperforms other methods for the cases of rotation and JPEG-compression attacks, which is, on average, $\geq$ 5% higher than the rest. For the remaining three attacks, namely shearing, noise, and cropping, our proposed methods exhibits the highest resistance, with an average margin of 4%. It is noteworthy that the proposed method is originally designed to embed 24-bit watermark, but it has then been modified to embed binary image solely for comparison purposes, because most conventional methods can only embed a binary image as the watermark.

For completion of discussion, the aSSIM and NC values of the extracted watermarks are analyzed when considering different parameter values. The results are shown in Fig. 8 for

Table 4 NC results after applying various attacks on watermarked Leena

Images	Mean	Median	Noise	Rotation	Cropping	Compression	Shear
Fruits	0.9458	0.9341	**0.9499**	0.9227	**0.9157**	0.9093	**0.9326**
Male	0.9442	0.9389	0.9472	0.9219	0.9141	0.9038	0.9399
[14]	**0.9750**	**0.9693**	0.9215	0.9604	0.9208	0.9327	0.9306
[32]	0.9567	0.9382	0.9466	0.9691	0.9188	0.9438	0.9291
[55]	0.8500	0.9211	0.9387	**0.9761**	0.9022	**0.9509**	0.9319
[16]	0.9388	0.9361	0.8805	0.8848	0.9031	0.9429	0.9311



(a) aSSIM, W (b) NC, W_b

Fig. 8 The aSSIM and NC graphs for the extracted watermarks for various values of α and β

various values of α and β. Similar to the observations on the quality of the watermarked image, the quality of the extracted watermark is higher when either α or β is small, and vice versa (see Figs.7 and 8). However, for security reasons, one should not always choose the smallest α and β when using the proposed method, although the best quality is achieved with these settings for both the watermarked image as well as the extracted watermark.

5 Conclusion

In this work, we proposed a salient-based image watermarking method. Specifically, we improve Bhowmik et al.'s salient object detection method [4]. Specifically, a salient object detection model is first proposed to extract the visually attentive area in the host image for generating a saliency mask. This mask is then applied to divide the foreground and background of the host and watermark images. Then, the *red*, *green* and *blue* watermark channels are encrypted by using Arnold, TDES and MFPE, respectively. Furthermore, the principal key is embedded in the singular diagonal of the *blue* channel that can be used subsequently to produce all dependent keys. Next, the *blue* channel is encrypted by using Okamoto-Uchiyama homomorphic encryption. These scrambled and encrypted watermark channels are then embedded into the respective host channels. Unlike the conventional method that hides a binary image into the host image, the proposed method can embed a 24-bit image of the same dimension as the host. In addition, more than one watermark can be embedded when such need arises, but at the expenses of a lower quality watermark. Analysis results also indicate that the proposed method outperforms SOTA methods in terms of imperceptibility, payload and robustness.

As future work, we want to analyze the effect of different wavelets and decomposition levels on the performance of proposed watermarking method. Furthermore, in case of multiple watermarks embedding, high resolution reconstruction of the watermark images along with contrast enhancement can be explored to extract high quality watermark.

Acknowledgements This work was supported by the Advanced Engineering Platform's Cluster Funding (account number AEP-2021-Cluster-04), Monash University Malaysia, Malaysia.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Data Availability The datasets analyzed during the current study are available in the Google drive repository - https://tinyurl.com/5p5zpwhw

Declarations Mentioned authors have no conflict of interest upon this article. This article does not contain any studies with human participants or animals performed by any of authors.

References

1. Abdallah HA, Faragallah OS, Elsayed HS, Shaalan AA, Abd El-samie FE et al (2016) Robust image watermarking method using homomorphic block-based klt. Optik 127(4):2374–2381
2. Alshanbari HS (2021) Medical image watermarking for ownership & tamper detection. Multimed Tools Appl, vol 80(11)
3. Bhalerao S, Ansari IA, Kumar A (2021) A secure image watermarking for tamper detection and localization. J Ambient Intell Humanized Comput 12(1):1057–1068
4. Bhowmik D, Oakes M, Abhayaratne C (2016) Visual attention-based image watermarking. IEEE Access 4:8002–8018
5. Chang C-C, Lin C-C, Su G-D (2020) An effective image self-recovery based fragile watermarking using self-adaptive weight-based compressed ambtc. Multimed Tools Appl 79(33):24 795–24 824
6. Chen W-C, Wang M-S (2009) A fuzzy c-means clustering-based fragile watermarking scheme for image authentication. Expert Syst Appl 36(2):1300–1307
7. Chen D, Wu Q, Wang Y (2006) Watermarking authentication: using the fragile mode or the robust mode? J Electr (China) 23(4):549–553
8. Chotikawanid P, Thongkor K, Amornraksa T (2015) Homomorphic filter based image watermarking. Appl Mech Materials Trans Tech Publ 781:543–546
9. Chu WC (2003) Dct-based image watermarking using subsampling. IEEE Trans Multimedi 5(1):34–38
10. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2007) Digital watermarking and steganography. Morgan Kaufmann
11. Gull S, Loan NA, Parah SA, Sheikh JA, Bhat GM (2020) An efficient watermarking technique for tamper detection and localization of medical images. J Ambient Intell Humanized Comput 11(5):1799–1808
12. Huang H-C, Fang W-C (2007) Intelligent multimedia data hiding: new directions. Springer, vol 58
13. Huang Y, Niu B, Guan H, Zhang S (2019) Enhancing image watermarking with adaptive embedding parameter and psnr guarantee. IEEE Trans Multimed 21(10):2447–2460
14. Jiang F, Gao T et al (2020) A robust zero-watermarking algorithm for color image based on tensor mode expansion. Multimed Tools Appl:1–16
15. Jithin K, Sankar S (2020) Colour image encryption algorithm combining arnold map, dna sequence operation, and a mandelbrot set. J Inf Secur Appl 50:102428
16. Kamili A, Hurrah NN, Parah SA, Bhat GM, Muhammad K (2020) Dwfcat: dual watermarking framework for industrial image authentication and tamper localization. IEEE Trans Industr Inf 17(7):5108–5117
17. Kavitha K, Priestly Shan B (2021) A watermarking scheme using intellectual encoding-encryption based blind reversible integer wavelet-singular value decomposition transform for authenticity detection. J Ambient Intell Humanized Comput 12(7):7715–7726
18. Khan A, Sarfaraz A (2019) Novel high-capacity robust and imperceptible image steganography scheme using multi-flipped permutations and frequency entropy matching method. Soft Comput 23(17):8045–8056
19. Khare P, Srivastava VK (2020) A reliable and secure image watermarking algorithm using homomorphic transform in dwt domain. Multidim Syst Sign Process:1–30

20. Li L, Wang S, Zhang S, Luo T, Chang C-C (2020) Homomorphic encryption-based robust reversible watermarking for 3d model. Symmetry 12(3):347
21. Lin C-C, He S-L, Chang C-C (2021) Pixel-based fragile image watermarking based on absolute moment block truncation coding. Multimed Tools Appl 80(19):29 497–29 518
22. Liu X-L, Lin C-C, Yuan S-M (2016) Blind dual watermarking for color images' authentication and copyright protection. IEEE Trans Circuits Syst Video Technol 28(5):1047–1055
23. Liu H, Liu J, Zhao M (2018) Visual saliency model-based image watermarking with laplacian distribution. Information 9(9):239
24. Liu F, Ma L, Liu C, Lu Z-M (2018) Zero watermarking scheme based on u and v matrices of quaternion singular value decomposition for color images. J Inf Hiding Multim Signal Process 9(3):629–640
25. Liu S-H, Yao H-X, Gao W, Liu Y-L (2007) An image fragile watermark scheme based on chaotic image pattern and pixel-pairs. Appl Math Comput 185(2):869–882
26. Loukhaoukha K (2012) On the security of digital watermarking scheme based on svd and tiny-ga. J Inf Hiding Multimed Signal Process 3(2):135–141
27. Lu C-S, Huang S-K, Sze C-J, Liao H-YM (2000) Cocktail watermarking for digital image protection. IEEE Trans Multimedia 2(4):209–224
28. Mitchell CJ (2016) On the security of 2-key triple des. IEEE Trans Inf Theory 62(11):6260–6267
29. Mohanarathinam A, Kamalraj S, Prasanna Venkatesan G, Ravi RV, Manikandababu C (2020) Digital watermarking techniques for image security: a review. J Ambient Intell Humanized Comput 11(8):3221–3229
30. Molina-Garcia J, Garcia-Salgado BP, Ponomaryov V, Reyes-Reyes R, Sadovnychiy S, Cruz-Ramos C (2020) An effective fragile watermarking scheme for color image tampering detection and self-recovery. Signal Process Image Commun 81:115725
31. Mukherjee DP, Maitra S, Acton ST (2004) Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans Multimed 6(1):1–15
32. Najafi E, Loukhaoukha K (2019) Hybrid secure and robust image watermarking scheme based on svd and sharp frequency localized contourlet transform. J Inf Secur Appl 44:144–156
33. Nejati F, Sajedi H, Zohourian A (2022) Fragile watermarking based on qr decomposition and fourier transform. Wirel Pers Commun 122(1):211–227
34. Pan PJ-S, Huang H-C, Jain LC (2004) Intelligent watermarking techniques (with Cd-rom). World scientific, vol 7
35. Peng H, Li B, Ling H, Hu W, Xiong W, Maybank SJ (2016) Salient object detection via structured matrix decomposition. IEEE Trans Pattern Anal Mach Intell 39(4):818–832
36. Prasad S, Pal AK (2020) Hamming code and logistic-map based pixel-level active forgery detection scheme using fragile watermarking. Multimed Tools Appl 79(29):20 897–20 928
37. Puteaux P, Ong S, Wong K, Puech W (2021) A survey of reversible data hiding in encrypted images – the first 12 years. J Vis Commun Image Represent 77:103085
38. Qi X, Xin X (2015) A singular-value-based semi-fragile watermarking scheme for image content authentication with tamper localization. J Vis Commun Image Represent 30:312–327
39. Ramasamy R, Arumugam V (2021) Robust image watermarking using fractional krawtchouk transform with optimization. J Ambient Intell Humanized Comput 12(7):7121–7132
40. Reddy IRS, Murali G (2017) A novel triple des to enhance e-governance security. In: 2017 International conference on energy, communication, data analytics and soft computing. IEEE, pp 2443–2446
41. Ridho A, Tulus, Efendi S (2020) Optimization of the gronsfeld cipher key using okamoto-uchiyama public-key cryptosystem for data security. In: 2020 3rd International conference on mechanical electronics, computer, and industrial technology, pp 123–126
42. Shehab A, Elhoseny M, Muhammad K, Sangaiah AK, Yang P, Huang H, Hou G (2018) Secure and robust fragile watermarking scheme for medical images. IEEE Access 6:10 269–10 278
43. Sidiropoulos P, Nikolaidis N, Pitas I (2009) Invertible chaotic fragile watermarking for robust image authentication. Chaos Solitons Fractals 42(5):2667–2674
44. Singh SP, Bhatnagar G (2020) A robust blind watermarking framework based on dn structure. J Ambient Intell Humanized Comput 11(5):1869–1887
45. Singh D, Singh SK (2017) Dct based efficient fragile watermarking scheme for image authentication and restoration. Multimed Tools Appl 76(1):953–977
46. Singh SK, Srivastava R (2020) A robust salient object detection using edge enhanced global topographical saliency. Multimed Tools Appl:1–18
47. Su G-D, Chang C-C, Chen C-C (2021) A hybrid-sudoku based fragile watermarking scheme for image tampering detection. Multimed Tools Appl 80(8):12 881–12 903

48. Su G-D, Chang C-C, Lin C-C (2020) Effective self-recovery and tampering localization fragile watermarking for medical images. IEEE Access 8:160 840–160 857
49. Suwandi R, Nasution SM, Azmi F (2016) Okamoto-uchiyama homomorphic encryption algorithm implementation in e-voting system. In: 2016 International conference on informatics and computing, pp 329–333
50. Thanki R, Borra S (2019) Fragile watermarking for copyright authentication and tamper detection of medical images using compressive sensing (cs) based encryption and contourlet domain processing. Multimed Tools Appl 78(10):13 905–13 924
51. Tohidi F, Paul M, Hooshmandasl MR (2021) Detection and recovery of higher tampered images using novel feature and compression strategy. IEEE Access 9:57 510–57 528
52. Verma D, Aggarwal A, Agarwal H (2017) Watermarking scheme based on singular value decomposition and homomorphic transform. AIP Conf Proc 1897(1):020036
53. Wang F-H, Pan J-S, Jain LC (2009) Digital watermarking techniques. In: Innovations in digital watermarking techniques. Springer, pp 11–26
54. Yu M, Wang J, Jiang G, Peng Z, Shao F, Luo T (2015) New fragile watermarking method for stereo image authentication with localization and recovery. AEU-Int J Electr Commun 69(1):361–370
55. Zhang Y, Sun Y (2019) An image watermarking method based on visual saliency and contourlet transform. Optik 186:379–389