Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.037798
Article

A Novel Color Image Watermarking Method with Adaptive Scaling Factor Using Similarity-Based Edge Region

Kali Gurkahraman[1,*], Rukiye Karakis[2] and Hidayet Takci[1]

[1]Faculty of Engineering, Department of Computer Engineering, Sivas Cumhuriyet University, Sivas, 58140, Turkey
[2]Faculty of Technology, Department of Software Engineering, Sivas Cumhuriyet University, Sivas, 58140, Turkey
*Corresponding Author: Kali Gurkahraman. Email: kgurkahraman@cumhuriyet.edu.tr
Received: 16 November 2022; Accepted: 06 January 2023; Published: 26 May 2023

Abstract: This study aimed to deal with three challenges: robustness, imperceptibility, and capacity in the image watermarking field. To reach a high capacity, a novel similarity-based edge detection algorithm was developed that finds more edge points than traditional techniques. The colored watermark image was created by inserting a randomly generated message on the edge points detected by this algorithm. To ensure robustness and imperceptibility, watermark and cover images were combined in the high-frequency subbands using Discrete Wavelet Transform and Singular Value Decomposition. In the watermarking stage, the watermark image was weighted by the adaptive scaling factor calculated by the standard deviation of the similarity image. According to the results, the proposed edge-based color image watermarking technique has achieved high payload capacity, imperceptibility, and robustness to all attacks. In addition, the highest performance values were obtained against rotation attack, to which sufficient robustness has not been reached in the related studies.

Keywords: Image watermarking; edge detection; discrete wavelet transform; singular value decomposition; adaptive scaling factor

1 Introduction

Encryption or data hiding techniques are used to ensure secure communication in open networks. Encryption is concerned with transforming a message into an incomprehensible format, while data hiding deals with embedding a secret message into the media. Data hiding is classified into steganography or watermarking, depending on the message type. Steganography embeds a message in a media such as text, image, or video file, whereas watermarking hides a media file in another media [1,2]. Images are the most preferred media in steganography and watermarking because of data hiding capacity and easily transfer in open networks. Embedding techniques used to hide the secret message into the image are performed in the spatial or transform domains [1,3,4]. The spatial domain techniques are simple and have a high hiding capacity. However, they are not robust to attacks such as filtering, scaling, Gaussian noise, joint photographic experts group (JPEG) compression, rotation,

or cropping [4]. In the transform domain, the message is embedded in the frequency coefficients of the images produced using Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), Integer Wavelet Transform (IWT), Lifting Wavelet Transform (LWT), Redundant DWT (RDWT), Contourlet Transform (CT), Karhunen–Loeve Transform (KLT), and Walsh Hadamard Transform (WHT) etc. [3–7]. Although transform methods are more complex than spatial techniques and provide a lower hiding capacity, they are more resistant to stegattacks. DWT is more useful in determining coefficients than DCT because it can distinguish between high and low-frequency components. Additionally, DWT is computationally efficient due to the use of simple filter convolution [3].

In digital watermarking, special data called a watermark is hidden in a multimedia file such as text, sound, image, or video. Watermarking is widely used for preventing illegal duplication and copyright protection, automated monitoring, fingerprinting, indexing, securing medical data, and content verification. Watermarking methods are grouped as robust or fragile. In robust watermarking, secret information in the cover is robust to attacks, and the information can be obtained later. In fragile watermarking, it is determined whether the cover is original or not based on the corruption caused by possible attacks [3]. The methods developed for the watermark extraction are grouped as blind, semi-blind, and non-blind. For the watermark extraction, while the blind method uses only watermarked images, the non-blind method processes both the original and watermarked images. In semi-blind watermarking, the watermark or original image may be required in addition to watermarked image for the extraction stage [3,5–7].

This study proposes a new color image watermarking technique using a similarity-based edge algorithm and adaptive scaling factor to solve the robustness, imperceptibility, and capacity problems that watermarking has to deal with.

The rest of the study is organized as follows. Related work is presented in Section 2, and the motivation and contributions of the study are described in Section 3. The material and the proposed methods are given in Section 4. Section 5 provides the experimental results of the proposed watermarking method and discussion. Finally, Section 6 includes the conclusion.

2 Related Work

The frequency domain is preferred in order to make the result of the data hiding more robust to attacks in image watermarking. In recent years, hybrid methods have been proposed since the robustness of the watermarking schemes in the frequency domain against steg-attacks cannot be improved further. Singular value decomposition (SVD) is widely used in many hybrid image watermarking methods because its implementation is simple and ensures high robustness [8].

Jane et al. [9] have presented a non-blind hybrid watermarking method based on DWT and SVD. With DWT, both the cover image and watermark were decomposed into four subbands, and SVD was applied in the low-frequency bands. The singular values obtained were combined by means of a scaling factor. Finally, low-low (LL) frequency band (approximation) coefficients were obtained using inverse SVD, and then a watermarked image was generated using inverse DWT. Watermarking with the developed method provided a 20% improvement in peak signal-to-noise ratio (PSNR) values. Although the proposed method is robust to most steg-attacks, it was unable to achieve high PSNR values, particularly in JPEG and rotation attacks. In another study [10], different levels of DWT were applied to watermark and cover images, and the singular values obtained by SVD of low and high-frequency bands were combined. The method was resistant to attacks such as noise, histogram equalization (HE), and cropping. Begum et al. [11] watermarked a 64×64 binary logo encoded by

Arnold transform (AT) on gray images with a hybrid watermarking technique that includes DCT, DWT, and SVD. The proposed scheme was robust to median filtering, salt and pepper noise, and rotation attacks. Al-Afandy et al. [12,13] watermarked the color logo into the same size 512×512 color images using two-hybrid methods including homomorphic-based SVD + DWT and Discrete Stationary Wavelet Transform (DSWT) + DCT methods. The developed hybrid techniques were robust to many attacks but were less resistant to rotation. However, the performance results of these studies [10,11] against JPEG attacks are not available, and the scaling factors for combining the singular values were determined manually [9–13].

The optimum scaling factor value of each image should be determined in order to strike a balance between the needed high PSNR and normalized correlation (NC) values in watermarking [8]. Optimization methods such as genetic algorithm (GA) [14], artificial bee colony [15], Wang–Landau sampling (WLS) [16], and least-square curve fitting [16] have been suggested for calculating the scaling factor. However, it is well established that these optimization methods have high computational costs. Considering the performances against attacks, especially rotation, it can be said that the optimization burden does not profoundly contribute to the results [16,17].

On the other hand, using the edges in the image as the watermarking regions improves the imperceptibility and robustness of the watermarking scheme. According to the literature, the message embedded in the edge regions is less affected by the distortion caused by the attacks than in the non-edge regions [18]. Traditional edge detection techniques, such as Sobel, Prewitt, and Canny, are insufficient for watermarking, requiring a high payload as they detect a limited number of edge points. As these techniques often find common edge points, it is difficult to reach the desired payload level even if the results are combined [19,20]. The edge-based data hiding in the frequency domain is generally more robust to attacks than the spatial domain techniques. Gong et al. [21] first applied contourlet transform to gray images to increase imperceptibility and robustness and then decomposed low-high (LH) subband coefficients into 4×4 subblocks. The edges of the coefficients in these blocks were determined by the Canny edge detection method, and SVD was applied to the selected blocks using a threshold value. The watermark was combined in the unitary matrix (U) component. In another study [22], the high-frequency band of DWT was divided into blocks, and the gradient values of each block were calculated. Positive gradient peaks were labeled as edges, and negative peaks were labeled as not edges. The edge and non-edge points were used to embed 1 and 0 values of the binary logo. In the study of Zhang et al. [23], the edge image obtained with the Sobel operator in the 2nd level LL band of the gray image was divided into blocks. The edge blocks and logo were first scrambled and then combined with the exclusive or (XOR) process. Kazemi et al. [24] used Zenzo edge detector and CT techniques to embed a binary logo into 512×512 color images. In this study, logo extraction was performed using differential and multilayer perceptron (MLP) techniques while hiding logo information in locations determined by GA. The methods proposed in these studies [21–24] were imperceptible and resistant to attacks except for rotation. In addition, the payload capacities of these studies [21–23] were 1024, 1024, and 3640 bits, respectively. In this context, there are limitations such as low capacity and vulnerability to rotation attacks [21–24].

In the literature, some studies calculate the optimum scaling factor with edge-based watermarking techniques. Mittal et al. [25] first obtained the edge surface image of the original image by using the Gaussian filter and first-order partial differential techniques in the proposed gray image watermarking method. Then, after applying CT to this edge surface image, they watermarked a 64×64 (4096-bit) binary logo on the coarse-level components. In addition, the scaling factor was calculated from the pixel density values of the original image. Although the method was resistant to filtering, rotation, noise, HE, and cropping attacks, its robustness to rescaling and JPEG compression was unclear.

Takore et al. [26] proposed a hybrid technique including LWT, DCT, and SVD techniques for watermarking a 32×32 (1024-bit) logo on 512×512 gray images. The canny method was used to detect high-edge pixels and these edges determined the optimal blocks for embedding the watermark. In another study from the same group [27], the regions for watermarking were first obtained using the canny edge detector. Second, two sub-images of the edge images are created using blocks with a high and low edge number. Finally, the first and second sub-images were used for embedding and extracting the logo. The scaling factor was estimated using particle swarm optimization (PSO) in [26] and GA in [27]. However, these methods have low capacity and are not rotation resistant. The watermarking technique developed by Dhar et al. [28] first divided the cover image into subbands with DWT and then calculated the entropy and edge values of each subband in order to select the appropriate band for watermarking. The scaling factor was determined with the fuzzy logic (FL) method, which used the entropy and edge values as inputs. Although the proposed method had high resistance to cropping, noise, median filtering, and rotation attacks, the payload capacity and JPEG compression performance were unclear. As a result, this study focused on developing a novel edge-based watermarking technique to overcome these limitations.

3 Motivation and Contributions

Our motivation is on the three main limitations faced by the above studies. First, low capacity problem due to the limited number of edge points detected by the traditional edge operators used in edge-based watermarking techniques. The second is to find the optimum value of the scaling factor used in combining the singular values obtained with SVD. Although a common value is used in some studies, this value should change depending on the image content. In addition, the computational cost of the algorithms used to optimize this value is quite high. Finally, the methods mentioned have the problem of insufficient robustness, especially against JPEG compression and rotation attacks. Therefore, in this context, the contributions of this study can be expressed as follows.

- By using a novel edge detection algorithm based on a similarity measure, detecting more edge points and thus providing a solution to the high-capacity requirement.
- Practical and image-based determination of the scaling factor by using the standard deviation, which is the measure of image contrast, and thus obtaining a satisfactory imperceptibility despite using high payload capacity.
- Using a high-frequency subband of DWT to achieve high resistance to JPEG and rotation attacks.

The non-blind watermarking method developed in the study has novelty because it overcomes the capacity, robustness, and imperceptibility problems. We expect the proposed method to be used as an alternative security scheme in the image watermarking field.

4 Material and Methods

In this study, as seen in Fig. 1, a novel image watermarking technique was proposed that enables the secret message to be transformed into a color image with an edge-based method and to be hidden in the cover image with DWT and SVD techniques. Two novel edge detection operators as similarity-based edge detection (VSIME) and horizontal similarity-based edge detection (HSIME) are developed in this study. The development of these algorithms was inspired by the edge detection method proposed by Demirci et al. [29], which is based on the similarity measure of a pixel with its neighbors. First, a binary mask image containing the edge points of the cover image was obtained using VSIME and

HSIME algorithms. In order to use the total payload of the color image, the message was randomly created to have a byte capacity of three times the number of edge pixels in the mask. Security was strengthened by encrypting the message with Advanced Encryption Standard (AES) encryption before the watermarking. According to the edge points in the mask, the bytes of the encrypted message were inserted sequentially into each pixel's red, green, and blue (RGB) channels to create a color message image. In the watermark embedding stage, the sub-bands of the message and cover images for each color channel were decomposed by 2nd-level DWT. SVD was applied to high-high (HH) frequency subbands of both images. The watermark is embedded after being weaked by a scaling factor to provide sufficient imperceptibility in the watermarking process. Although the LL band is widely used in existing studies, the change in the image caused by the embedding process can be impercepted more easily. While the use of HH and LL bands provides robustness against compression, filtering and scaling attacks [30], the margin provided by the HH band in terms of imperceptibility gives the opportunity to hide less attenuated watermark [31]. In addition, high-frequency regions are selected since an edge-based algorithm detects the hiding location. Therefore, the use of the HH band is compatible with the proposed watermarking method [30,31]. After combining the singular values with an adaptive scaling factor (α), the watermarked image was created by using first inverse SVD and then inverse DWT.

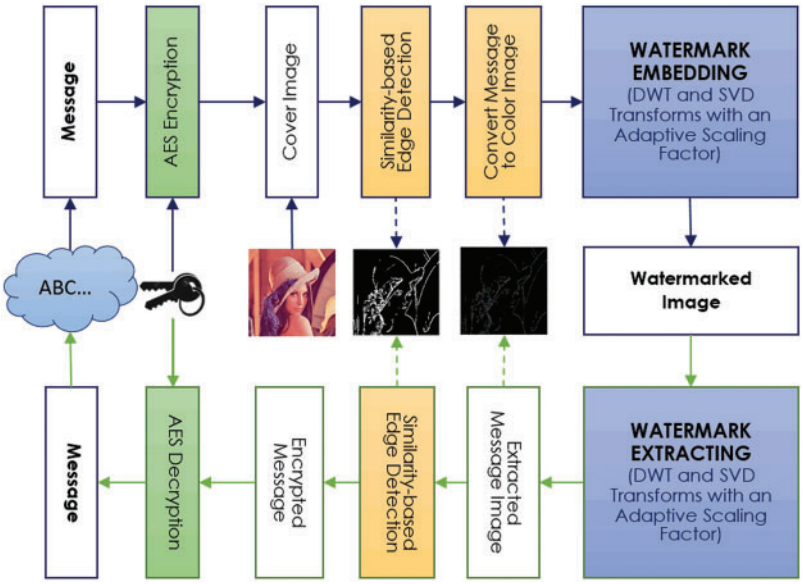

Figure 1: Flowchart of proposed image watermarking scheme

In the watermark extracting phase, cover and watermarked images were decomposed into sub-bands using DWT. SVD was applied to HH bands of images. The colored-message image was calculated using the singular values of HH bands and α coefficient. The mask image containing the edges of the cover image was also obtained using the similarity-based edge detection methods. The pixels of the colored-message image of which the locations were taken from the mask image are first converted to byte-array and then to message by AES decryption. In this study, the performance of the developed method was measured using PSNR and structural similarity index measure (SSIM) metrics among the cover and watermarked images [1]. The robustness analysis was performed using NC and Bit Error Rate (BER). Four color images (Airplane, Mandrill, Peppers, and Lena) with 512×512 size from the Misc-Dataset (http://sipi.usc.edu/database/database.php?volume=misc) were

used in our proposed watermarking scheme. The coding and test analysis of the proposed method were implemented in MATLAB.

The algorithm steps of the proposed color image watermarking method are as follows:

1- Get the color image (I) and obtain the edge mask and scaling factor(α) using the SIME (VSIME or HSIME) algorithm

$(I_{edge}, I_{sim}) \leftarrow SIME(I)$

$\alpha = 0.15 + std(I_{sim})$

2- Generate random text with byte capacity of the number of edge pixels and encrypt text with AES

$M \leftarrow random(length(I_{edge}))$

$M_e \leftarrow AES\_encryption(M, public\_key1, private\_key2)$

3- Obtain the colored-message image (I_W) by inserting the message sequentially into the RGB channels of the edge points

$I_W \leftarrow MessagetoImage(M_e)$

4- Obtain LL, LH, HL, and HH sub-bands by applying DWT to I and I_W images

$[LL_I, LH_I, HL_I, HH_I] \leftarrow DWT(I), [LL_{Iw}, LH_{Iw}, HL_{Iw}, HH_{Iw}] \leftarrow DWT(I_W)$

5- Apply SVD for each color channel in HH sub-bands

$SVD(HH_I) = U_{HH_I} \times S_{HH_I} \times V^T_{LL_I}, SVD\left(HH_{I_W}\right) = U_{HH_{I_W}} \times S_{HH_{I_W}} \times V^T_{HH_{I_W}}$

6- α for each color channel

$S_{HH_M} = S_{HH_I} + \alpha \times S_{HH_{I_W}}$

7- Obtain HH band using S_{HH_M}

$HH_{SVD} = U_{HH_I} \times S_{HH_M} \times V^T_{HH_I}$

8- Obtain watermarked image (W) using inverse DWT

$W = IDWT(LL_I, LH_I, HL_I, HH_{SVD})$

The pseudo-code for extracting the watermark is as follows.

1- Get the watermarked image (W) and cover image (I)

2- Find the edge mask using the SIME algorithm

$(I_{edge}, I_{sim}) \leftarrow SIME(I)$

$\alpha = 0.15 + std(I_{sim})$

3- Obtain LL, LH, HL, and HH sub-bands by applying DWT to I and W images

$[LL_I, LH_I, HL_I, HH_I] \leftarrow DWT(I), [LL_W, LH_W, HL_W, HH_W] \leftarrow DWT(W)$

4- Apply SVD for each color channel in HH sub-bands

$SVD(HH_I) = U_{HH_I} \times S_{HH_I} \times V^{T_I}_{HH}, \; SVD(HH_W) = U_{HH_W} \times S_{HH_W} \times V^T_{HH_W}$

5- Calculate S values of the colored-message image using S values of HH subbands and α for each color channel

$S_{HH_M} = (S_{HH_I} - S_{HH_W})/\alpha$

6- Obtain HH band using S_{HH_M}

$$HH_{SVD} = U_{HH_{I_W}} \times S_{HH_M} \times V^T_{HH_{I_W}}$$

7- *Obtain colored-message image* (I_W) *using inverse DWT*

$$I_W = IDWT(LL_W, LH_W, HL_W, HH_{SVD})$$

8- *Obtain the encrypted message from the color channels of the pixels in the image*

$$M_e = ImagetoMessage(I_W, I_{edge})$$

9- *Decrypt the encrypted message using AES*

$$M \leftarrow AES\_decryption(M_e, public\_key1, private\_key2)$$

4.1 Similarity-Based Edge Detection Method

Edge refers to the boundary between an object and background or the boundary between overlapping objects. Edge detection aims to find regions in an image that have a sharp change in intensity [32,33]. Although different edge detection algorithms usually detect the same edge pixels, there are also regions where they produce different results. The derivative operator used in edge detection basically calculates the changes in the gray level, and the amount of change close to an edge is much bigger than the ones in a slowly varying intensity area. However, the change needs to be evaluated in many directions in a two-dimensional image. Therefore, in gradient-based edge detection operators, partial derivatives of the image in vertical and horizontal directions are taken. While gradient-based edge detection methods such as Sobel, Prewitt, and Canny have been used in practice, the Canny edge detector is often preferred as an optimal edge detection method.

In the canny edge detection method, Gaussian smoothing, finding local maxima, and filtering false positive edge points using two threshold values are steps to obtain a strong edge map. On the other hand, these steps cause the side-effect of eliminating a significant number of the real edge points, which can be used in embedding a watermark. For this reason, in this study, the similarity-based edge detection (SIME) method was adapted to obtain vertical and horizontal edges by decreasing the noise of the image without attenuating the edge region [29]. It uses the difference of adjacent pixel values in the image. The similarity measure of two pixels is calculated with Euclidean distance in color space to obtain a similarity degree to determine the edges and non-edges pixels in an image [29]. The pixels in color images have three component intensities red (R), green (G), and blue (B). The SIME method [29] maps three color channels into a single channel.

In this study, a novel edge detection approach was proposed to find vertical and horizontal edges, inspired by the SIME method. For VSIME and HSIME, similarity values are calculated in the horizontal and vertical directions, respectively. First, a 3×3 window is slided over the cover image. At a certain position of the 3×3 sliding window in the image, the pixel value of the edge image, whose position is the center of the window, is calculated by the average of the similarity matrix (S) values in Fig. 2b. The values of the S matrix are binarized using a certain threshold value. While calculating the elements of this S matrix, the relation network in Fig. 2a is used. Here, P_i, P_{i+1}, and P_{i+2} are pixels in a particular row in the window region of the cover image. The similarity values are calculated for each pixel with the other two pixels. The similarity calculation is performed using Eqs. (1)–(3) inspired from [29]. P_x in Eqs. (1) and (2) refers to the reference pixel whose similarity with the other two pixels is calculated. Therefore, since the similarity with the other two pixels is calculated for each pixel in a specific row, a total of three similarity values are obtained. Considering 3 rows in the window, a total of 9 similarity values at a certain location of the window are found as the elements of the S matrix shown in Fig. 2b. For example, S_1 is calculated using two similarity values. One is the similarity between P_1 and P_2, and the other is the similarity between P_1 and P_3. Similarly, S_2 is obtained from two similarity

values between P_2 and P_1, as well as between P_2 and P_3. For example, when calculating the values S_1, S_2, and S_3 for the first row, P_x in Eqs. (1)–(2) are matched to points P_1, P_2, and P_3, respectively. The other two points that do not match the P_x reference point are assigned to P_y and P_z.

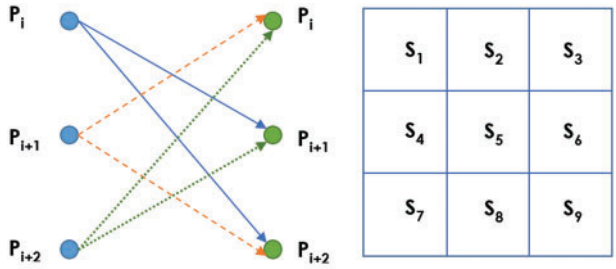

Figure 2: a) Similarity relation network, b) similarity (S) matrix

$$S_x = \begin{cases} 1, \ if \ f\left(P_x, P_y, P_z\right) > T_h \\ 0, otherwise \end{cases} \tag{1}$$

$$f\left(P_x, P_y, P_z\right) = \exp\left(\frac{-\left(d_{P_x P_y} + d_{P_x P_z}\right)}{D_n}\right) \tag{2}$$

$$d_{P_i P_j} = \frac{1}{\sqrt{3}}\left(\Delta R^2 + \Delta G^2 + \Delta B^2\right)^{1/2} \tag{3}$$

$$\Delta R = \left|P_{i(R)} - P_{j(R)}\right|$$
$$\Delta G = \left|P_{i(G)} - P_{j(G)}\right|$$
$$\Delta B = \left|P_{i(B)} - P_{j(B)}\right|$$

where the threshold value (T_h) in Eq. (1) and the normalization coefficient (D_n) in Eq. (2). In an 8-bit image with a maximum intensity value of 255, the normalization can be performed by choosing a D_n value of up to 510 since two dissimilarity values are summed in the numerator of Eq. (2). $P_{i(R)}$ and $P_{j(R)}$ are the intensity values of red channel for P_i and P_j pixels. ΔR, ΔG, and ΔB in Eq. (3) are the gray level differences between two pixels in red, green, and blue channels.

Eq. (4) is repeated for all pixels to create a similar image from the color image whose values range between 0 and 1 [29].

$$I_{sim}(r, c) = \frac{1}{9}\sum_{n=1}^{9} S_n \tag{4}$$

In this study, the edge mask of the color-similar image (I_{edge}) was found by using Eq. (5) [29].

$$I_{sim\_binary}(r, c) = \begin{cases} 1 \text{ if } I_{sim}(\mathrm{r}, \mathrm{c}) \geq S_{Th}, \\ 0 \text{ if } I_{sim}(\mathrm{r}, \mathrm{c}) < S_{Th} \end{cases} \tag{5}$$

$$I_{edge} = 1 - I_{sim\_binary}$$

where $I_{sim\_binary}(r, c)$ is the pixel whose similarity value is calculated and matched with the center of the sliding window. S_{Th} is the threshold value that determines whether the center pixel is a similarity pixel or not. In this study, we assumed that the S_{Th} value should be greater than or equal to 0.5 when deciding on the similarity of a pixel. The selection of T_h and D_n values affects the number of detected edge points [29]. As the D_n value in the denominator in Eq. (2) increases, higher similarity values are obtained as the exponential curve decrease faster. Therefore, capturing more similar points results in fewer edge points. As seen in Fig. 3, while the number of edges decreases as D_n increases, the increase in the T_h increases the number of edges for a given D_n. On the other hand, very high T_h selection causes fake edge points. Choosing high D_n results in fewer edge points. In this study, D_n and T_h values, which keep a high imperceptibility and provide sufficient payload capacity, were determined as 32 and 0.5, respectively.

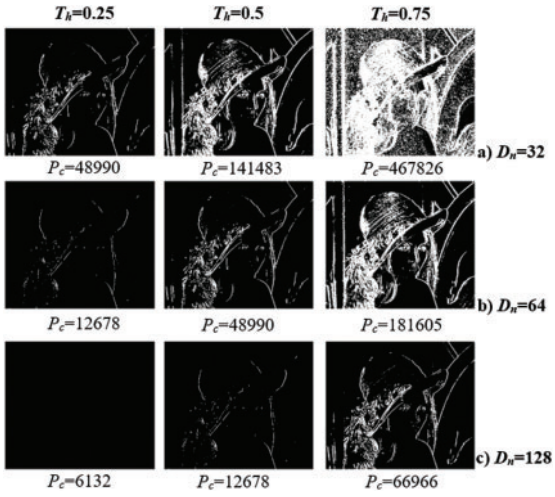

Figure 3: Effects of D_n and T_h on payload capacity (P_c) in VSIME for $S_{Th} = 0.5$, where P_c is the number of edge pixels. The results are given for the $512 \times 512 \times 3$ Lena image

A vertical edge image is generated as a result of these calculations. For the horizontal edge image, the same calculations are performed by taking the transpose of the matrix containing the image pixels in the 3×3 sliding window position.

By using these two edge detection approaches (HSIME and VSIME), the edge masks of the cover images were obtained, and the high-capacity message was inserted sequentially in the RGB channels of the edge pixels. Fig. 4 shows the similarity images, thresholded similarity images (I_{sim}), and edge masks (I_{edge}) of the Lena images for the HSIME and VSIME methods, respectively. The edge masks of the Canny, Sobel, and Prewitt methods are also illustrated in Fig. 4. The numbers of edge pixels obtained from the Canny, Sobel, Prewitt, HSIME, and VSIME techniques for the color Lena image are 22527, 8314, 8242, 141483, and 86697, respectively.

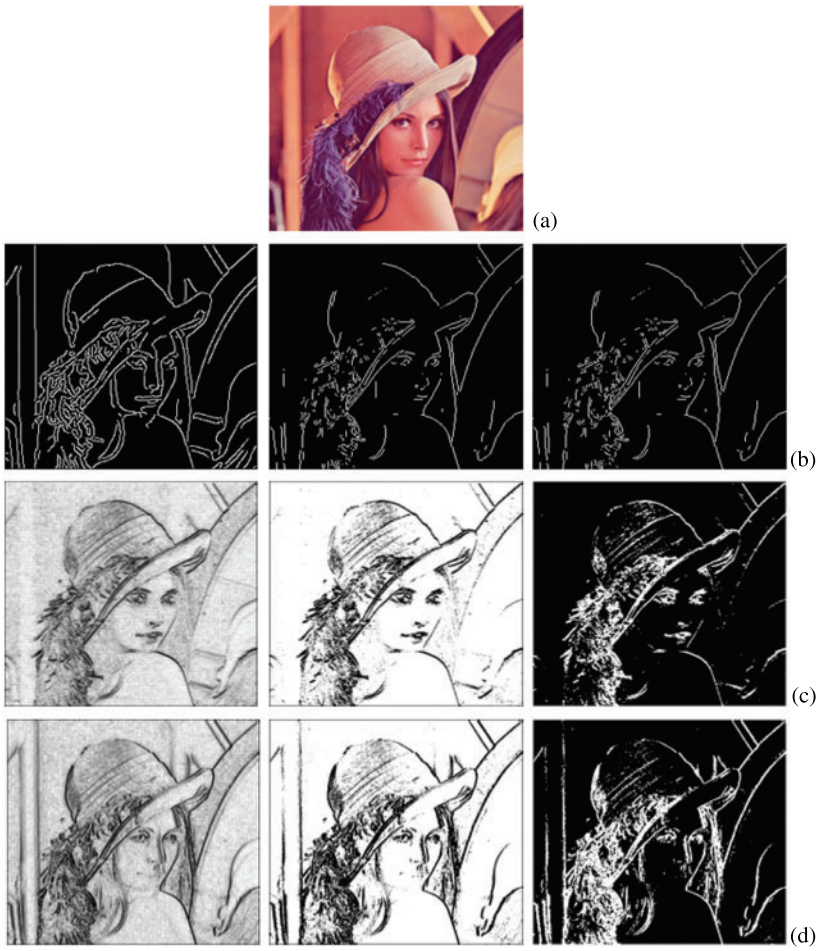

Figure 4: a) Cover image (Lena), b) Edge masks of Canny, Sobel, and Prewitt methods from left to right, c) Similarity image, thresholded similarity image and edge mask of HSIME method, d) Similarity image, thresholded similarity image and edge mask of VSIME method

4.2 Discrete Wavelet Transform

In discrete wavelet transform (DWT), a signal is decomposed into wavelets based on a selected wavelet signal (such as Haar wavelet and Daubechies set wavelets) rather than frequencies, and DWT provides the time-frequency representation of the signal. The basic idea in wavelet transform is to use a set of essential functions (called wavelets) that allow the locations of the frequencies in the signal to be presented as well. In the DWT application of two-dimensional (2D) image, spatial location information is obtained instead of time. In DWT, the high pass filter H gives the detail coefficients of the image, and the low pass filter L gives the approximate coefficients. The decomposition stage is repeated up to several levels. With the one-level decomposition of 2D DWT, it creates four sub-bands, low-low (LL), low-high (LH), high-low (HL), and high-high (HH). The LL sub-band is half the size of the image, and most information for image reconstruction is obtained from this band. Other LH, HL, and HH sub-bands give the image's vertical, horizontal, and diagonal details, respectively. The

DWT of an image $f(x, y)$ with $M \times N$ dimensions is calculated using Eqs. (6)–(7) [34].

$$W_{\varphi}(j_0, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\, \varphi_{j_0,m,n}(x, y) \tag{6}$$

$$W_{\psi}^{i}(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\, \psi_{j_0,m,n}^{i}(x, y) \tag{7}$$

where $W_{\varphi}(j_0, m, n)$ is the approximate coefficients obtained for the initial scale j_0, and $W_{\psi}^{i}(j, m, n)$ is the horizontal, vertical, and diagonal detail of the image for $j > j_0$ [34].

In this study, LL, LH, HL, and HH sub-bands of cover and colored-message images were obtained with second-level 2D DWT. Data hiding was realized by applying SVD to the approximate coefficients of HH subbands.

4.3 Singular Value Decomposition

SVD is widely used in statistical analysis and digital signal processing. SVD is the transformation of a square matrix into a real or complex matrix of *mxn* size by factoring it through polar decomposition. SVD is carried out with Eq. (8) [8].

$$M = U \Sigma V^{*} \tag{8}$$

where M is the real or complex matrix with $m \times n$ dimensions. If M is a complex matrix, U and V are complex unit matrices with dimensions $m \times m$ and $n \times n$ respectively. Σ is a rectangular diagonal matrix of dimensions $m \times n$ and contains non-negative real numbers in its diagonal. If M is a real matrix, then U and $V^{T} = V^{*}$ are real and orthogonal matrices, respectively. The diagonal values ($\sigma_{i} = \Sigma_{ii}$) of the matrix Σ are also singular values of the M matrix. According to the literature, hiding data to singular values obtained by SVD protects the message against attacks and ensures imperceptibility [3]. In this study, the singular values of the cover and colored-message images were used for data hiding.

The second level DWT of the cover (I) and message (M) images were used to get the diagonal HH_I and HH_M coefficients, respectively. Then the singular values of the HH subbands were obtained separately for each color channel by using Eq. (8), and these values (Σ_I and Σ_M) were concatenated using Eq. (9) [8].

$$\Sigma_W = \Sigma_I + (\alpha * \Sigma_M) \tag{9}$$

where the α value is a scaling factor. In this study, it was adaptively calculated by adding a bias value to the standard deviation, which is a metric of image contrast from Eq. (10).

$$\alpha = bias + std(I_{sim}) = 0.15 + \sqrt{\frac{1}{M \times N} \sum_{i=1}^{N} \sum_{j=1}^{M} (I_{sim}(i, j) - \mu)^2} \tag{10}$$

where μ is the mean of the thresholded similarity image (I_{sim}). The proposed watermarking scheme obtained the HH_I subband of the watermarked cover image by performing inverse SVD. Finally, the watermarked image was created with inverse DWT using subbands of the cover image (LL_I, HL_I, LH_I, and HH_I).

5 Experimental Results and Discussion

In this study, image watermarking was performed on four color images using two similarity-based edge methods: VSIME and HSIME. The capacity, imperceptibility, and robustness analyses of the developed methods were measured using the test images.

5.1 Capacity and Imperceptibility Analyses

The Lena, Peppers, Airplane, and Mandrill test images and the encrypted message inserted in the RGB channels of the edge pixels in the edge mask obtained by the VSIME and HSIME methods are given in Figs. 5b–5c, respectively.

Figure 5: a) Original test images, b) Message-watermark images obtained using vertical edges, c) Message-watermark images obtained using horizontal edges

The scaling factor (α) used in the watermarking is crucial in terms of imperceptibility and robustness. A low α value results in poor robustness performance, while a high α value yields low imperceptibility. In this context, we found $\alpha = 0.15$ to guarantee an NC value of 0.75 in the DWT + SVD watermarking method, since the most fragile NC value occurred in JPEG compression attacks (Table 1). As seen in Tables 1 and 2, when using only the standard deviation for α, we reached both acceptable imperceptibility and higher robustness compared to $\alpha = 0.15$. It is shown in Table 2 that the α value, which allows for keeping imperceptibility above 35 dB in the test images, should be less than 0.5. As a result, the scaling factor calculated using the bias + standard deviation given in Eq. (10) provided a watermarking with high robustness and acceptable imperceptibility. For the scaling factor analysis, only the VSIME results are shown in Tables 1 and 2, as VSIME and HSIME analyses produce

similar outputs. In Table 1, the robustness against JPEG compression attacks, in which the DWT + SVD method is the most fragile, is given together with some other attack results.

Table 1: NC values of the VSIME method for various attacks related to $\alpha = 0.15$, 0.5, and std (I_{sim})

Attack type	Airplane			Mandrill			Peppers			Lena			Mean		
	0.15	std[a]	0.5	0.15	std[a]	0.5	0.15	std[a]	0.5	0.15	std[a]	0.5	0.15	std[a]	0.5
No-Attack	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Median filtering (3×3)	0.97	0.98	0.99	0.92	0.97	0.98	0.97	0.98	0.99	0.97	0.98	0.99	0.96	0.98	0.99
Average filtering (3×3)	0.93	0.96	0.98	0.88	0.95	0.97	0.94	0.97	0.98	0.93	0.97	0.98	0.92	0.96	0.98
Gaussian noise (0.05)	0.79	0.89	0.97	0.95	0.99	1.00	0.80	0.90	0.97	0.80	0.90	0.97	0.84	0.92	0.98
Sharpening (0.2)	1.00	1.00	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Motion blur	0.87	0.93	0.96	0.83	0.93	0.96	0.89	0.94	0.97	0.86	0.93	0.96	0.86	0.93	0.96
Rotation (10°)	1.00	1.00	0.99	0.99	0.99	0.99	1.00	0.99	0.99	0.99	0.99	0.99	1.00	0.99	0.99
JPEG-2000 (CR[b] = 2)	**0.76**	0.87	0.94	**0.75**	0.88	0.94	**0.75**	0.82	0.90	**0.78**	0.89	0.95	0.76	0.87	0.93
JPEG(QF[c] = 10)	**0.77**	0.88	0.95	**0.75**	0.88	0.94	0.93	0.96	0.98	**0.78**	0.89	0.95	0.81	0.90	0.96
Mean	**0.90**	**0.95**	**0.98**	**0.89**	**0.95**	**0.97**	**0.92**	**0.95**	**0.98**	**0.90**	**0.95**	**0.98**	**0.90**	**0.95**	**0.98**

Note: [a] std: standard deviation of similar image (std(I_{sim})), [b] CR: compression ratio, [c] QF: quality factor.

Table 2: Imperceptibility analysis of VSIME based method for various scaling factors

Test ımage	α = std(I_{sim})∗	PSNR for various α			
		0.15	std(I_{sim})	0.15 + std(I_{sim})	0.5
Airplane	0.2568	50.7768	46.1263	**42.2574**	40.5269
Mandrill	0.2842	46.1018	40.5333	**36.9737**	**35.5425**
Peppers	0.2649	50.2920	45.5188	**41.5392**	39.4003
Lena	0.2820	50.5043	45.4471	**41.3865**	40.1677
Mean	0.2720	49.4187	44.4064	**40.5392**	38.9094

Note: ∗std(I_{sim}): standard deviation of similar image.

In Table 3, the payload capacities and PSNR and SSIM performance measures of VSIME and HSIME techniques using adaptive scaling factors (bias + standard deviation) are shown. Since we use all edge points in message embedding, the number of edges also refers to the total payload capacity. In addition, the ratio of the payload (bpp: bit per pixel) of the test images is given. According to the mean values in Table 3, the payload capacity of the VSIME technique is higher than the HSIME, and the PSNR and SSIM values of both methods are close the each other. The imperceptibility depends on the distortion created by the watermark signal in the cover image after watermarking. As the capacity of the secret message increases, the PSNR and SSIM metrics, which measure imperceptibility, decrease simultaneously. The Airplane image had the highest PSNR value in the study, while the Mandrill image had the lowest. As seen from Table 3, the image was more distorted because of the extremely high payload of the Mandrill. In general, low PSNR values would be expected at such a high payload

rate. The adaptive scaling factor, which is dependent on each test image contrast value, avoided further degradation of the performance measures.

Table 3: Payload capacities and performance comparison of edge detection-based methods

Image label	VSIME					HSIME				
	Number of edges	Payload (bpp)	PSNR	SSIM	Scaling factor (α)	Number of edges	Payload (bpp)	PSNR	SSIM	Scaling factor (α)
Airplane	113868	1.1580	42.2574	0.9953	0.4068	121389	1.2350	42.4214	0.9954	0.4123
Mandrill	577758	5.8770	36.9737	0.9946	0.4342	575529	5.8550	36.4170	0.9939	0.4450
Peppers	140505	1.4290	41.5392	0.9985	0.4149	121710	1.2380	39.7405	0.9977	0.4989
Lena	141483	1.4390	41.3865	0.9982	0.4320	86697	0.8820	41.7519	0.9984	0.4641
Mean	243404	2.4758	40.5392	0.9967	0.4220	226331	2.3025	40.0827	0.9964	0.4551

5.2 Robustness Analyses

In this study, various attacks were performed to determine the robustness of the proposed VSIME and HSIME edge detection-based watermarking. Table 4 shows the NC values obtained against noise and filtering attacks performed for the four test images. Gaussian, Salt and Pepper, and Speckle noise attacks were carried out using different variance values such as 0.05, 0.01, 0.001, and 0.005. Median and average filtering on watermarked images were applied using 3×3 and 5×5 window sizes. In the Gaussian low pass filtering (LPF) attack, 0.1 and 0.5 standard deviation values were used for each of the 3×3 and 5×5 window sizes. Both proposed edge detection techniques have obtained high resistance values against Median, Average, and Gaussian LPF attacks. In particular, the best robustness values were obtained against Gaussian LPF attacks. In addition, high NC values were found against noise attacks. Both developed methods had the most significant resistance to Salt and Pepper attacks. As expected, the larger the window size and the greater the noise variance, the greater the loss of robustness.

Table 4: NC values of the VSIME and HSIME methods for the filtering and noise attacks

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
Median filtering	3×3	0.9885	0.9884	0.9801	0.9801	0.9890	0.9898	0.9891	0.9868	0.9867	0.9862
	5×5	0.9694	0.9689	0.9527	0.9528	0.9756	0.9769	0.9709	0.9642	0.9671	0.9657
	Mean	0.9790	0.9787	0.9664	0.9665	0.9823	0.9834	0.9800	0.9755	0.9769	0.9760
Average filtering	3×3	0.9771	0.9767	0.9713	0.9713	0.9794	0.9815	0.9787	0.9739	0.9766	0.9759
	5×5	0.9497	0.9488	0.9461	0.9464	0.9587	0.9634	0.9540	0.9430	0.9521	0.9504
	Mean	0.9634	0.9628	0.9587	0.9589	0.9691	0.9725	0.9664	0.9585	0.9644	0.9632
Gaussian LPF	3×3-$\sigma = 0.1$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	3×3-$\sigma = 0.5$	0.9966	0.9965	0.9954	0.9954	0.9968	0.9972	0.9968	0.9961	0.9964	0.9963
	5×5-$\sigma = 0.1$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	5x5-$\sigma = 0.5$	0.9966	0.9965	0.9954	0.9954	0.9968	0.9971	0.9968	0.9961	0.9964	0.9963
	Mean	0.9983	0.9983	0.9977	0.9977	0.9984	0.9986	0.9984	0.9981	0.9982	0.9982
Gaussian noise (σ^2)	0.05	0.9482	0.9466	0.9950	0.9951	0.9568	0.9718	0.9588	0.9499	0.9647	0.9658
	0.01	0.9756	0.9746	0.9982	0.9984	0.9822	0.9889	0.9828	0.9776	0.9847	0.9849

(Continued)

Table 4: Continued

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
	0.001	0.9993	0.9992	1.0000	1.0000	0.9997	0.9998	0.9995	0.9994	0.9996	0.9996
	0.005	0.9906	0.9899	0.9995	0.9995	0.9934	0.9963	0.9934	0.9921	0.9942	0.9944
	Mean	0.9784	0.9776	0.9982	0.9983	0.9830	0.9892	0.9836	0.9798	0.9858	0.9862
Salt and	0.05	0.9966	0.9966	0.9998	0.9998	0.9976	0.9986	0.9977	0.9971	0.9979	0.9980
peppers noise	0.01	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
(σ^2)	0.001	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	0.005	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Mean	0.9992	0.9992	1.0000	1.0000	0.9994	0.9997	0.9994	0.9993	0.9995	0.9995
Speckle noise	0.05	0.9323	0.9295	0.9970	0.9973	0.9729	0.9827	0.9728	0.9666	0.9688	0.9690
(σ^2)	0.01	0.9892	0.9883	0.9998	0.9998	0.9974	0.9985	0.9967	0.9959	0.9958	0.9956
	0.001	0.9997	0.9997	1.0000	1.0000	1.0000	1.0000	0.9999	0.9999	0.9999	0.9999
	0.005	0.9960	0.9960	0.9999	0.9999	0.9992	0.9996	0.9989	0.9986	0.9985	0.9985
	Mean	0.9793	0.9784	0.9992	0.9993	0.9924	0.9952	0.9921	0.9903	0.9908	0.9908

Note: $*\sigma$: standard deviation, σ^2: variance.

In our edge detection methods, there is no significant relation between payload capacity and robustness against filtering and noise attacks, as seen in Tables 3 and 4. For example, although Mandrill has the highest payload, its robustness is very close to other images. Similarly, Lena has the lowest horizontal payload, but its robustness is not more significant than the other test images.

Table 5 shows the NC values obtained against sharpening, gamma correction, motion blur, and HE attacks. Although the average robustness values against these attacks in both edge-based watermarking were very close, the results for VSIME were slightly higher. As shown in Table 5, both approaches were relatively more resistant to sharpening and HE attacks, resulting in high NC values. The changes in the sharpening attack's strength value and the gamma correction attack's gamma value did not affect the robustness.

Table 5: NC values of the VSIME and HSIME methods for the attacks

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
Sharpening	0.2	0.9990	0.9990	0.9989	0.9989	0.9996	0.9997	0.9992	0.9990	0.9992	0.9992
(strength)	0.6	0.9919	0.9918	0.9910	0.9911	0.9964	0.9969	0.9931	0.9920	0.9931	0.9929
	Mean	0.9955	0.9954	0.9950	0.9950	0.9980	0.9983	0.9962	0.9955	0.9962	0.9961
Gamma	0.2	0.9190	0.9172	0.9285	0.9292	0.9369	0.9458	0.9293	0.9118	0.9284	0.9260
Correction	0.6	0.9190	0.9172	0.9285	0.9292	0.9366	0.9456	0.9293	0.9118	0.9283	0.9259
(gamma)	Mean	0.9190	0.9172	0.9285	0.9292	0.9368	0.9457	0.9293	0.9118	0.9284	0.9260
Motion Blur		0.9543	0.9535	0.9560	0.9561	0.9623	0.9664	0.9565	0.9460	0.9573	0.9555
HE*		0.9448	0.9432	0.9911	0.9913	0.9990	0.9991	0.9931	0.9919	0.9820	0.9814

Note: *HE: Histogram Equalization.

Table 6 shows the NC values obtained against the rescaling, cropping, and rotation attacks. Both edge-based watermarking methods have achieved high resistance values against these attacks. In the rescaling operation, in the representation of x $\rightarrow$ 512, x denotes the size in which the original image was first converted, then the image was resized back to 512. In rescaling attacks, the resistance to down-scaling was lower than the up-scaling. Resistance to rotational attacks was very high, with

no significant difference depending on the angle. Each subband of DWT was tested against various attacks. It was concluded from the results that the high-frequency band provides a considerable improvement against rotation and other attacks.

Table 6: NC values of the VSIME and HSIME methods for the attacks

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
Rescaling	1024→512	0.9992	0.9992	0.9987	0.9987	0.9991	0.9992	0.9992	0.9990	0.9991	0.9990
	256→512	0.9874	0.9872	0.9824	0.9824	0.9881	0.9892	0.9882	0.9856	0.9865	0.9861
	128→512	0.9370	0.9357	0.9364	0.9368	0.9501	0.9563	0.9436	0.9299	0.9417	0.9397
	Mean	0.9745	0.9740	0.9725	0.9726	0.9791	0.9816	0.9770	0.9715	0.9758	0.9749
Crop	1/16	0.9995	0.9995	0.9999	0.9999	0.9999	0.9999	1.0000	1.0000	0.9998	0.9998
	1/4	0.9960	0.9961	0.9983	0.9984	0.9985	0.9986	0.9998	0.9997	0.9982	0.9982
	Mean	0.9977	0.9978	0.9991	0.9991	0.9992	0.9992	0.9999	0.9998	0.9990	0.9990
Rotation (Angle)	2	0.9922	0.9922	0.9945	0.9940	0.9918	0.9914	0.9923	0.9908	0.9927	0.9921
	10	0.9951	0.9952	0.9931	0.9927	0.9928	0.9921	0.9928	0.9915	0.9935	0.9929
	30	0.9978	0.9980	0.9918	0.9913	0.9948	0.9935	0.9910	0.9892	0.9939	0.9930
	45	0.9990	0.9990	0.9920	0.9914	0.9979	0.9966	0.9944	0.9933	0.9958	0.9951
	60	0.9978	0.9980	0.9916	0.9910	0.9958	0.9946	0.9918	0.9902	0.9942	0.9934
	90	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Mean	0.9970	0.9971	0.9938	0.9934	0.9955	0.9947	0.9937	0.9925	0.9950	0.9944

Table 7 shows the NC values obtained against JPEG-2000 and JPEG compression attacks. Although the resistance values of both edge-based watermarking methods against these attacks are close to each other, HSIME and VSIME have achieved a slightly better results against JPEG-2000 and JPEG compression attacks, respectively. The lowest robustness values in JPEG-2000 attacks were found for the Peppers image. As the quality factor (QF) increased in JPEG compression attacks, resistance was not significantly decreased in all images except Peppers. In subband selection trials for watermarking, it was determined that using the HH band increased the resistance to JPEG attack, as in rotation.

Table 7: NC values of the VSIME and HSIME methods for the compression attacks

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
JPEG-2000 (CR∗)	2	0.9258	0.9238	0.9325	0.9332	0.8759	0.8953	0.9410	0.9260	0.9188	0.9196
	4	0.9258	0.9238	0.9325	0.9332	0.8759	0.8953	0.9410	0.9260	0.9188	0.9196
	8	0.9258	0.9238	0.9325	0.9332	0.8773	0.8965	0.9410	0.9260	0.9191	0.9199
	Mean	0.9258	0.9238	0.9325	0.9332	0.8764	0.8957	0.9410	0.9260	0.9189	0.9197
JPEG Compression (QF∗)	10	0.9298	0.9262	0.9304	0.9311	0.9791	0.9847	0.9393	0.9230	0.9447	0.9413
	30	0.9302	0.9280	0.9311	0.9318	0.9596	0.9688	0.9446	0.9303	0.9414	0.9397
	50	0.9311	0.9285	0.9316	0.9323	0.9468	0.9580	0.9486	0.9353	0.9395	0.9385
	70	0.9312	0.9292	0.9318	0.9326	0.9394	0.9525	0.9493	0.9359	0.9379	0.9375

(Continued)

Table 7: Continued

Attack type		Airplane		Mandrill		Peppers		Lena		Average	
		VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME	VSIME	HSIME
	90	0.9310	0.9289	0.9322	0.9330	0.9354	0.9486	0.9491	0.9356	0.9369	0.9365
	Mean	0.9307	0.9282	0.9314	0.9322	0.9521	0.9625	0.9462	0.9320	0.9401	0.9387

Note: ∗CR: compression ratio, QF: quality factor.

When the capacity, imperceptibility, and robustness results obtained by the two edge-based watermarking methods proposed in the study are compared, it can be said that the VSIME based watermarking is partially prominent, although the results are close. In Fig. 6, the watermarked images obtained with the VSIME-based watermarking technique and the results of the extracted images after various attacks are shown. BER and PSNR values were calculated between the watermark and extracted watermark. As seen in Fig. 6, BER and PSNR values were obtained in the range of 0–0.044 and 35.64–Inf dB, respectively, after the attack. The robustness NC values given in the tables, PSNR, and BER values shown in Fig. 6 indicated the resistance to attacks of the developed method.

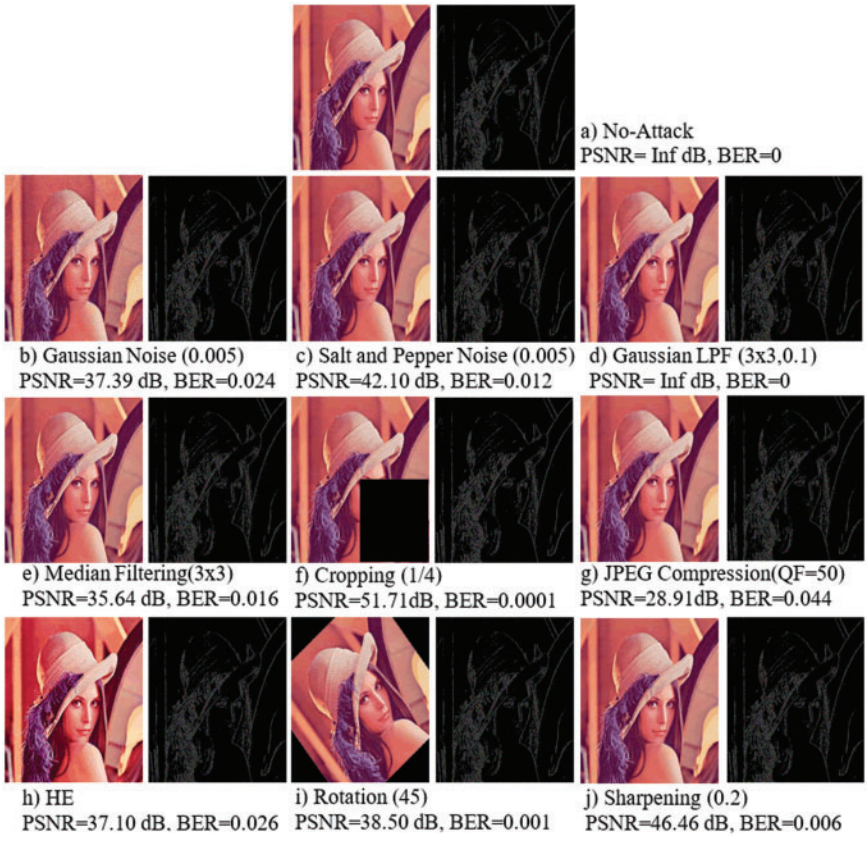

Figure 6: Watermarked imaged and extracted images after various attacks

5.3 Literature Comparison and Discussion

In this study, the proposed watermarking scheme uses edge-based, a high-capacity colored watermark, and an adaptive scaling factor. Therefore, the literature comparison was made in terms of these contexts. As a result of the literature review, the studies in Table 8 were selected, and comparisons were made for the Lena image. Among these studies, [21,23], and [24] were chosen because they are edge-based, [35,36], and [37] perform high-capacity color watermarking, and [16,26,11], and [38] calculate the scaling factor adaptively. As seen in Table 8, while the watermark capacity increases, the imperceptibility performances decrease. Prabha et al. [35] used Walsh Hadamard Transform (WHT) to hide a high-capacity color watermark by less distorting the cover image; however, this method found low robustness to attacks. In our proposed method, high robustness values were obtained with high-capacity watermarking.

Table 8: Comparison between literature and proposed VSIME and HSIME-based watermarking methods

Parameters	Method	Comparison Reason	Image type	Watermark type	Watermark size	Security	PSNR (dB)	SSIM
Gong et al. [21]	CT[a] + Canny + SVD	Edge-based	Gray	Binary	1024 bits	-	45.02	0.986
Zhang et al. [23]	Sobel + DWT	Edge-based	Gray	Binary	3640 bits	XOR	-	-
Kazemi et al. [24]	Zenzo + CT[a]	Edge-based	Color	Binary	-	Arnold	48.81	-
Prabha et al. [35]	WHT	High-capacity	Color	Color	24300 bytes	-	49.21	0.995
Kathpal et al. [36]	DWT + SVD	High-capacity	Color	Gray	262144 bytes	-	36.67	-
Luo et al. [37]	Interblock + AME[b] + CS[c]	High-capacity	Color	Color	24576 bits	CS[c]	48.56	0.999
Wang et al. [16]	DWT + SVD + WLS	High-capacity, Scaling factor	Gray	Binary	16384 bits	-	40.74	1.000
Takore et al. [26]	LWT + DCT + SVD	Scaling factor	Gray	Binary	1024 bits	-	45.51	-
Begum et al. [11]	DCT + DWT + SVD	Scaling factor	Gray	Binary	4096 bits	Arnold	57.63	0.998
Gao et al. [38]	DWT + SVD	Scaling factor	Gray	Binary	4096 bits	Arnold	33.77	-
Proposed	**VSIME + DWT + SVD**	**Edge-based, High-capacity, Scaling factor**	**Color**	**Color**	**141483 bytes**	**AES**	**41.39**	**0.998**
	HSIME + DWT + SVD		**Color**	**Color**	**86697 bytes**	**AES**	**41.75**	**0.998**

Note: [a]CT: Contourlet Transform, [b]AME: Approximate Maximum Eigenvalue, [c]CS: Compressive Sensing.

Table 9 shows the robustness performance values of the edge-based approaches against various attacks. Traditional edge-detection algorithms such as Canny, Sobel, and Zenzo were used in the studies in [21,23,24]. Methods using traditional edge detection algorithms have low payload capacity [19,20]. When the payload capacity is low, the model is assumed to have a high resistance to attacks. This study, which uses a novel edge detection algorithm that achieves high capacity, nevertheless achieved better results compared to the other studies, especially against noise, Gaussian LPF, cropping, and rotation attacks. Although the payload capacity is unclear in Kazemi et al. [24], it has less capacity than ours since they used a binary watermark. Although the method of Gong et al. [21] is slightly more resistant to JPEG, median filtering, and HE attacks than ours, their method has both low capacity and

low imperceptibility SSIM values. In addition, in comparison to our study, the noise resistance was lower, and no rotation attack was performed.

Table 9: Robustness comparison of edge-based methods

Attacks	Gong et al. [21]	Zhang et al. [23]	Kazemi et al. [24]	VSIME	HSIME
Gaussian Noise (0.005)	-	-	0.9531	0.9942	0.9944
Gaussian Noise (0.001)	0.9769	-	-	0.9995	0.9994
Gaussian Noise (0.01)	-	0.8876	-	0.9828	0.9776
Salt and Pepper Noise (0.005)	0.9502	-	-	1.00	1.00
Salt and Pepper Noise (0.01)	-	-	0.9785	1.00	1.00
Gaussian LPF	0.9754	-	-	1.00	1.00
Median Filtering (3×3)	0.9928	0.9747	0.9609	0.9891	0.9868
JPEG (QF = 50)	0.9964	0.9663	0.9900	0.9486	0.9353
Cropping 1/16	0.9495	0.9345	0.9141	1.00	1.00
HE	0.9964	-	0.5039	0.9931	0.9919
Rotation (2°)	-	0.7781	-	0.9923	0.9908

Table 10 shows the comparison with studies using high payload capacity [35–37] or scaling factors [16,26,11,38]. The methods in [35–37] used significantly high payload capacity. It is important that our proposed method achieves higher values against all attacks compared to these similar studies. Wang et al. [16] used both a high capacity and an adaptive scaling factor, as in our study. They embedded 16384 bits and the PSNR and SSIM values were calculated to be 40.74 dB and 1.00, respectively. Although the JPEG compression and median filtering robustness of the method are higher, we obtained higher resistance against cropping, HE, gamma correction, and rotation attacks in comparison to this study. In addition, the capacity values of our VSIME and HSIME techniques were 141483 bytes (1131864 bits) and 86697 bytes (693576 bits). In this context, while the PSNR values of the high-capacity VSIME and HSIME methods for the Lena image were 41.39 and 41.75 dB, the SSIM values were found to be 0.998 and 0.998. In the block-based color watermarking technique by Luo et al. [37], a 32×32 color logo (24576 bits) was embedded into a 512×512 color image. Although the developed method has high imperceptibility values, its capacity and resistance to noise, filtering, rescaling, and rotation attacks are lower than our proposed methods. According to the gray image watermarking technique based on LWT + DCT + SVD proposed by Takore et al. [26], a 32×32 (1024 bits) logo was hidden. Although the method has a high resistance to JPEG attacks in general, the NC value for particular JPEG attack with QF = 10 is 0.8983, which is lower than our obtained results of 0.9393 and 0.9230. In addition, the robustness of the 5-degree rotation attack is only 0.753. As a result, although our JPEG and gamma correction performances are slightly lower, our other robustness results are higher, especially rotation. They also used the PSO method, which has a high computational cost for determining the scaling factor. In the DCT + DWT + SVD-based gray image watermarking technique proposed by Begum et al. [11], high resistance to median filtering and salt and pepper noise and rotation attacks has been achieved. On the other hand, the method has low capacity and is vulnerable to JPEG compression attacks.

Table 10: Robustness comparison with respect to high payload capacity or scaling factor

Attacks	Prabha et al. [35]	Kathpal et al. [36]	Luo et al. [37]	Wang et al. [16]	Takore et al. [26]	Begum et al. [11]	Gao et al. [38]	Proposed	
								VSIME	HSIME
GN[a] (0.001)	-	-	-	0.9983	0.9985	-	-	0.9995	0.9994
GN (0.005)	-	0.5756	-	-	-	-	-	0.9934	0.9921
GN (0.01)	-	-	0.9428	-	-	-	0.8700	0.9828	0.9776
SPN[b] (0.005)	-	0.9857	0.9700	-	-	0.9956	-	1.0000	1.0000
SPN (0.01)	0.9971	-	0.9575	0.9868	0.9658	0.9912	-	1.0000	1.0000
Gaussian-LPF	0.9075	-	0.9578	0.9959	0.9977	-	-	1.0000	1.0000
MF[c] (3 × 3)	0.8370	0.7729	0.9402	0.9971	0.9782	1.0000	0.9200	0.9891	0.9868
JPEG (QF = 30)	-	-	-	0.9982	0.9864	-	-	0.9446	0.9303
JPEG (QF = 50)	-	-	0.8900	-	0.9900	-	0.9600	0.9486	0.9353
JPEG (QF = 70)	-	-	0.9330	-	1.0000	-	-	0.9493	0.9359
JPEG (QF = 90)	-	0.7877	0.94	-	1.0000	-	-	0.9491	0.9356
JPEG-2000 (2)	0.8577	-	-	-	-	-	-	0.9410	0.9260
Cropping 1/4	0.8661	-	0.9376	-	0.9746	-	-	0.9998	0.9997
RS[d] (512-256-512)	0.9014	-	0.9413	0.9977	0.9987	-	0.9800	0.9882	0.9990
HE	-	0.8650	0.9774	0.8176	0.9859	-	0.8900	0.9931	0.9919
Rotation (10°)	0.9378	-	0.94	-	-	0.9993	-	0.9928	0.9915
Rotation (45°)	-	0.3186	-	-	-	-	0.9400	0.9944	0.9933
GC[e]	0.9471	-	-	0.7637	0.9975	-	0.8500	0.9293	0.9118
Sharpening	0.9123	-	0.9689	0.9891	0.9964	-	0.8600	0.9992	0.9990

Note: [a]GN: Gaussian Noise, [b]SPN: Salt &Pepper Noise, [c]MF: Median Filtering, [d]RS: Rescaling, [e]GC: Gamma Correction.

In Table 11, the computational time values of edge detection, watermark embedding, and extracting of VSIME and HSIME methods are given. The time measures were obtained as a mean value of a 5-fold run for each image using MATLAB 2021b on a personal computer with Intel XEON ES-2680 V4 @ 2.40 Hz CPU and 128 GB RAM. Total time was calculated as (2 × edge-detection + embedding + extracting) since edge detection was performed for both embedding and extraction processes.

In this study, a high-capacity image watermarking technique is proposed using a novel similarity-based edge detection algorithm. Our hybrid method, including DWT and SVD, increased resistance to attacks. Unlike other optimization techniques, it calculated the adaptive scaling factor without computational cost. For this value to be image-based adaptive, the standard deviation value, the contrast measure of the related similarity image, was used. Calculation of the image-based scaling factor ensured that imperceptibility and robustness were high for each image. In addition, when the results obtained are compared with the literature, it has been concluded that noise, filtering, sharpening, cropping, and rotation results were higher, and the resistance to other attacks was also significantly increased. According to the literature, a slight performance vulnerability has been observed for JPEG compression attacks. When vertical and horizontal edge-based watermarking techniques are compared, it has been concluded that VSIME is partially better in imperceptibility, capacity, and robustness.

Table 11: Mean computational times with a 5-fold run of the proposed VSIME and HSIME methods

Image	VSIME times (s)				HSIME times (s)			
	Edge-Det.	Embedding	Extracting	Total	Edge-Det.	Embedding	Extracting	Total
Airplane	6.1207	0.6091	0.4735	13.3241	6.1938	0.5921	0.5004	13.4800
Mandrill	6.0320	1.5966	1.4756	15.1363	6.1369	1.5515	1.3820	15.2074
Peppers	6.1200	0.6444	0.5435	13.4279	6.0834	0.5845	0.4997	13.2509
Lena	6.1109	0.6213	0.5482	13.3913	6.1479	0.4989	0.4141	13.2087
Mean	**6.0959**	**0.8679**	**0.7602**	**13.8199**	**6.1405**	**0.8068**	**0.6990**	**13.7868**

6 Conclusion

In this study, our motivation is to overcome the main challenges, such as capacity, robustness, and imperceptibility in image watermarking. There are many different approaches to overcoming these challenges in the literature. This study proposed a novel similarity-based edge detection algorithm to increase the payload capacity, which can produce more edge points than conventional edge detection algorithms. The colored watermark image was created by inserting a randomly generated message on the edge points detected by this algorithm. To overcome the robustness problem, we used a DWT, and SVD-based hybrid method since the watermarking techniques in the transform domain have recently gained prominence due to their resistance to attacks. In the watermarking process, high-frequency bands in the wavelet domain were first obtained for cover and watermark images. Then, the singular values of these bands were combined using the scaling factor. In general, choosing a high scaling factor value increases resistance to attacks at the expense of low imperceptibility. Considering robustness and imperceptibility performances, it is impossible to determine a common scaling factor for all images. Therefore, the standard deviation, the contrast measure of the similarity image, was used to optimize the scaling factor adaptively for each image without computational cost. The obtained results can be summarized in two aspects. First, using all edge points detected by the similarity-based edge detection algorithm provided high payload capacity. Second, optimizing the scaling factor in color image watermarking achieved imperceptibility with increased robustness to all attacks, especially rotation.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, 2010.

[2] B. Li, J. He, J. Huang and Y. Q. Shi, "A survey on image steganography and steganalysis," *J. Inf. Hiding Multim. Signal Process.*, vol. 2, no. 2, pp. 142–172, 2011.

[3] M. Begum and M. S. Uddin, "Digital image watermarking techniques: A review," *Information*, vol. 11, no. 2, pp. 110, 2020.

[4] R. Karakış, İ Güler, I. Capraz and E. Bilir, "A novel fuzzy logic-based image steganography method to ensure medical data security," *Computers in Biology and Medicine*, vol. 67, pp. 172–183, 2015.

[5] A. Ray and S. Roy, "Recent trends in image watermarking techniques for copyright protection: A survey," *International Journal of Multimedia Information Retrieval*, vol. 9, no. 4, pp. 249–270, 2020.

[6] D. K. Mahto and A. K. Singh, "A survey of color image watermarking: State-of-the-art and research directions," *Computers & Electrical Engineering*, vol. 93, no. 107255, pp. 1–16, 2021.

[7] W. Wan, J. Wang, Y. Zhang, J. Li, H. Yu *et al.,* "A comprehensive survey on robust image watermarking," *Neurocomputing*, vol. 488, no. 2022, pp. 226–247, 2022.

[8] Z. Zainol, J. S. Teh, M. Alawida and A. Alabdulatif, "Hybrid SVD-based image watermarking schemes: A review," *IEEE Access*, vol. 9, pp. 32931–32968, 2021.

[9] O. Jane and E. Elbaşi, "Hybrid non-blind watermarking based on DWT and SVD," *Journal of Applied Research and Technology*, vol. 12, no. 4, pp. 750–761, 2014.

[10] F. Yasmeen and M. S. Uddin, "An efficient watermarking approach based on LL and HH edges of DWT–SVD," *SN Computer Science*, vol. 2, no. 2, pp. 1–16, 2021.

[11] M. Begum, J. Ferdush and M. S. Uddin, "A hybrid robust watermarking system based on discrete cosine transform, discrete wavelet transform, and singular value decomposition," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 8, pp. 5856–5867, 2022.

[12] K. A. Al-Afandy, O. S. Faragallah, E. S. M. EL-Rabaie, F. E. Abd El-Samie and A. ELmhalawy, "Efficient color image watermarking using homomorphic based SVD in DWT domain," in *Proc. JEC-ECC*, Cairo, Egypt, pp. 43–47, 2016.

[13] K. A. Al-Afandy, W. El-Shafai, E. S. M. El-Rabaie, F. E. Abd El-Samie, O. S. Faragallah *et al.,* "Robust hybrid watermarking techniques for different color imaging systems," *Multimedia Tools and Applications*, vol. 77, no. 19, pp. 25709–25759, 2018.

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

[15] A. M. Abdelhakim, H. I. Saleh and A. M. Nassar, "A quality guaranteed robust image watermarking optimization with artificial bee colony," *Expert Systems with Applications*, vol. 72, pp. 317–326, 2017.

[16] B. Wang and P. Zhao, "An adaptive image watermarking method combining SVD and wang-landau sampling in DWT domain," *Mathematics*, vol. 8, no. 5, 691, pp. 1–20, 2020.

[17] 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 and Applications*, vol. 77, no. 11, pp. 13197–13224, 2018.

[18] S. K. Ghosal, A. Chatterjee and R. Sarkar, "Image steganography based on kirsch edge detection," *Multimedia Systems*, vol. 27, no. 1, pp. 73–87, 2021.

[19] A. Ioannidou, S. T. Halkidis and G. Stephanides, "A novel technique for image steganography based on a high payload method and edge detection," *Expert Systems with Applications*, vol. 39, no. 14, pp. 11517–11524, 2012.

[20] S. A. Parah, J. A. Sheikh, J. A. Akhoon, N. A. Loan and G. M. Bhat, "Information hiding in edges: A high capacity information hiding technique using hybrid edge detection," *Multimedia Tools and Applications*, vol. 77, no. 1, pp. 185–207, 2018.

[21] L. H. Gong, C. Tian, W. P. Zou and N. R. Zhou, "Robust and imperceptible watermarking scheme based on canny edge detection and SVD in the contourlet domain," *Multimedia Tools and Applications*, vol. 80, no. 1, pp. 439–461, 2021.

[22] H. B. Razafindradina and A. M. Karim, "Blind and robust images watermarking based on wavelet and edge insertion," *International Journal on Cryptography and Information Security (IJCIS)*, vol. 3, no. 3, pp. 23–30, 2013.

[23] L. Zhang, P. Cai, X. Tian and S. Xia, "A novel zero-watermarking algorithm based on DWT and edge detection," in *Proc. CISP*, Shanghai, China, vol. 2, pp. 1016–1020, 2011.

[24] M. F. Kazemi, M. A. Pourmina and A. H. Mazinan, "Analysis of watermarking framework for color image through a neural network-based approach," *Complex Intell. Syst.*, vol. 6, no. 1, pp. 213–220, 2020.

[25] M. Mittal, R. Kaushik, A. Verma, I. Kaur, L. M. Goyal *et al.,* "Image watermarking in curvelet domain using edge surface blocks," *Symmetry*, vol. 12, no. 822, pp. 1–15, 2020.

[26] T. T. Takore, P. R. Kumar and G. L. Devi, "A new robust and imperceptible image watermarking scheme based on hybrid transform and PSO," *International Journal of Intelligent Systems and Applications*, vol. 10, no. 11, pp. 50–63, 2018.

[27] T. T. Takore, P. R. Kumar and G. L. Devi, "A robust and oblivious grayscale image watermarking scheme based on edge detection, SVD, and GA," in *Proc. of ICMEET*, Visakhapatnam, India, pp. 51–61, 2018.

[28] J. P. Dhar, M. Islam and M. A. Ullah, "A fuzzy logic based contrast and edge sensitive digital image watermarking technique," *SN Applied Sciences*, vol. 1, no. 7, pp. 1–9, 2019.

[29] R. Demirci, "Similarity relation matrix-based color edge detection," *AEU-International Journal of Electronics and Communications*, vol. 61, no. 7, pp. 469–477, 2007.

[30] K. Fares, A. Khaldi, K. Redouane and E. Salah, "DCT & DWT based watermarking scheme for medical information security," *Biomedical Signal Processing and Control*, vol. 66, no. 102403, pp. 1–9, 2021.

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

[32] P. Arbelaez, M. Maire, C. Fowlkes and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, 2010.

[33] J. R. Parker, "Chapter 2: Edge-detection techniques," in *Algorithms for Image Processing and Computer Vision*, 2nd ed., New York, USA: John Wiley & Sons Inc., pp. 1–43, 1997.

[34] R. Gonzales and R. E. Woods, "Wavelet and other image transforms," in *Digital Image Processing*, 2nd ed., N. J., USA: Pearson Prentice Hall, pp. 463–538, 2008.

[35] K. Prabha and I. S. Sam, "An effective robust and imperceptible blind color image watermarking using WHT," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 6, pp. 2982–2992, 2020.

[36] A. Kathpal and S. Jindal, "Dual image watermarking algorithm with SVD-DWT and edge detection on different layers of colored image," *International Journal of Computer Applications*, vol. 126, no. 5, pp. 6–9, 2015.

[37] Y. Luo, F. Wang, S. Xu, S. Zhang, L. Li *et al.,* "CONCEAL: A robust dual-color image watermarking scheme," *Expert Systems with Applications*, vol. 208, no. 118133, pp. 1–17, 2022.

[38] H. Gao and Q. Chen, "A robust and secure image watermarking scheme using SURF and improved artificial bee colony algorithm in DWT domain," *Optik*, vol. 242, no. 166954, pp. 1–11, 2021.