Cluster Computing (2025) 28:736
https://doi.org/10.1007/s10586-025-05582-9

Image watermarking based on spiking neural networks

Mürsel Ozan İncetaş[1] · **Mahmut Kılıçaslan**[2]

Received: 9 December 2024 / Revised: 18 May 2025 / Accepted: 5 June 2025 / Published online: 11 September 2025

Abstract
Image watermarking serves as a crucial technique for protecting copyrights and verifying the ownership of digital images. The watermarking process involves embedding data into images to prevent unauthorized usage of digital content. In this study, an innovative image watermarking method, using Spiking Neural Networks to embed robust and imperceptible watermarks, is proposed. The proposed method benefits from the edge detection capabilities of spiking neural networks to identify optimal regions for watermark locations. By targeting edge detection, this watermarking approach ensures significant resistance to common image processing attacks such as compression, noise addition, and cropping, while maintaining minimal perceptual distortion. The edge image obtained with the spiking neural network approach is divided into 16×16 non-overlapping blocks, and edge pixel definitions are made. Moreover, to increase the security level, the watermark image is scrambled with the help of a chaotic substitution box. The scrambled image is placed on the pixels marked as edges in the edge image. Afterward, it is divided into sub-bands by applying a discrete wavelet transform. The watermark image is inserted into the HH (High-High) band with the help of the discrete wavelet transform and the singular value decomposition approach. In the extraction stage, the HH band of the original image is used together with the watermarked image. Comprehensive experiments are conducted to evaluate the proposed technique, revealing its superiority in preserving both image quality and watermark integrity compared to conventional approaches.

Keywords Digital image watermarking · Spiking neural network · Discrete wavelet transform · Edge detection

1 Introduction

Rapid technological progress has greatly enhanced the use of network technologies. As a result, people are currently conducting transactions involving large volumes of data over the internet. The transmission and transport of digital content is used in many fields such as the military, medical, and education industries. For this reason, the protection of the privacy and security of such data has become a crucial issue. Recently, various frameworks for digital image watermarking have emerged as effective mechanisms for protecting copyrights and ensuring the integrity of digital image content against malicious attacks and spoofing. Digital image watermarking, which is characterized by hiding information within the source image, requires the integration of the watermark into the cover data with a degree of robustness that exceeds common signal distortions, including those caused by unintentional attacks [1, 2]. Many industries use image watermarking as an effective way to secure digital documents. An efficient watermarking system must guarantee that the embedded data remains unnoticeable. Maintaining the quality of the original image is essential when applying watermarking. Furthermore, the effectiveness of the system is also assessed based on its robustness, which measures the resilience of the watermark against alterations caused by attacks. Another crucial criterion for evaluating a watermarking system is its capacity, which denotes the volume of information that can be embedded in the image. However, it is anticipated that as the quantity of hidden data increases, the extent of degradation also escalates.

Image watermarking is generally analyzed in two categories: spatial domain (SD) and transform domain (TD) [3]. Techniques developed in the SD have attracted the attention of researchers due to their advantages especially

✉ Mahmut Kılıçaslan
m.kilicaslan@ankara.edu.tr

[1] Department of Computer Technologies, Alanya Alaaddin Keykubat University, Antalya, Turkey

[2] Department of Computer Technologies, Ankara University, Ankara, Turkey

in real-time situations, and many studies have been published [4–7]. Su and Chan proposed a new approach based on Schur decomposition in color images. In the SD method, an improved color image blind watermarking scheme based on Schur decomposition, which differed from using a binary or grayscale image as a watermark, was introduced [4]. In 2016, a technique was introduced for watermarking color images by employing the principal component analysis (PCA) transform and histogram matching techniques. The method adopted the YUV color space, where the luminance component was partitioned into non-overlapping blocks. Subsequently, the watermark was incorporated into the initial component of PCA following histogram matching. A robust and effective result was obtained in the aforementioned study [6]. In an alternative strategy, a watermarking technique grounded in the Schur decomposition principle of the SD was introduced. The maximum value of the SD in the Schur decomposition was derived through mathematical operations, and the watermark was subsequently inserted into the maximum eigenvalue of the SD [7]. In architectures using a TD, data is hidden in the cover image in the TD. Some transforms that convert the original image into the transform domain include the Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), and Discrete Wavelet Transform (DWT) [8–13]. SD watermarking directly modifies pixel values, while TD watermarking involves applying mathematical transforms to the image and embedding the watermark in the transformed space. Utilizing watermarking in the SD offers the benefit of simplicity and lower computational complexity. TD approaches can involve more complex mathematical operations and can be more computationally costly [14]. On the other hand, TD-based watermarking approaches offer a more robust architecture against attacks using different methods and components such as noise, rotation, and scaling [15].

It is seen that many studies have embedded the data to be hidden by changing the least significant bit (LSB) of the cover image. Additionally, the use of edge information in the aforementioned approaches has provided an efficient watermarking structure in recent years. Fahem et al. established a digital watermarking method utilizing the LSB, including image gradient and chaotic map elements. They divided the cover image into non-overlapping blocks of size 16×16, subsequently conducting gradient computations within these blocks. The scrambling of the watermark was achieved through a chaotic substitution box (S-Box) employing a piecewise linear chaotic map. They achieved high robustness and imperceptibility in their study [16]. In another study carried out in 2023, a digital image watermarking technique using the LSB and Canny edge detection was proposed. The proposed approach ensured satisfactory security while incurring minimal computational cost. The primary innovation involved the identification of optimal positions for watermark insertion and the enhancement of watermark security through the amalgamation of the watermark image. Suitable locations were determined using gradients, followed by the application of convolution masks to ascertain gradient direction and magnitude. Non-maximum suppression was subsequently employed, and the final step involved the use of the LSB for embedding the watermark [17]. Similarly, a chaos-based digital image watermarking method was proposed in another paper. Using a chaos-based approach, the data was encrypted. The watermark was then integrated into the original image, resulting in an enhanced level of security for the embedded watermark. In the study, watermark embedding employed the LSB in conjunction with a compass edge detector filter. The primary aim of the compass edge detector was to identify edges in various directions [18].

For frequency domain frameworks, DWT and DCT techniques have attracted attention, and there are many relevant approaches [2, 19]. In the proposed scheme to check the performance-capacity-robustness trade-off, we used DCT due to its strong compression properties, which produced successful results [20]. Likewise, DCT was employed in the study performed by Gül and Toprak. They introduced a resilient image watermarking technique that integrated the Contourlet transform, DCT, and singular value decomposition (SVD), leveraging the Artificial Bee Colony (ABC) algorithm to enhance robustness. In their proposed approach, the power factor and embedding locations were optimally selected for each image through the ABC algorithm [21]. In another the DCT-based watermarking approach, the most suitable region for watermark embedding was identified, ensuring a high level of imperceptibility. Demonstrating strong resilience against various attacks, the researchers emphasized a reduced computational cost [22]. Employing the frequency domain is preferred to improve the robustness of data-hiding results against attacks in image watermarking. In 2023, researchers introduced a novel blind watermarking technique that used a combination of DCT and DWT for color images. Initially, a self-adjusting color selection approach was used to take the blue or green channel of the cover image for embedding purposes. Following this, the selected color was separated into non-overlapping sub-blocks, and DCT was performed for each sub-block [23]. Naima et al. introduced a novel frequency-domain watermarking technique tailored for medical images. Their recent and effective approach employed the Fractional Discrete Cosine Transform (FDCT) in conjunction with Schur decomposition to achieve robust and secure watermark embedding. The watermark bits were incorporated by modulating the derived Schur coefficients, ensuring the preservation of visual quality while enhancing the security of the

watermarking process [24]. In another study, a watermarking methodology was recommended to increase the robustness and security of digital data [25]. The method integrated DWT and SVD. The embedding and extraction operations were carried out through multi-layered procedures utilizing both DWT and SVD. A novel reversible fragile digital watermarking method for medical image authentication and integrity was introduced. The proposed algorithm processed 8×8 blocks, transforming each into the frequency domain using DWT. The watermark information was embedded within the DWT coefficients, and second-level high- and mid-frequency DWT sub-bands were selected for watermark embedding [26]. One of the latest studies in this field examined a combined DWT and SVD technique for image watermarking, exploring the effect of a deeper SVD level on resilience and invisibility against typical signal processing and geometric distortions. In this method, two integrated watermarking models were developed: the first incorporated DWT and a primary-level SVD, while the second employed the same structure with a secondary-level SVD [27]. It is well-known that the use of image edges as watermarking regions boosts the imperceptibility and robustness of the watermarking architecture. Gong et al. proposed a watermarking approach combining Canny edge detection, contourlet transform, and singular value decomposition [28]. Similarly, watermarking algorithms based on edge detection have been introduced [16–18, 29]. Furthermore, an outstanding watermarking structure was proposed by Gürkahraman et al. in 2023. In the related approach, the pixel similarity-based edge detection technique was used. The pixel similarity-based edge detection approach is an easy-to-compute and effective solution. However, it proposes a linear solution like conventional edge detection techniques [30].

The commonality among the mentioned edge-based watermarking methods is that the calculation of gray-level differences between neighboring pixels is based on gradient-based techniques. Gradient methodologies based on linear differences between gray-level values have found widespread application in various fields thanks due to their computational simplicity. The gradient is typically computed using small blocks, such as those of size 3×3. The primary disadvantages of the gradient are its high sensitivity to noise and its tendency to detect false edges. Recently, edge detection methods that mimic the human visual system (HVS) have gained popularity over gradient-based techniques. In 1993, a dual-layer network project for edge detection was proposed an HVS image processing study [31]. This double-layer network structure has inspired subsequent research by providing a successful and efficient edge detection approach. In 2005, a method based on the leaky integrate and fire (LIF) neuron model, designed to emulate the HVS, was introduced for the segmentation of grayscale images [32]. Methods developed using spiking neural network (SNN) techniques, which process information through spikes generated by interconnected neurons, proved effective in HVS-based image processing [33]. An SNN-based edge detection method was proposed by Wu et al. [34]. The researchers introduced a network model where gray-level values from pixels were transmitted to intermediate-layer neurons via a mix of excitatory and inhibitory synaptic connections from the receptor layer. The spikes produced by neurons in the intermediate layer were then sent to corresponding neurons in the output layer through excitatory synaptic connections. Using Wu et al.'s approach, numerous studies have explored diverse configurations, experimenting with different receptors, intermediate layers, matrices, and window sizes [35–38]. SNN-based methodologies designed to emulate the HVS exhibit superior edge detection capabilities compared to gradient-based methods [34, 39, 40]. This sensitive effectiveness arises primarily from the fact that SNN-based approaches identify edges in the image by simulating neurons within the HVS, as opposed to relying on linear differences between pixel gray-level values. Moreover, the rapid execution of calculations with SNNs is facilitated by means of the development of models and analytical solutions [40, 41].

In recent years, it has been observed that edge detection-based digital watermarking studies have become popular. However, the edge detection approaches used in these studies are generally gradient-based, incompatible with the HVS, albeit computationally efficient. Gradient-based methods are usually noise-sensitive and have difficulty identifying salient features in images. Recent developments in biologically inspired models, especially SNNs, have shown similar results to HVS-like edge detection. Nevertheless, many existing SNN-based models are computationally intensive, hindering their integration into practical watermarking systems. This study aims to address these limitations by presenting a novel SNN-based edge detection approach that balances computational efficiency with improved watermarking performance.

In this paper, a novel image watermarking architecture is suggested using an SNN-based edge detection approach which effectively simulates the HVS. In the proposed approach, the cover image is first divided into 16×16 non-overlapping blocks. After this, a new SNN-based framework is employed for edge detection on each block. The edge pixels obtained using the SNN model are embedded with the watermark image. Later, the outcome image is integrated into the cover image using DWT-SVD techniques, completing the watermarking process.

The following are the major contributions of the proposed model:

- The proposed model uses a simplified 3× 3 receptive field, which reduces the computational complexity by approximately 2.5 times compared to many existing SNN-based approaches.
- The proposed biologically inspired approach increases edge detection accuracy by focusing on more prominent edge pixels while minimizing noise and false edges.
- The novel model is integrated with a DWT-SVD-based watermarking technique instead of conventional edge detection methods. Thus, SNN-based edge detection approaches are introduced to watermarking techniques for the first time.
- The experimental results show that the proposed approach is a significant improvement over existing edge-based techniques. Using the proposed SNN model greatly preserves computational efficiency while increasing watermarking performance.

Chapter 2 contains theoretical information about the techniques used in the proposed watermarking approach. The proposed edge detection method using an SNN and watermarking approach is explained in Sect. 3. Experimental results and comparisons to other related algorithms are given in Sect. 4. In the last section, conclusions and recommendations for future studies are presented.

2 Background

2.1 Conductance-Based integrate and fire neuron model

In the early 1950 s, Hodgkin and Huxley conducted a study to characterize human neurons. They employed a series of mathematical processes to describe neuronal functions. In other words, they used differential equations to simulate neuronal behavior [42]. The complex mathematical solutions in the efficient proposal of Hodgkin and Huxley increased its computational cost. Therefore, various neuron models, including the integrate-and-fire (IF), FitzHugh-Nagumo, and Izhikevich methods, have been introduced. In the IF neuron model, the variation of membrane potential v(t) is formulated as:

$$c_m = \frac{dv(t)}{dt} = g_l(E_l - v(t)) + \frac{w_{ex}g_{ex}(t)}{A_{ex}}(E_{ex} - v(t)) + \frac{w_{ih}g_{ih}(t)}{A_{ih}}(E_{ex} - v(t)) \quad (1)$$

Here, c_m, g_l, and E_l denote the membrane capacity, membrane conductance, and membrane reverse potential, respectively. E_{ex} represents the reversal potential of excitatory synapses, while E_{ih} represents the reversal potential of inhibitory synapses. The variable $g_{ex}(t)$ indicates the time-varying conductance of excitatory synapses, and similarly, $g_{ih}(t)$ indicates the time-varying conductance of inhibitory synapses. Excitatory connections in a neuron can transmit a stronger signal than inhibitory connections. When this occurs, the neuron's membrane potential surpasses the threshold voltage v_{th} at time t, triggering the neuron to generate a new action potential. Following this, starting at time $t + 1$, the neuron's membrane potential remains fixed at its initial value v_{reset} for a duration τ_{ref}, known as the refractory period. The value of τ_{ref} is crucial for simplifying model calculations. Thus, in this study, it is set to 0 as in similar studies [36, 38, 43]. Figure 1 shows a basic model of a neuron. In the corresponding figure, neurons that are synaptically connected to each other are shown to produce a new spike.

2.2 Chaotic map and substitution box

A Substitution Box (S-Box) is widely used in symmetric encryption with a single nonlinear component. A strong S-Box approach increases the security of a cryptographic algorithm. The strong nonlinearity of S-Box makes it effective against attacks [44]. S-Box architectures are categorized into algebraic and chaos-based groups. Algebraic S-Boxes are computationally complex, requiring a large number of mathematical procedures. On the other hand, chaos-based S-Boxes have the advantage of being less costly. Additionally, chaos-based approaches depend on the initial value. Therefore, various S-Boxes can be constructed depending on different initial values [45]. In this study, a chaos-based S-Box based on a piecewise linear chaotic map (PWLCM) and optimization technique is used, as in the methods found in the literature [16–18]. The PWLCM is calculated as

$$x_{n+1} = \begin{cases} x_n, 0 \le x_n < p \\ \frac{x_n - p}{0.5 - p}, p \le x_n < 0.5 \\ \frac{1 - p - x_n}{0.5 - p}, 0.5 \le x_n < 1 - p \\ \frac{1 - x_n}{p}, 1 - p \le x_n < 1 \end{cases} \quad (2)$$

where $x_0 \in [0, 1)$. The p-value is called the control factor and is defined as $p \in (0, 0.5)$. Chaotic maps are utilized to produce random numbers, which undergo additional processing to create the S-Box. The PWLCM demonstrates higher randomness and preserves a constant density function. The S-Box values used in our proposed approach are given in Table 1.

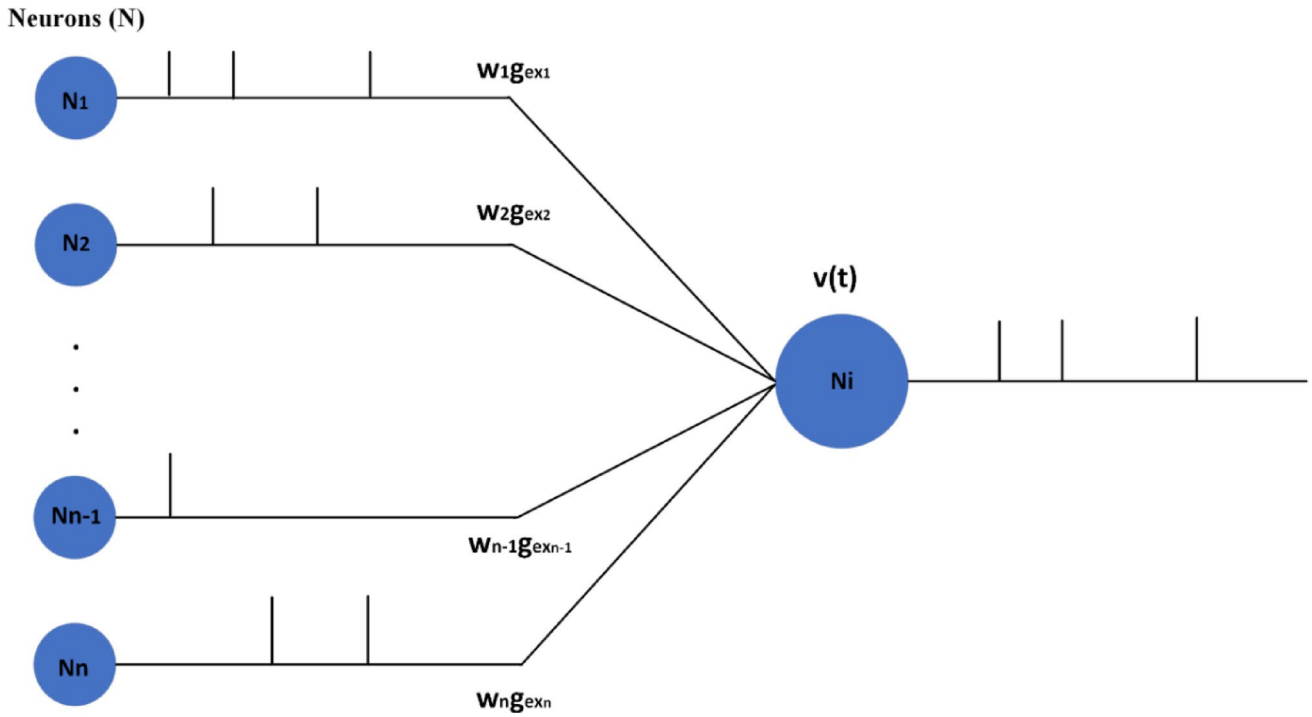

Fig. 1 Synaptic connections of conductance-based IF neuron model

2.3 Discrete wavelet transform and singular value decomposition

DWT is one of the most prevalently preferred mathematical transformations with applications in many fields [47]. DWT represents a given input signal using wavelets instead of decomposing it into frequencies [48]. Due to the significant robustness of DWT against attacks, it has been the center of attention of researchers in watermarking studies. 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 main image is transformed into 4 different sub-bands with DWT. These are the low-high (LH), high-low (HL), high-high (HH), and low-low (LL) sub-bands. The majority of the information in the primary image is concentrated in the LL sub-band after a single level of DWT. Typically, the majority of the energy of an image is concentrated in the low-frequency coefficient cluster (LL). While embedding a watermark in this region can substantially degrade image quality, it also enhances robustness. Conversely, the high-frequency coefficient set (HH) contains edge and texture information, to which the HVS is less sensitive. This makes the HH sub-band a suitable choice for imperceptible watermark embedding, preserving visual quality. HH has better performance against attacks such as compression or noise attacks. Therefore, the HH sub-band is used as an important feature in watermarking studies [30, 49, 50]. The results of several studies have indicated that DWT-based watermarking techniques offer advantages such as multiresolution representation, effective energy compaction, and high imperceptibility, making them suitable for image watermarking applications [51–53]. Nevertheless, a significant limitation of DWT-based watermarking is its vulnerability to geometric distortions [54]. This drawback can be mitigated by leveraging matrix decomposition to extract the geometric characteristics of an image. Consequently, methods integrating DWT and matrix decomposition have been extensively used in image processing and watermarking to enhance robustness against geometric attacks [54–56]. Among various matrix decomposition techniques, SVD is the most frequently utilized in watermarking.

SVD is a matrix transformation. It decomposes a symmetric matrix into three sub-matrices such that the singular values are separated in the form of a diagonal matrix [57]. Consider the image matrix $K \in R^{nxn}$, U and V are right and left orthogonal matrices $U = [u_1, u_2, \cdots, u_n] \in R^{nxn}, V = [v_1, v_2, \cdots, v_n] \in R^{nxn}$. S is a diagonal matrix and is denoted as $S = diag(\alpha_i)$. Where $\alpha_i, i = 1,2,3 \ldots, n$ are singular values of the diagonal matrix [58, 59]. The SVD of image *K* is computed as

Table 1 S-Box table values [46]

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	179	182	224	91	159	102	75	150	151	215	234	211	145	212	85	17
2	105	95	197	147	183	181	117	188	187	255	137	9	26	41	32	83
3	189	161	52	171	93	22	223	193	236	2	228	62	227	1	172	162
4	74	166	33	126	155	140	201	63	119	149	133	191	10	25	254	167
5	99	142	207	252	176	218	40	97	232	64	199	86	131	160	190	158
6	173	23	20	19	113	129	251	165	198	70	15	237	244	128	139	61
7	122	5	130	121	214	21	30	144	48	87	170	60	68	36	163	123
8	239	37	247	235	3	110	73	206	136	81	65	107	80	219	92	229
9	100	231	50	125	24	205	96	23	230	154	72	4	57	98	146	8
10	175	152	27	249	156	28	164	55	127	177	196	116	47	216	58	124
11	115	94	169	38	108	178	148	7	253	204	54	222	203	246	217	245
12	220	157	134	242	51	106	194	45	153	78	111	18	35	118	202	114
13	0	168	238	82	31	192	59	12	180	109	208	44	221	34	49	241
14	209	135	112	104	195	67	43	76	174	225	250	11	243	69	185	29
15	16	233	210	186	56	77	6	184	120	101	84	71	79	39	248	226
16	103	138	14	240	46	66	42	88	141	200	143	90	89	13	53	132

$$
\begin{aligned}
SVD(K) &= USV^T \\
&= [u_1, u_2, \cdots, u_n]\, x \begin{pmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & 0 & \cdots & \alpha_n \end{pmatrix} \\
&\quad x\, [\nu_1, \nu_2, \cdots, \nu_n] \\
&= \sum_{i=1}^{r} \alpha_i u_i \nu_i
\end{aligned} \tag{3}
$$

where r is the length of matrix K.

3 Proposed method

The structure of the SNN-based edge detection model utilized in the proposed watermarking method is initially clarified in this section. Subsequently, the relevant method is explained.

3.1 SNN model for edge detection

In this paper, a novel 3-layered SNN model is developed to detect edge pixels. The proposed SNN model is considered a 3-layered architecture, similar to existing models in the literature [34, 36, 38, 40, 41]. As illustrated in Figure 2, the first layer of the proposed model is the input layer. In the input layer, there are as many receptors as the number of pixels in the image. These receptors are synaptically connected to four neurons in the hidden layer through weight matrices. Weight matrices are used to determine whether the synaptic connections are excitatory or inhibitory. In the weight matrices, the symbol X represents excitatory connections, and the symbol Δ represents inhibitory connections. Each pair of neurons in the hidden layer has only excitatory synaptic connections with the neuron it is connected to in the output layer. The gray-level values of pixels in the output image are determined by the number of firings generated by the output neurons during duration T. Thus, edges in the images are detected through the proposed SNN model. The operation of the model is described in detail below.

In the input layer, there exists a receptor field (RF) of size 3×3 centered around each pixel in the image. The receptors in the RF are connected to neurons N_{x1}, N_{x2}, N_{y1}, and N_{y2} in the hidden layer via the weight matrices w_{x1}, w_{x2}, w_{y1}, and w_{y2}, respectively. These matrices are utilized to determine the coefficients and types (excitatory/inhibitory) of synaptic connections originating from the receptors in the RF. For the proposed SNN model in Figure 2, it is seen that the connections in the first row of the weight matrix of neuron N_{x1} are excitatory (X), while those in the last row are inhibitory (Δ). Receptors without any synaptic connections in the second row are represented by the symbol O. Through the w_{x1}

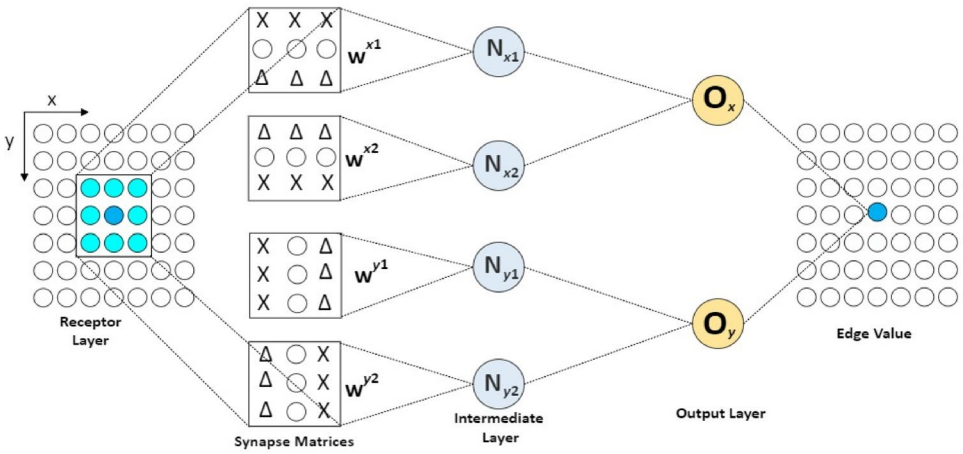

Fig. 2 Proposed SNN model for edge detection

matrix, neuron N_{x1} has excitatory synaptic connections with pixels in the first row of the *RF* and inhibitory synaptic connections with pixels in its third row. If the gray-level values of excitatory pixels are higher than those of inhibitory pixels, the membrane potential of neuron N_{x1} rises, leading to the generation of spike(s). On the contrary, if the gray-level values of excitatory pixels are lower than those of inhibitory pixels, the membrane potential of neuron N_{x1} remains unaffected, and it will not generate any spike. On the other hand, due to the structure of the w_{x2} weight matrix, neuron N_{x2} will generate spike(s) in such a scenario. Based on the structure of the IF neuron model, an increase in the difference between signals from excitatory and inhibitory synapses to the same neuron results in a higher spike rate per unit time. When the gray-level values of excitatory and inhibitory pixels in the *RF* are equal, the membrane potentials of both neurons remain unchanged, and no spikes are generated by either neuron. Gradient-based techniques utilize differences in gray-level values between pixels to detect edge pixels. However, the HVS detects edges based on the firing rates of synaptically connected neurons [60]. Through the w_{x1} and w_{x2} weight matrices, the relationship between pixel arrays is determined similarly to the HVS, enabling edge detection. If there is a difference in gray-level values between pixels in the first and third rows of the *RF*, one of the neurons N_{x1} or N_{x2} generates spike(s) intermittently, and these spike(s) are conveyed to the output neuron O_x. With the help of the spike count of neuron O_x per unit time, horizontal edges in the image are detected. The detection of vertical edges is similarly made with the aid of the excitatory signals conveyed by neurons N_{y1} and N_{y2} to neuron O_y.

For each *RF* in the input layer, there are two output neurons in the output layer. In the proposed network structure, the output neurons O_x and O_y in the output layer have excitatory synaptic connections with neurons in the intermediate layer. The output neuron O_x produces spike(s) at certain intervals according to the excitatory signals it receives from the neuron pair N_{x1} and N_{x2}. Similarly, the output neuron O_y produces spike(s) at certain intervals according to the excitatory signals it receives from the neuron pair N_{y1} and N_{y2}. By calculating the number of spikes generated by the neurons O_x and O_y corresponding to each pixel in the image, horizontal and vertical edges for the entire image are determined similarly to the HVS.

In the model in Fig. 2, the coordinates of the receptors of the pixels in the *RF* are denoted by (*x*,*y*). q_{ex} and q_{ih} are the peak conductance values of the receptors, and they are formulated as follows:

$$q_{ex} = \alpha G_{x,y}; \quad q_{ih} = \beta G_{x,y} \tag{4}$$

.

where x and y are the coordinates of the pixel, and $G_{x,y}$ is the gray-level value at that coordinate. α and β are the normalization coefficients and are taken as 1/255 as in many studies. Equations 5 and 6 are used when obtaining the conductance values for the neuron N_{x1}.

$$\frac{g_{ex}(t)}{dt} = \sum\nolimits_{(x,y)\in RF} \frac{{w_{ex}}^{x1}(x,y) q_{ex}(x,y)}{A_{ex}} - \frac{1}{\tau_{ex}} g_{ex}(t) \tag{5}$$

$$\frac{g_{ih}(t)}{dt} = \sum\nolimits_{(x,y)\in RF} \frac{{w_{ih}}^{x1}(x,y) q_{ih}(x,y)}{A_{ih}} - \frac{1}{\tau_{ih}} g_{ih}(t) \tag{6}$$

τ_{ex} in Eq. 5 is the time constant for the excitatory synaptic connection. Similarly, τ_{ih} in Eq. 6 is the time constant for the inhibitory synaptic connection and denotes the weights of the synapses to which the neuron N_{x1} is connected. For N_{x1}, g_{ex} is the conductance of the receptors in the first row of the *RF*. g_{ih} is the conductance of the receptors in the third row of the *RF*. For the calculation of N_{x2}, g_{ex} is the conductance of the receptors in the third row of the *RF*, and g_{ih} is the conductance of the receptors in the first row of the *RF*. In the calculation of the neurons N_{y1} and N_{y2}, the receptors in the

first and third columns of the *RF* are connected in the form of excitatory and inhibitory connections. For each of the neurons N_{x1}, N_{x2}, N_{y1}, and N_{y2} in the intermediate layer, the synaptic current generated by the receptors in the *RF* is I_{syn}.

$$I_{syn} = -g_{ex}E_{ex} - g_{ih}E_{ih} \tag{7}$$

The analytical solution of Vemuru [40] used to calculate the membrane potential of a neuron can be seen in Eq. 8.

$$\begin{aligned}\nu_{Nx_1} = (\frac{1}{g_l})\{(-\exp(\frac{g_l(t)}{c_m}))(I_{syn} + 70g_l + g_lE_l) \\ + I_{syn} + g_lE_l\}\end{aligned} \tag{8}$$

.

For neurons in the intermediate layer to produce spikes, membrane potentials must reach a threshold voltage. The spikes produced by a neuron per unit time are stored in an array and are called Spike Trains. For example, the spike train S_{Nx1} for the neuron N_{x1} is generated using Eq. 9.

$$S_{Nx1}(t) = \begin{cases} 1 & \text{if neuron fires a spike at time t} \\ 0 & \text{if there is no spike at time t} \end{cases} \tag{9}$$

.

For each *RF* in the receptor layer, there are two separate output neurons, O_x and O_y, in the output layer. There is only the excitatory synaptic connection between the pair of neurons N_{x1} and N_{x2} in the intermediate layer and the neuron O_x in the output layer. The same is true for the neuron pair N_{y1} and N_{y2} and the neuron O_y. The calculations for the output neuron O_x are performed via Eqs. 10, 11 and 12. The same equations are used for O_y.

$$\frac{g_{out}(t)}{dt} = -\frac{1}{\tau_{out}}g_{out}(t) + \frac{S_{N_{x1}}(t) + S_{N_{x2}}(t)}{A_{ex}} \tag{10}$$

$$I_{out} = -g_{out}E_{out} \tag{11}$$

$$\begin{aligned}v_{Ox} = (\frac{1}{g_l})\{(-\exp(\frac{g_lt}{c_m}))(I_{out} + 70g_l + g_lE_l) \\ + I_{out} + g_lE_l\}\end{aligned} \tag{12}$$

In the calculations for the neuron O_x, the spike trains of N_{x1} and N_{x2} are used for the g_{out} conductance value. In Eq. 10, τ_{out} is the time constant. The synaptic current to the neuron O_x is expressed as I_{out}. The spike train S_{Ox} of the neuron O_x is obtained using Eq. 9. Similar operations are performed for the neuron O_y. With the help of the S_{Ox} and S_{Oy} spike trains obtained for O_x and O_y, the P_x and P_y spike counts are calculated. As seen in Eq. 13, P_x and P_y are the number of spikes generated by the neurons O_x and O_y during time T, respectively.

$$P_x = \sum_{t=0}^{T} S_{Ox}(t) \text{ and } P_y = \sum_{t=0}^{T} S_{Oy}(t) \tag{13}$$

With the help of the P_x and P_y values, horizontal and vertical edges are obtained. These edges are converted into a single edge value using Eq. 14. Thus, the edges in the image are detected similar to the HVS with the help of the proposed SNN model.

$$O_{edge} = \sqrt{\frac{{P_x}^2 + {P_y}^2}{2}} \tag{14}$$

The edges are detected by assigning the O_{edge} values obtained for all pixels in the image as gray-level values. The edge detection results of an example image are shown in Fig. 3. Figure 3(b) shows the result of the conventional gradient approach, while Fig. 3(c) shows the edge detection result of the proposed SNN model. There are two main differences between the edges detected with the proposed SNN model compared to the edges detected with the gradient method. The edges detected with the SNN are thinner and sharper. Additionally, with the gradient approach, every detail in the image appears as an edge, whereas with the SNN, the most prominent regions that the human eye perceives at first glance are detected as edges.

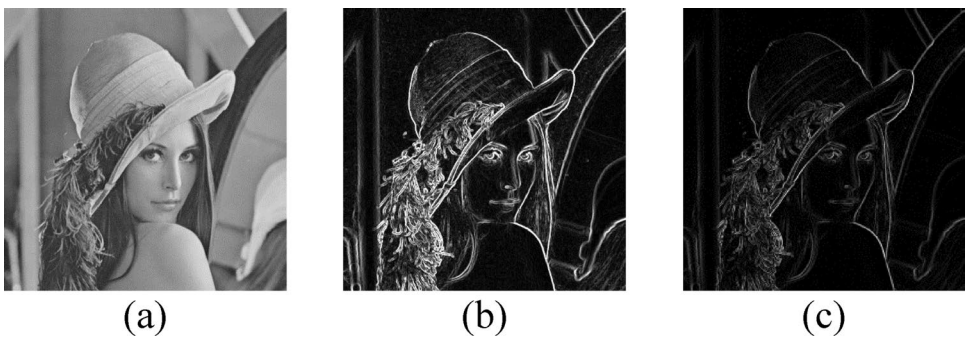

Fig. 3 Edge detection results of Lena image. (**a**) Original image, (**b**) Gradient, (**c**) Proposed SNN model

3.2 Image watermarking using SNN-based edge detection

An overview of the proposed method is shown in Fig. 4, while Fig. 5 describes the detailed steps. The proposed watermarking method consists of two stages: insertion and extraction. Initially, the watermark image undergoes scrambling using a chaotic map [16–18]. Then, the edge pixels of the cover image are labelled. The proposed SNN model is used to classify a pixel as an edge. The edge image (I_E) of the host image (I_H) is generated. The edge image I_E is partitioned into non-overlapping sub-regions of dimensions 16 ×16. If the gray-level value of a pixel in the I_E edge image is greater than or equal to the average gray-level of the 16 ×16 sub-region it belongs to, the pixel is designated as an edge. If fewer edge pixels are formed than the number of pixels in the watermark logo image, the pixels with the highest gray value in the edge image are marked as edges. The gray-level values of the scrambled watermark image (I_{WS}) pixels are subsequently assigned to the pixels marked as edges in the host image to produce the I_{WE}. Consequently, the concealed image is obtained with the same dimensions as the host image. Sub-bands for both the image to be concealed (I_{WE}) and the host image (I_H) are generated using DWT. Given that the HH sub-band is recognized for its success in imperceptibility among the four sub-bands produced by DWT [30, 50], this study utilizes the HH sub-band. SVD is performed on the HH bands derived from both the host image and the image to be hidden. By combining the obtained singular values with the scaling factor (α), the watermarked image (I_{WM}) is constructed through the application of inverse SVD and inverse DWT operations. Thus, the insertion phase is completed.

In the watermark extraction stage, sub-bands for the host image (I_H) and the watermarked image (I_{WM}) are derived using DWT. At this point, the singular value obtained from applying SVD to the HH bands of both images should be subtracted from the singular value of the hidden image. To enhance the success of the extraction process, the method proposed by Naffouti et al. [13] is employed. If the correlation between the original and watermarked images is equal to 1, the scaling factor (α) used during insertion is included in the calculation. Otherwise, the singular value of the watermarked image is computed without incorporating the scaling factor (α). The hidden image is then retrieved using inverse SVD and inverse DWT. Finally, the hidden image is substituted back using the S-Box to obtain the extracted watermark image (I_{EW}). The insertion and extraction stages are shown below step by step:

Insertion Stage:

1. Input Image Host (I_H).
2. Input Image Watermark (I_W).
3. Calculate S-Box substitution (I_{WS}) of I_W.
4. SNN-based Edge Detection (I_E).
5. Divide edge image I_E into non-overlapping blocks B_k (each block size is 16 × 16 k = 1,…, 1024 for image size 512 × 512).

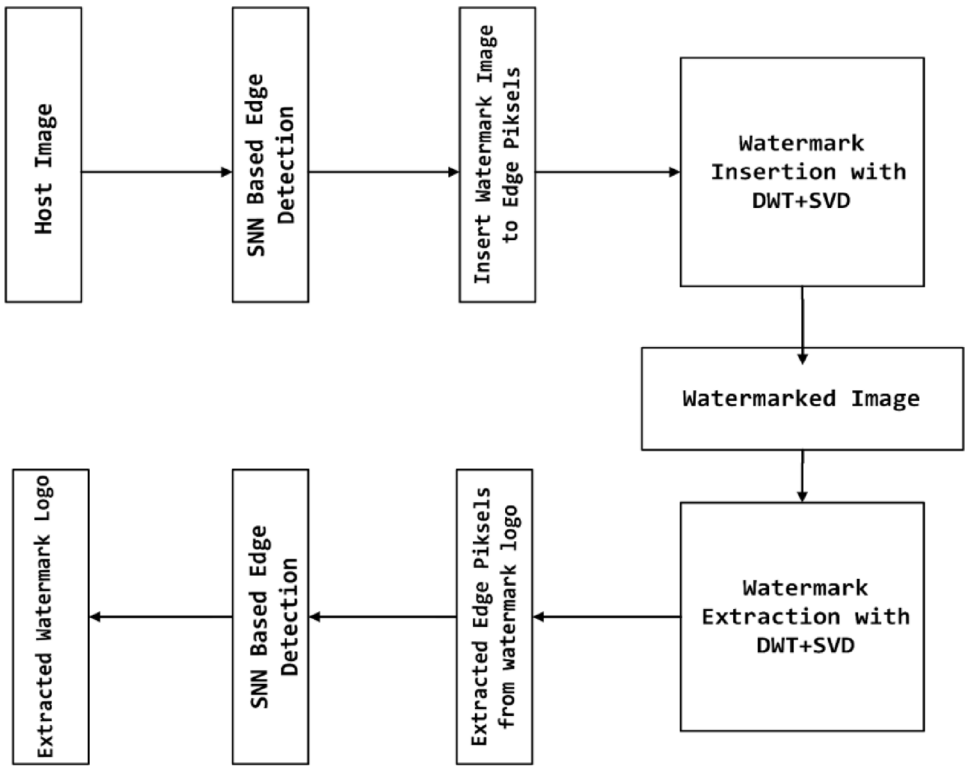

Fig. 4 General diagram of the proposed method

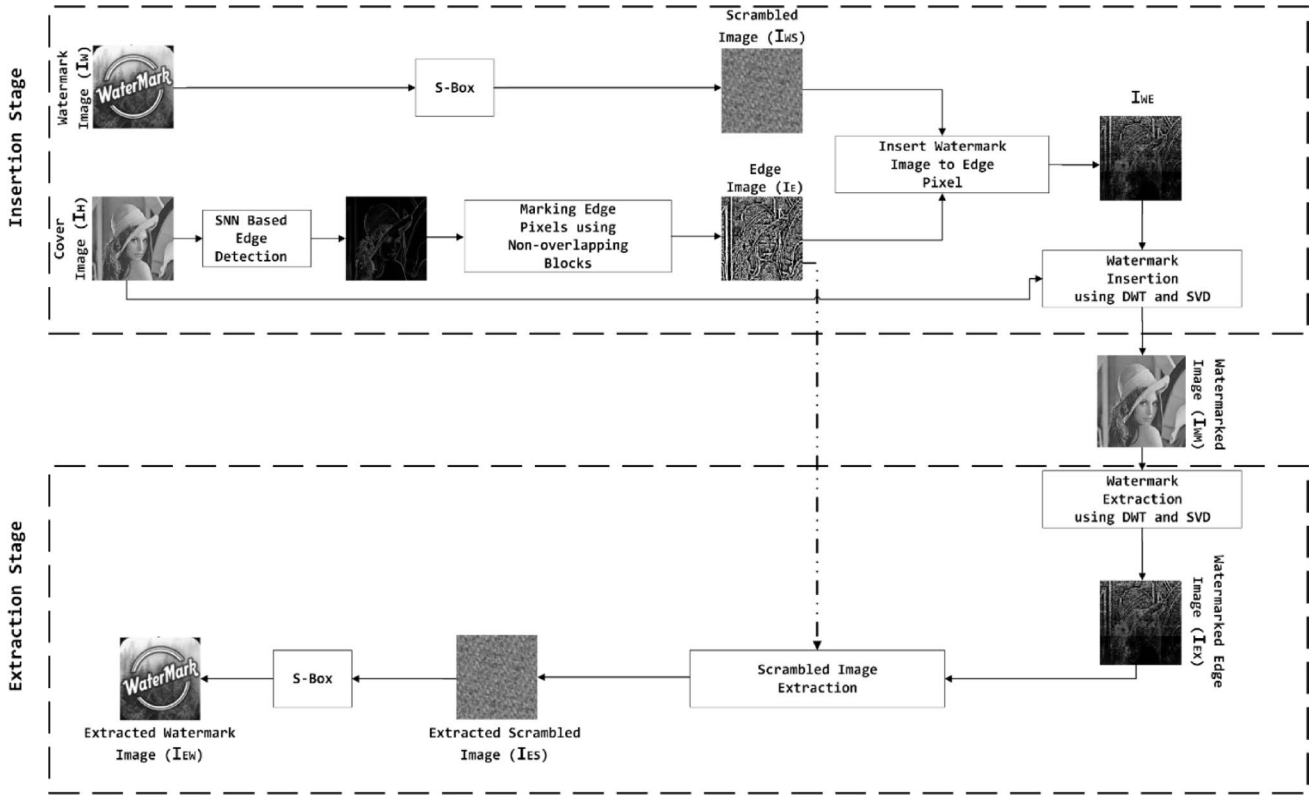

Fig. 5 Detailed flow chart of the proposed method

6. Find average edge value $E_{k,\,avg}$ of each edge block B_k.

$$E_{k,avg} = \text{average}\,(\mathbf{B_{k,\,i,\,j}})\,(i = 1 \ldots 16 \text{ and } j = 1 \ldots 16, \text{ for the image size } 512 \times 512)$$

7. Mark the edge pixel by assigning the value 1 ($E_{k,\,i,\,j}$ =1).

$$E_{k,i,j} = 1 \;\; (\text{edge pixel}),\; B_{k,i,j} >= E_{k,\,avg}$$

$$E_{k,i,j} = 0 \;\; (\text{non} - \text{edge pixel}),\; \text{otherwise}$$

8. Obtain I_{WE} by insert watermark image I_{WS} to edge pixels of I_E ($E_{k,\,i,j}$=1).
9. Apply DWT to I_H and obtain LL_H, LH_H, HL_H, HH_H sub-bands for I_H.
10. Apply DWT to I_{WE} and obtain LL_{WE}, LH_{WE}, HL_{WE}, HH_{WE} sub-bands for I_{WE}.
11. Apply SVD to HH_H sub-band of I_H.

$$[HU_H,\; HS_H,\; HV_H] \;=\; \text{SVD}\,(HH_H)$$

12. Apply SVD to HH_{WE} sub-band of I_{WE}.

$$[HU_{WE},\; HS_{WE},\; HV_{WE}] \;=\; \text{SVD}\,(HH_{WE})$$

13. Obtain modified singular value.

$$HS_M = \; HS_H \; + \alpha * HS_{WE}$$

14. Obtain modified HH band.

$$HH_M = \; HU_H * \; HS_M * \; HV_H$$

15. Obtain watermarked image (I_{WM}).

$$IW_M = \; \text{idwt}\,(LL_H,\; LH_H,\; HL_H,\; HH_M)$$

Extraction Stage:

1. Input Image Host (I_H).
2. Input watermarked image (I_{WM}).
3. Apply DWT to I_{WM} and obtain LL_{WM}, LH_{WM}, HL_{WM}, HH_{WM} sub-bands for I_{WM}.
4. Apply SVD to HH_{WM} sub-band of I_{WM}.

$$[HU_{WM},\; HS_{WM},\; HV_{WM}] \;=\; \text{SVD}\,(HH_{WM})$$

5. Obtain extracted singular value.

$$r \;=\; \text{correlation}\,(HS_{WM},\; HS_H)$$
$$\text{if } r < 1 \text{ then } HS_{EX} = \; HS_{WM} - \; HS_H$$
$$\text{else } HS_{EX} = \; (HS_{WM} - \; HS_H)\,/\alpha$$

6. Obtain extracted HH band.

$$HH_{EX} = HU_{WE} x HS_{EX} x HV_{WE}$$

7. Obtain extracted watermarked image (I_{EX}) using Inverse DWT (idwt).

$$I_{EX} = idwt\,(LL_{WE},\ LH_{WE},\ HL_{WE},\ HH_{EX})$$

8. Obtain I_{ES} from I_{EX} using edge pixels in I_E.
9. Obtain extracted watermark image I_{EW} with S-Box substitution of I_{ES}.

4 Experimental results and discussions

In this study, the University of Southern California SIPI (Signal and Image Processing Institute) image dataset is used for calculating the experimental results. The dataset is available at https://sipi.usc.edu/database/. The experiments are performed on 20 images of 512 ×512 size. Three color images, other than the 17 gray-level images, are converted to gray-level. A grayscale image (256 ×256) is used as a watermark. In this section, the imperceptibility and robustness of the proposed scheme are tested. Figure 6 shows some cover images in SIPI and the watermark used in this study. The performance of watermarking approaches is generally examined in terms of imperceptibility and robustness. PSNR, SSIM, UIQI, and NC are various objective performance measures used to evaluate the effectiveness of proposed approaches [13]. High PSNR values and SSIM values close to 1 indicate that the watermarked image is imperceptibly different from the cover image. The robustness of the proposed system is tested against various attack scenarios such as no attack, (a) Gaussian noise (0.01), (b) salt & pepper noise (0.05), (c) speckle noise (0.05), (d) Gaussian Filter ([3 ×3] and 0.5), (e) average filter ([3 ×3]), (f) median filter ([3 ×3]), (g) Wiener Filter ([3 ×3]), (h) histogram equalization (64 bins), (i) sharpening (amount: 1 and radius: 0.8), (j) gamma correction (Gamma value: 0.2), k) translation (80 ×80), l) rescaling (512 →256→512), m) rescaling (512→1024→512), n) cropping (1/4), o) rotation, p) blurring, q) white Gaussian noise, r) JPEG compression and s) Motion Blur.

The proposed method's performance is evaluated using the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Universal Image Quality Index (UIQI), and Normalized Correlation (NC) metrics, which are widely utilized in watermark studies. PSNR quantifies the quality of the watermarked image relative to the original, with higher PSNR values indicating greater similarity. This value is expressed in decibels (dB). UIQI assesses the impact of watermarking by evaluating correlation loss, brightness distortion, and contrast degradation. SSIM, based on human visual perception, measures the similarity between two images by analyzing brightness, contrast, and structural components. While SSIM operates similarly to UIQI, it performs a more refined local analysis. NC evaluates the direct correlation between the original and processed images, primarily focusing on watermark preservation. Unlike UIQI and SSIM, NC emphasizes pixel-level similarity and is less responsive to structural modifications [61–63].

The recommended SNN model is executed in MATLAB. The SNN model parameters are obtained from Wu et al. and İncetaş [34, 64]. The excitatory and inhibitory weight matrices of the neuron N_1 are as follows:

$$w_{ex}^{x1} = \begin{bmatrix} 0.21 & 0.36 & 0.21 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } w_{ih}^{x1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.21 & 0.36 & 0.21 \end{bmatrix}$$

4.1 Imperceptibility analysis

Imperceptibility is a significant performance criterion in watermarking approaches. The PSNR and SSIM metrics are widely used performance measures for imperceptibility analyses. A PSNR value greater than 48 indicates a very effective result. This means that even though information is hidden in the cover image, the distortion of the original image is minimal. A value between 35 and 48 is

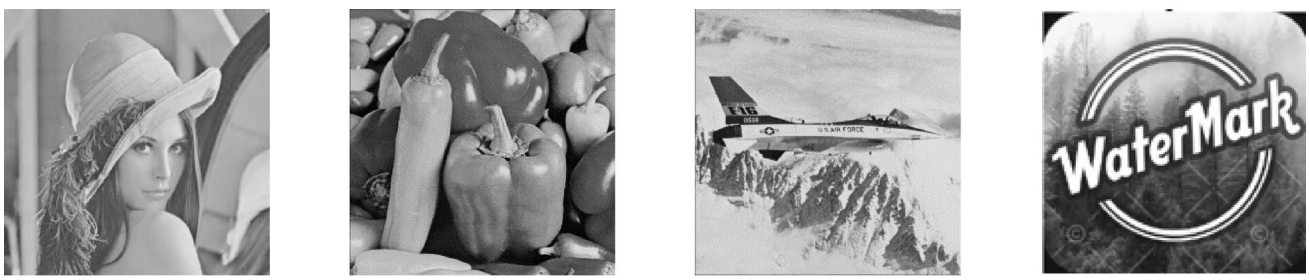

Fig. 6 (**a**)-(**c**) Cover images: Lena, Peppers, Plane, (**d**) Watermark image

Table 2 Results of watermarked images based on various metrics

Cover image	PSNR	SSIM	UIQI	NC
Lake	54.8730	0.9999	0.9995	1.0000
Lena	54.8730	1.0000	0.9998	1.0000
Boat	54.8730	1.0000	0.9999	1.0000
Peppers	54.8731	1.0000	1.0000	1.0000
Plane	54.8750	1.0000	1.0000	1.0000
Mandrill	54.8738	0.9998	0.9999	1.0000
Couple	54.8730	1.0000	1.0000	1.0000
4.02.01	52.5993	0.9971	0.9999	1.0000
5.02.09	54.6523	1.0000	0.9998	1.0000
5.02.10	53.8312	1.0000	0.9999	1.0000
7.01.01	54.0301	0.9993	0.9997	1.0000
7.01.02	56.6645	1.0000	0.9997	1.0000
7.01.03	52.9688	0.9986	0.9996	1.0000
7.01.04	53.5525	0.9987	0.9998	1.0000
7.01.05	53.6312	0.9998	0.9998	1.0000
7.01.06	53.4124	0.9997	0.9997	1.0000
7.01.07	53.1233	0.9994	0.9995	1.0000
7.01.08	53.4354	0.9982	0.9996	1.0000
7.01.09	53.5535	0.9995	0.9998	1.0000
7.01.10	53.3800	0.9987	0.9996	1.0000
Average	54.1474	0.99943	0.9997	1.0000

Table 3 Results of extracted watermark based on various metrics

Cover image	PSNR	SSIM	UIQI	NC
Lake	48.0341	1.0000	0.9867	0.9999
Lena	47.9854	1.0000	0.9901	0.9999
Boats	47.9499	1.0000	0.9768	0.9999
Peppers	48.1480	1.0000	0.9724	1.0000
Plane	48.1126	1.0000	0.9754	0.9996
Mandrill	47.9577	1.0000	0.9765	1.0000
Couple	48.0031	1.0000	0.9875	0.9998
4.02.01	48.3900	1.0000	0.9912	0.9986
5.02.09	47.4398	1.0000	0.9827	1.0000
5.02.10	48.1782	1.0000	0.9831	0.9998
7.01.01	49.4578	1.0000	0.9927	0.9999
7.01.02	45.5344	1.0000	0.9857	0.9997
7.01.03	47.2498	1.0000	0.9907	1.0000
7.01.04	48.0831	1.0000	0.9876	1.0000
7.01.05	47.9213	1.0000	0.9866	0.9999
7.01.06	48.2162	1.0000	0.9792	1.0000
7.01.07	47.8520	1.0000	0.9785	1.0000
7.01.08	47.4761	1.0000	0.9817	1.0000
7.01.09	47.7049	1.0000	0.9927	1.0000
7.01.10	48.4302	1.0000	0.9844	1.0000
Average	47.8807	1.0000	0.9837	0.9999

indicative of acceptable image quality [13]. The results for the watermarked and extracted watermark images are given in Tables 2 and 3, respectively.

Table 2 shows that the average PSNR and SSIM values for the watermarked image are 54.8734 and 0.9999, respectively. Moreover, the UIQI and NC values are 0.9999 and above. Clearly, the relevant experimental results show that there is very little distortion in the watermarked image. The metrics produce values close to each other in different images. However, the 'Plane' image is the most effective image for the proposed approach with a value of 54.8750 dB, which is better than the average. These results prove the effectiveness of the SNN technique adapted to the HVS. The non-linear nature of the SNN approach produces better results than gradient-based approaches. Measurements such as PSNR, MSE, SSIM, UIQI, and NC are once again employed to evaluate the proposed watermark extraction algorithm without exposure to attacks. Table 3 explains all criterion outcomes between the original and extracted watermarks. As indicated in Table 3, a maximum SSIM of 1.0000 is achieved for all extracted watermark images, and UIQI and NC values were very close to 1, which signifies that the extracted watermark images are obtained with high quality. Furthermore, the PSNR and SSIM values also confirm the output. In other words, the grayscale watermark image can be extracted very close to the original without attack exposure. Figure 7 shows the watermarked and extracted watermark images of some of the samples used in the experiments. The corresponding figure represents the results without any attacks.

Tables 4 and 5 illustrate the performance levels obtained in some previous fundamental watermarking studies [8, 13, 25, 59, 65–78] compared to the proposed method using the Lena and Peppers images, respectively. The tables present the evaluation outcomes of the watermarked image and the extracted image separately. The values in the tables are the results obtained when there is no attack on the watermarked images. The information in Table 4 reveals that the proposed method surpasses the others. It is clearly seen in Table 4 that the technique proposed in another study conducted in 2020 [71] offers an effective architecture. Notwithstanding, it can be seen in Tables 4 and 5 that the recommended method is the most successful method. The numerical PSNR values also prove the success of the method. For example, compared to the results of a recent study [13], the proposed approach shows an improvement of approximately 4% in terms of the PSNR value of the watermarked image. The PSNR value for the extracted watermark is approximately 48 dB with the proposed technique, compared to about 36 dB in the aforementioned study [13]. Additionally, our proposed technique exhibits better PSNR values than other techniques [25, 65, 68]. Table 5 presents similar results to those in Table 4. The data in both tables clearly indicate that the proposed method is effective regarding the imperceptibility parameter.

In Tables 4 and 5, the proposed SNN-based digital watermarking method shows remarkable results in the performance comparisons performed on Lena and Peppers test images. Especially, when the PSNR, SSIM and NC metrics measured regarding the quality of both water-marked

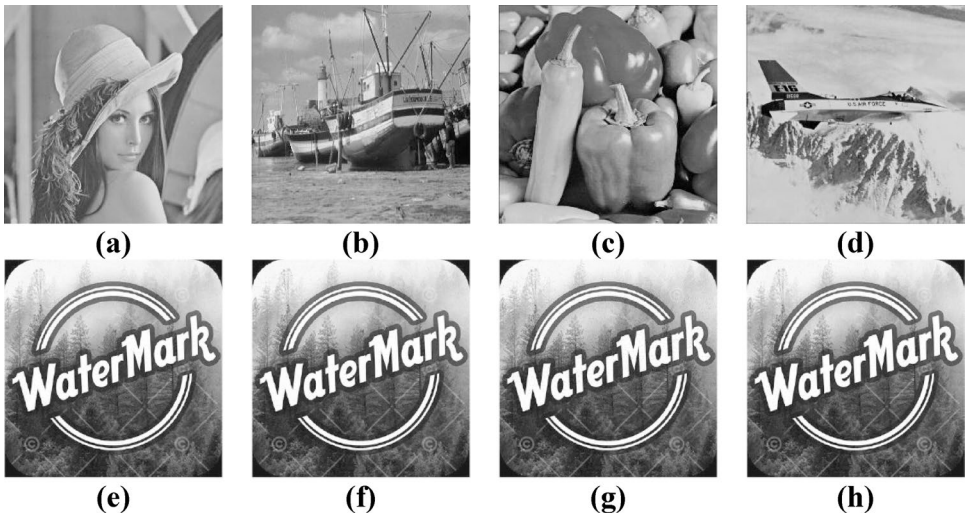

Fig. 7 Output of watermarked images and extracted watermark images without any attack: (**a**)-(**d**) watermarked images, (**d**)-(**e**) extracted watermark images

Table 4 Performance evaluation of proposed and other methods for Lena

	Watermarked Image		Extracted Image	
Method	PSNR	SSIM	PSNR	NC
[8]	43.6461	–	–	1.0000
[13]	50.2806	1.0000	36	0.9996
[59]	44.2729	–	–	1.0000
[25]	45.7942	0.9927	35	0.9934
[65]	45.2634	0.9821	–	1.0000
[66]	30.8072	0.8872	–	1.0000
[67]	-	0.9975	–	1.0000
[68]	53.1866	0.9795	–	1.0000
[69]	43.3648	–	–	1.0000
[70]	42.5597	0.9998	–	1.0000
[71]	62.6643	0.9998	–	1.0000
[72]	47.2422	–	–	–
[73]	49.203	–	–	1.0000
[74]	41.9860	0.9832	–	1.0000
[75]	39.626	–	–	1.0000
[76]	40.696	–	–	1.0000
[77]	53.187	0.9972	–	–
[78]	38.8895	–	–	0.9981
Proposed	54.8730	1.0000	48.0272	0.9999

Table 5 Performance evaluation of proposed and other methods for peppers

	Watermarked Image		Extracted Image	
Method	PSNR	SSIM	PSNR	NC
[8]	43.8612	–	–	1.0000
[13]	50.2807	1.0000	38	0.9996
[65]	42.0182	0.9604	–	0.9945
[66]	35.8320	0.9471	–	1.0000
[67]	-	0.9968	–	0.9872
[68]	51.0910	0.9847	–	1.0000
[68]	39.8958	–	–	1.0000
[70]	41.8911	0.9998	–	1.0000
[71]	62.9223	0.9998	–	1.0000
[73]	47.1437	–	–	1.0000
[74]	44.1371	0.9857	–	1.0000
[75]	39.4553	–	–	1.0000
[76]	40.7115	–	–	1.0000
[77]	54.3020	0.9901	–	–
[78]	38.9708	–	–	0.9985
Proposed	54.8731	1.0000	48.1480	1.0000

images and extracted images are evaluated, it is seen that the proposed method outperforms most of the existing methods. The PSNR value obtained by the proposed method on the Lena image with water-marked image is among the highest values with 54.8730, and it offers excellent visual quality with an SSIM value of 1.0000 in terms of structural similarity. Similarly, it is observed that the extracted watermarked image exhibits a competitive recovery success with a PSNR value of 48.03 and a NC value of 0.9999. The performance on the Peppers image is also at a similarly high level. The proposed method stands out as one of the best methods in preserving both visual quality and watermark integrity with values such as PSNR 54.87, SSIM 1.0000, extracted watermark PSNR 48.15 and NC 1.0000. These findings show that the proposed SNN-based method is successful in terms of both visibility and robustness and exhibits superior performance compared to many methods in the literature. Especially high SSIM and NC values show that the method produces high-quality results that cannot be perceived by the human eye and that the watermark can be recovered without degradation. In addition, obtaining consistent results on both images reveals the generalizability and reliability of the method.

4.2 Robustness analysis

Robustness evaluations are crucial in analyzing watermarking architectures. The robustness of a proposed method is assessed by subjecting the watermarked images to various

attacks. In other words, robustness is the efficiency of extracting the desired image after the attacks applied to the images. To assess the robustness and reliability of the proposed method, the watermarked images are subjected to 19 different attacks commonly used in watermarking studies. Figure 8 shows images where Gaussian, salt & pepper, and speckle attacks are applied to the Lena and Peppers watermarked images. Additionally, the watermark images extracted from these images are shown. Noise attacks are among the most challenging attacks for robustness analyses. The corresponding outputs in this study are visually satisfactory, and the numerical evidence is provided in Table 6. Table 6 demonstrates the performance values of the extracted watermarks. It is revealed that the highest UIQI and NC values are 0.999 and 1, respectively. Besides, the developed technique demonstrates strong robustness against types of noise such as Gaussian, salt & pepper, and speckle noise. It also proves to be robust against other types of attacks. Although cropping the watermarked images by ¼ introduces distortions, these losses are anticipated.

To validate the comparison data, the developed technique is compared to various algorithms, ensuring a reasonable evaluation in terms of robustness. Tables 7 and 8 detail the

Fig. 8 Sample attacked watermarked images and extracted watermark: (**a**)-(**c**) noisy watermarked Lena with Gauss (0.01), Salt & Pepper (0.05) and Speckle (0.05), respectively, (**d**)-(**f**) extracted watermark from noisy watermarked Lena, (**g**)-(**i**) noisy watermarked Peppers with Gauss (0.01), Salt & Pepper (0.05) and Speckle (0.05), respectively, (**j**)-(**l**) extracted watermark from noisy watermarked Peppers

NC and UIQI performance criteria for the Lena and Peppers images in the context of different attack scenarios. In comparison to other algorithms, the proposed algorithm exhibits competitive performance, particularly in the noise attacks. It demonstrates superior performance, especially under noisy conditions. The effectiveness of the algorithm across three different types of noise for both the Lena and Peppers images clearly illustrates its robustness. One of the most recent approaches in the literature [13] achieved NC values of 0.9995, 0.9989, and 0.9993 for Gaussian, salt & pepper, and speckle noises, respectively, for the Lena image, whereas these values are 0.9996, 0.9993, and 0.9996 in the proposed technique. These results clearly highlight the achievement of the SNN-based watermarking method. The primary reason for this success is the ability of the SNN approach to excellently simulate the HVS and accurately detect edge pixels. Similar results are observed in Table 8. All techniques used for comparison yield successful results in filter attacks (d-g). Moreover, the proposed method remains competitive in these attacks. For instance, in another study [73], the researchers obtained an NC value of 1 in the Gaussian Filter attack, which was remarkable. Still, considered the attacks overall, our proposed approach clearly emerges as a viable alternative watermarking architecture, outperforming those reported in previous studies [66, 69, 79]. Table 9 displays the results of the comparisons of the proposed method to other current watermarking approaches using gradient-based edge detection techniques.

The primary contribution of this study is the introduction of SNNs, which are compatible with the HVS and commonly used in edge detection approaches, to image watermarking systems for the first time. Additionally, the results of this study demonstrate that the proposed SNN-based edge detection model enhances the achievement of watermarking. The most significant advantage of the employed SNN model is its ability to detect edges without prior information, thus eliminating the need for a learning process involving large datasets. Another benefit of the proposed method is that it involves fewer computations compared to existing SNN models. The initial examples of existing SNN models [35, 36] usually used a 5×5 *RF*. However, in recent years [41, 64], 3×3 *RF*s have been employed. While the proposed model also uses a 3×3 *RF*, it further reduces the number of computations by decreasing the number of pixels used within the *RF*, resulting in fewer operations required than current models.

Table 9 presents the results of the comparative evaluation of the proposed SNN-based edge detection watermarking method against three recently published gradient-based edge detection techniques. The evaluation is conducted using the PSNR and SSIM metrics. The results reveal that the proposed method consistently outperforms the reference

approaches by yielding substantially higher PSNR and SSIM values across all test cases. Specifically, the method achieves an average PSNR of 54.14 dB and an average SSIM of 0.9994, demonstrating its ability to insert watermarks with high accuracy while maintaining exceptional visual quality. The superior performance of the proposed approach can be attributed to the precise and noise-robust edge information extracted via the SNN-based edge detection process, which plays a critical role in preserving both data integrity and visual fidelity during watermark embedding. While some previous studies [17, 18] utilized a 128 × 128 watermark image due to capacity limitations, the proposed method is tested with a 256 × 256 watermark image. Still, the proposed technique exhibits a notably superior performance. However, the Canny edge detection algorithm based on fixed thresholds was used in one of the above-mentioned studies [17]. This method may have difficulty in detecting contrast differences in different regions of the image with the same sensitivity, which may affect the accuracy of edge detection in various images. Additionally, the LSB method has low robustness and may be vulnerable to the compression, rotation, or scaling of the image, which may distort the embedded watermark. Similarly, in another study [18], a combination of the Compass edge detector and LSB was used. As the Compass edge detector relies on fixed threshold values, its performance in edge detection may become inconsistent when faced with varying image characteristics. The reliability of the watermark may decrease under different lighting conditions, and the watermark embedded in the image may exhibit vulnerability to certain types of attacks. Additionally, the performance of the pixel similarity-based approach developed by Gürkahraman et al. in 2023 [30] was lower than that of the proposed method. The main distinction of the proposed method from the one reported by Gürkahraman et al. [30] is that it uses an SNN-based model, which simulates the HVS, to determine pixel similarities through grayscale differences. Therefore, it is evident that the achievement of the proposed method is largely attributed to SNN-based edge detection. In contrast to the limitations in these studies, SNN-based watermarking offers significant advantages over conventional methods.

Figure 9 presents the comparative analysis of watermarking studies performed on 20 images of 512 × 512 size in the SIPI database in terms of average PSNR and SSIM metrics. In Figure 9(a), it is seen that the PSNR values of conventional gradient-based edge detection methods [17, 18, 30] were around 40 dB. On the other hand, the proposed method provides a significant advantage over other methods with a PSNR value above 50 dB. This shows that the SNN-based edge detection method causes less distortion during the watermarking process and preserves the image quality better. Similarly, looking at the SSIM metric results in Figure 9(b), it is seen that conventional methods had SSIM values ranging from 0.98 to 0.985, while the proposed method offers an SSIM value above 0.99. This finding shows that the SNN-based edge detection method preserves image details better and maintains structural similarity at a higher-level during watermarking.

Tables 10 and 11 show the NC values of traditional gradient-based methods and the proposed SNN-based approach for various attacks in Lena and Peppers images, respectively. In addition, Figure 10 illustrates the average NC values for SIPI. The evaluation results clearly show that the proposed method is superior to the traditional methods in terms of both the overall average success and the minimum success level. Over the 19 different attack scenarios examined, the proposed method achieved the highest NC score in almost all cases and maximized its robustness by reaching 0.9900 in many attack scenarios. This shows that the proposed method can protect the watermark information in a highly resilient way against corruption. Among other methods [30], produced relatively better results, outperforming [17] and [18] on many datasets. However, this method also shows significant performance degradation in some attacks. The methods in [17] and [18], on the other hand, exhibited lower performance in general. Overall, the proposed method clearly outperforms the proposed method in terms of robustness. The competitive performance of the proposed method is promising.

There are several reasons for this superior performance of the SNN-based method. First of all, SNNs detect differences between pixels in a way that simulates biological neural networks and are based on spike generation. While conventional edge detection methods directly focus on gradient changes, SNN-based approaches can produce results more similar to the HVS. Moreover, while conventional gradient-based methods may detect edges incorrectly, especially in images containing high-frequency noise, SNNs do not create unnecessary edges by reacting only to significant changes, thus preserving image details during edge detection. Additionally, SNNs provide a more robust edge map and thus less distortion in the embedded data. As a result, the obtained findings show that the SNN-based edge detection method significantly improves watermarking performance compared to conventional gradient-based methods. In particular, the high PSNR and SSIM values obtained in this study prove that this method preserves the image quality better and causes less degradation after the watermarking process. In light of this information, it can be stated that the SNN-based edge detection method should be utilized as a new alternative in the field of watermarking, and it is important to verify its effectiveness with more comprehensive studies using different datasets and under different attack scenarios in the future.

Table 6 UIQI and NC metrics of the watermarks extracted from the watermarked images under various attack scenarios

	Attack İndex	No Attack	a	b	c	d	e	f	g	h	i	j	k	l	m	*n*	o	*p*	q	*r*
Lake	UIQI	0.9779	0.9162	0.8960	0.9066	0.9729	0.9671	0.9704	0.9655	0.9815	0.9818	0.9732	0.9744	0.9658	0.9687	0.9762	0.9927	0.9642	0.9999	0.9671
	NC	0.9999	1.0000	0.9996	0.9997	0.9998	0.9997	0.9998	0.9998	1.0000	1.0000	0.9998	0.9999	0.9997	0.9853	0.9599	0.9965	0.9832	1.0000	0.9846
Lena	UIQI	0.9777	0.9901	0.8731	0.8947	0.9727	0.9679	0.9732	0.9710	0.9829	0.9853	0.9703	0.9716	0.9658	0.9615	0.9756	0.9961	0.9550	0.9999	0.9702
	NC	0.9996	0.9993	0.9990	0.9996	0.9998	0.9996	0.9998	0.9998	1.0000	1.0000	0.9997	0.9999	0.9996	0.9770	0.9899	0.9982	0.9796	1.0000	0.9865
Boats	UIQI	0.9774	0.9045	0.8758	0.9117	0.9704	0.9627	0.9680	0.9661	0.9835	0.9852	0.9681	0.9662	0.9602	0.9321	0.9752	0.9971	0.9352	0.9999	0.9653
	NC	0.9999	0.9999	0.9993	0.9998	0.9997	0.9995	0.9997	0.9997	1.0000	1.0000	0.9997	0.9999	0.9995	0.9357	0.9599	0.9986	0.9688	1.0000	0.9833
Peppers	UIQI	0.9724	0.9013	0.8677	0.8892	0.9688	0.9595	0.9680	0.9666	0.9827	0.9851	0.9715	0.9622	0.9518	0.9043	0.9723	0.9937	0.9617	0.9998	0.9636
	NC	0.9995	0.9992	0.9964	0.9996	0.9997	0.9992	0.9994	0.9996	0.9999	0.9998	0.9998	0.9998	0.9997	0.9885	0.9995	0.9970	0.9822	0.9999	0.9931
Plane	UIQI	0.9771	0.9236	0.9050	0.9178	0.9614	0.9404	0.9624	0.9336	0.9828	0.9796	0.9623	0.9491	0.9322	0.9160	0.9735	0.9960	0.7635	0.9999	0.9171
	NC	0.9999	1.0000	0.9999	1.0000	0.9996	0.9994	0.9996	0.9996	1.0000	1.0000	0.9995	0.9999	0.9993	0.9856	0.9998	0.9981	0.8866	1.0000	0.9610
Mandrill	UIQI	0.9776	0.9055	0.8875	0.8773	0.9523	0.9235	0.9428	0.9344	0.9876	0.9848	0.9404	0.9275	0.9134	0.8978	0.9605	0.9895	0.8512	0.9999	0.9675
	NC	1.0000	1.0000	0.9998	0.9996	0.9993	0.9985	0.9990	0.9992	1.0000	1.0000	0.9989	0.9997	0.9983	0.7980	0.9996	0.9946	0.9232	1.0000	0.9801
Couple	UIQI	0.9776	0.8990	0.8615	0.8899	0.9652	0.9526	0.9639	0.9599	0.9766	0.9853	0.9644	0.9472	0.9427	0.9510	0.9731	0.9972	0.9443	0.9999	0.9678
	NC	0.9999	0.9998	0.9990	0.9993	0.9996	0.9994	0.9996	0.9996	1.0000	1.0000	0.9996	0.9998	0.9993	0.9967	0.9998	0.9986	0.9725	1.0000	0.9841
4.02.1	UIQI	0.9999	0.9905	0.9240	0.7422	0.9644	0.9644	0.9657	0.9650	0.8259	0.9281	0.9298	0.8524	0.9639	0.9488	0.7982	0.9970	0.9572	0.9993	0.9649
	NC	0.9999	0.9957	0.8549	0.8810	0.9840	0.9840	0.9845	0.9842	0.9194	0.9666	0.9569	0.8853	0.9837	0.9770	0.8206	0.9986	0.9807	0.9997	0.9842
5.02.9	UIQI	0.9998	0.9866	0.7667	0.9806	0.8054	0.8037	0.7600	0.7213	0.7544	0.8286	0.9492	0.9145	0.9557	0.8678	0.8695	0.9989	0.9189	0.8658	0.8971
	NC	0.9999	0.9934	0.8862	0.9905	0.7649	0.7643	0.8811	0.8606	0.8796	0.9155	0.9780	0.9411	0.9783	0.9349	0.8822	0.9994	0.9601	1.0000	0.9495
5.02.10	UIQI	0.9998	0.8533	0.9257	0.9274	0.8545	0.8542	0.9562	0.9298	0.9657	0.9873	0.9698	0.9598	0.9498	0.9298	0.9598	0.9498	0.9698	0.9598	0.9698
	NC	0.9999	0.9952	0.9916	0.9915	0.9834	0.9835	0.9846	0.9841	0.9419	0.9799	0.9999	0.9998	0.9829	0.9799	0.9899	0.9563	0.9699	0.9997	0.9799
7.01.1	UIQI	0.9999	0.8362	0.7540	0.7924	0.9114	0.9101	0.7962	0.9329	0.7247	0.8988	0.9265	0.8559	0.9573	0.9098	0.8006	0.9938	0.9416	0.9999	0.7655
	NC	0.9999	0.8113	0.8765	0.8953	0.8315	0.8310	0.8961	0.8404	0.8618	0.9482	0.9568	0.8858	0.9785	0.9546	0.7881	0.9969	0.9706	0.9999	0.8801
7.01.2	UIQI	0.9997	0.9938	0.9852	0.9883	0.7240	0.7232	0.9717	0.9713	0.9912	0.9089	0.9286	0.8607	0.9687	0.9186	0.8009	0.9954	0.9542	0.9999	0.9716
	NC	0.9998	0.9978	0.9948	0.9959	0.8988	0.8985	0.9900	0.9899	0.9969	0.9663	0.9568	0.8858	0.9890	0.9713	0.8127	0.9984	0.9839	1.0000	0.9900
7.01.3	UIQI	0.9999	0.8100	0.7515	0.7395	0.8098	0.8092	0.9467	0.9450	0.8383	0.8640	0.9314	0.8632	0.9435	0.8538	0.8014	0.9904	0.9195	0.9998	0.9464
	NC	0.9999	0.7978	0.8715	0.8657	0.7493	0.7490	0.9722	0.9713	0.8074	0.9284	0.9569	0.8858	0.9705	0.9235	0.7881	0.9950	0.9580	0.9999	0.9721
7.01.4	UIQI	0.9999	0.8140	0.7411	0.7566	0.9316	0.9306	0.8061	0.9550	0.7435	0.9060	0.9488	0.9034	0.8568	0.9186	0.8554	0.9925	0.9395	0.9998	0.7605
	NC	0.9999	0.7989	0.8667	0.8743	0.8342	0.8338	0.8974	0.9765	0.8673	0.9504	0.9782	0.9410	0.8030	0.9574	0.8822	0.9960	0.9683	0.9999	0.8729
7.01.5	UIQI	0.9999	0.8456	0.7748	0.8206	0.7808	0.7881	0.8632	0.9519	0.7255	0.8151	0.9435	0.8915	0.9497	0.7796	0.8672	0.9920	0.9139	0.9999	0.8483
	NC	0.9999	0.8137	0.8855	0.9082	0.7227	0.7226	0.7743	0.9754	0.8612	0.9050	0.9779	0.9410	0.9744	0.8866	0.8910	0.9959	0.9560	1.0000	0.8052
7.01.6	UIQI	0.9999	0.8386	0.7612	0.8666	0.7223	0.7572	0.8706	0.9496	0.7206	0.8177	0.9200	0.8730	0.9476	0.7891	0.8560	0.9908	0.9130	0.9999	0.8264
	NC	0.9999	0.8097	0.8775	0.9308	0.7146	0.7145	0.7751	0.9740	0.8574	0.9053	0.9772	0.9409	0.9731	0.8906	0.8908	0.9953	0.9551	1.0000	0.7940
7.01.7	UIQI	0.9999	0.8148	0.7593	0.8056	0.7002	0.8000	0.8853	0.9464	0.8422	0.8227	0.9414	0.8933	0.9445	0.7988	0.8676	0.9916	0.9114	0.9999	0.9486
	NC	0.9999	0.8004	0.8758	0.8990	0.7037	0.7036	0.7790	0.9721	0.8098	0.9071	0.9778	0.9411	0.9711	0.8944	0.8910	0.9956	0.9538	0.9999	0.9732
7.01.8	UIQI	0.9999	0.8196	0.7519	0.7500	0.8349	0.8350	0.9531	0.9520	0.8866	0.8852	0.9504	0.9056	0.8453	0.8763	0.8665	0.9950	0.9318	0.9998	0.9529
	NC	0.9999	0.8092	0.8780	0.8772	0.8055	0.8055	0.9769	0.9764	0.8009	0.9425	0.9781	0.9406	0.7754	0.9390	0.8908	0.9975	0.9664	0.9999	0.9768
7.01.9	UIQI	0.9999	0.8300	0.7579	0.7631	0.8139	0.8132	0.9273	0.9532	0.7587	0.8548	0.9471	0.9094	0.9516	0.8570	0.8691	0.9955	0.9274	0.9999	0.9551
	NC	0.9999	0.8057	0.8755	0.8780	0.7507	0.7504	0.8334	0.9757	0.8755	0.9239	0.9780	0.9411	0.9749	0.9255	0.8555	0.9977	0.9622	0.9999	0.9767
7.01.10	UIQI	0.9999	0.8994	0.7382	0.7513	0.9208	0.9195	0.8023	0.9353	0.7078	0.9024	0.9521	0.9171	0.8432	0.9191	0.8700	0.9902	0.9378	0.9997	0.7506
	NC	0.9999	0.7950	0.8632	0.8697	0.8269	0.8264	0.8937	0.8327	0.8477	0.9476	0.9781	0.9411	0.7940	0.9570	0.8555	0.9948	0.9669	0.9999	0.8653

Table 6 (continued)

	Attack İndex	No Attack	a	b	c	d	e	f	g	h	i	j	k	l	m	*n*	o	*p*	q	*r*
Average	UIQI	0.9918	0.8886	0.8279	0.8486	0.8769	0.8791	0.9127	0.9403	0.8681	0.9153	0.9494	0.9149	0.9355	0.8949	0.8944	0.9918	0.9255	0.9978	0.9138
	NC	0.9999	0.9111	0.9295	0.9427	0.8784	0.8781	0.9318	0.9655	0.9163	0.9593	0.9824	0.9534	0.9572	0.9429	0.9073	0.9949	0.9624	0.9999	0.9441

4.3 More discussion

One of the most important problems encountered in watermarking techniques is the false-positive problem (FPP). FPP is the extraction of an unembedded or false watermark from the host image. Many SVD-DWT based techniques have high probability of FPP vulnerability [80]. However, FPP is eliminated by scrambling the watermark logo image using a chaotic map [51, 81, 82].

In this study, a chaotic substitution box (S-Box) created with the help of PWLCM is used in both insertion and extraction stages. For the effect of the S-Box on the FPP vulnerability, tests were performed with two different S-Boxes. S-Box1 contains the real values used in the extraction and insertion process, while S-Box2 represents the fake S-Box used by a malicious attacker. The general structure of the tests is shown in Figure 11. In the first test in Figure 11 (a), S-Box1 is used for both insertion and extraction. The resulting logo is visually almost identical to the inserted logo. The UIQI and NC values of the resulting logo are 0.9901 and 0.9999 respectively. In the second test in Figure 11 (b), insertion was performed with S-Box1 and extraction with S-Box2. Due to the use of different S-boxes, it seems unlikely that the logo could be extracted in a visually meaningful way by someone else. The extracted logo image in Figure 11 (b) is both visually meaningless and has poor image quality (UIQI =0.0414, NC =0.7250). In the third test in Figure 11 (c), insertion was performed with S-Box2 and extraction with S-Box1. In this case, it can be seen that a logo inserted by an attacker in a host image cannot be extracted in a visually understandable way and the image quality is very low (UIQI =0.0345, NC =0.7429). Finally, in the fourth test in Figure 11 (d), the watermark logo was extracted from an image without a logo. In this case, no visually meaningful logo image is obtained. The obtained results show that the proposed method has low vulnerability to FPP.

The average times taken for the insertion and extraction of watermarks for an image in the tests are given in Table 12 for different techniques. With the DWT-SVD-based technique [13], which did not include any edge detection process, the watermarking insertion-extraction process was completed in approximately 0.2262 s for an image. While this time was 0.4173 s for the technique combining the gradient-based edge approach with SVD-DWT [30], the process takes 0.68 s with the proposed method. Of course, the SNN-based edge detection approach has a computational cost. However, according to these tests performed with a 4th-generation i7 processor, the computational cost of the proposed approach does not significantly affect its execution time. Additionally, the overall computational complexity for an SNN-based edge detection model is expressed as

Table 7 Comparison of NC values of the extracted watermark for Lena under various attack scenarios

	[66]	[79]	[69]	[70]	[73]	[74]	[28]	[13]	Proposed
a	0.6362	0.9614 (0.005)	0.832 (0.05)	0.9983 (0.001)	0.939 (0.005)	0.8142	0.9769 (0.001)	0.9995	0.9993
b	-	0.9309(0.03)	0.9141(0.01)	0.9868(0.01)	0.983	0.8386(0.02)	0.9502(0.005)	0.9989	0.9990
c	-	0.8451(0.04)	0.9102(0.01)	0.9955(0.01)	0.980(0.005)	0.8075	0.9631(0.005)	0.9993	0.9996
d	-	0.9346	0.957	0.9959	-	1	0.9754	0.9995	0.9998
e	-	0.9346	0.8633	-	-	0.9641	-	0.9993	0.9996
f	-	0.9377	0.9258	0.9971	0.956	0.9967	0.9928	0.9995	0.9998
g	-	-	-	-	-	-	-	0.9995	0.9998
h	0.9776	1	0.9336	0.8176	0.962	0.9953	0.9495	0.9998	1.0000
i	0.4561	1	0.9648	0.9891	-	0.9993	-	0.9998	1.0000
j	0.981	-	-	0.7637	-	-	-	0.9994	0.9997
k	-	0.4128(10.10)	0.7891(2.15)	-	-	-	-	0.9996	0.9999
l	−0.6178	0.9922	0.9453	0.9977	0.6408	0.9987	0.9946	0.9993	0.9996
m	-	0.9628	0.9218	0.9841	-	-	-	0.9978	0.9770
n	-	0.8048	0.9922(10%)	-	-	0.9574	0.9964(1/16)	0.9996	0.9899
o	-	0.9972	0.9863	0.9686	-	-	0.9912	0.9980	0.9982
p	-	0.9576	0.9645	-	-	-	-	0.9801	0.9796
q	0.8977	0.9883	0.9874	-	-	0.9799	0.9897	0.9999	1.0000
r	0.9002	0.9749	0.9788	0.9615	-	0.9712	0.9789	0.9812	0.9865

The horizontal line shows that there is no data. The values in parentheses indicate the constraints of the attacks

Table 8 Comparison of NC values of the extracted watermark for pepper under various attack scenarios

	[66]	[79]	[69]	[70]	[73]	[74]	[28]	[13]	Proposed
a	0.6684	0.8928(0.005)	0.8008(0.05)	0.9986(0.001)	0.904	0.8052	0.9646(0.001)	0.9995	0.9992
b	-	0.8883(0.03)	0.9102(0.01)	0.9782(0.01)	0.982	0.8544(0.02)	0.9611(0.005)	0.9988	0.9964
c	-	0.8398(0.04)	0.8906(0.01)	0.9964(0.01)	0.995(0.005)	0.8166	0.9982	0.9992	0.9996
d	-	0.9536	0.9609	0.9955	-	1	0.9498	0.9994	0.9997
e	-	-	0.8594	-	-	0.9723	-	0.9992	0.9992
f	-	0.919	0.918	0.998	-	0.9765	0.9911	0.9994	0.9994
g	-	-	-	-	-	-	-	0.9994	0.9996
h	0.8968	0.9743	0.9375	0.8579	-	0.996	1	0.9998	0.9999
i	0.9926	0.9883	0.9766	0.9877	-	0.9993	-	0.9999	0.9998
j	0.9921	-	-	0.7647	-	-	-	0.9995	0.9998
k	-	0.4712(10.10)	0.7773(2.15)	-	-	-	-	0.9996	0.9998
l	−0.5945	0.9072	0.9414	0.9942	-	0.9987	0.9786	0.9992	0.9997
m	-	0.9369	0.9248	0.9537	-	-	-	0.9799	0.9885
n	-	0.8544	0.9727(10%)	-	-	0.9588	0.9302(1/16)	0.9996	0.9995
o	-	0.9845	0.9752	0.9878	-	-	0.9902	0.9975	0.9970
p	-	0.9687	0.9568	-	-	-	-	0.9801	0.9822
q	0.8754	0.9785	0.9799	-	-	0.9786	0.9912	1.0000	0.9999
r	0.8999	0.9631	0.9721	0.9458	-	0.9634	0.9752	0.9795	0.9931

The horizontal line shows that there is no data. The values in parentheses indicate the constraints of the attacks

$O(n) = M\text{x}N\text{x}T\text{x}R$ [83]. Here, MxN is the number of pixels in the image, T is the number of observed steps, and R is the RF size. The proposed model using a 3 ×3 RF (R= 9) reduces the computational cost by approximately 2.5 times compared to models involving 5× 5 RFs (R= 25) [34, 40, 84].

$$bpp = \frac{B}{MxN} \tag{15}$$

The calculation of the bit per pixel (*bpp*) metric regarding how many bits of data can be stored on average for a pixel in the host image is shown in Eq. 15. In the equation, B is the number of bits in the watermark logo. In this study, it is 256 ×256 ×8. Additionally, M and N are the numbers of rows and columns in the host image, respectively. In the proposed approach, the pixels of the 256 ×256 logo image are placed on the SNN-based edge pixels obtained for all host images. For this purpose, the number of determined edge pixels must be at least 256 ×256. To ensure this, pixels with the highest gray value after the SNN-based edge detection process are

Table 9 Watermarking performance comparisons of gradient-based algorithms and the proposed approach for watermarked images

	[17] (logo 128 × 128)		[18] (logo 128 × 128)		[30] (logo 256 × 256)		Proposed (logo 256 × 256)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Lake	44.69	0.9834	42.48	0.9860	41.88	0.9860	54.87	0.9999
Lena	44.48	0.9789	42.41	0.9840	41.89	0.9840	54.87	1.0000
Boat	44.31	0.9830	42.12	0.9856	41.53	0.9856	54.87	1.0000
Peppers	44.37	0.9820	41.94	0.9791	41.20	0.9790	54.87	1.0000
Plane	44.41	0.9794	41.91	0.9887	41.38	0.9888	54.88	1.0000
Mandrill	44.55	0.9926	42.76	0.9916	42.17	0.9916	54.87	0.9998
Couple	44.70	0.9834	42.49	0.9861	41.89	0.9861	54.87	1.0000
4.02.01	44.15	0.9813	41.96	0.9839	41.38	0.9839	52.59	0.9971
5.02.09	44.32	0.9839	42.13	0.9865	41.54	0.9865	54.65	1.0000
5.02.10	43.66	0.9837	41.50	0.9864	40.92	0.9864	53.83	1.0000
7.01.01	43.81	0.9826	41.65	0.9852	41.07	0.9852	54.03	0.9993
7.01.02	45.95	0.9833	43.68	0.9859	43.07	0.9859	56.66	1.0000
7.01.03	42.95	0.9819	40.83	0.9845	40.25	0.9845	52.96	0.9986
7.01.04	43.43	0.9820	41.28	0.9847	40.70	0.9847	53.55	0.9987
7.01.05	43.49	0.9831	41.34	0.9858	40.76	0.9858	53.63	0.9998
7.01.06	43.31	0.9830	41.17	0.9857	40.60	0.9857	53.41	0.9997
7.01.07	43.08	0.9827	40.95	0.9853	40.38	0.9853	53.12	0.9994
7.01.08	43.33	0.9815	41.19	0.9841	40.61	0.9841	53.43	0.9982
7.01.09	43.43	0.9828	41.28	0.9854	40.70	0.9854	53.55	0.9995
7.01.10	43.29	0.9820	41.15	0.9846	40.58	0.9846	53.38	0.9987
Average	43.98	0.9828	41.81	0.9854	41.22	0.9854	54.14	0.9994

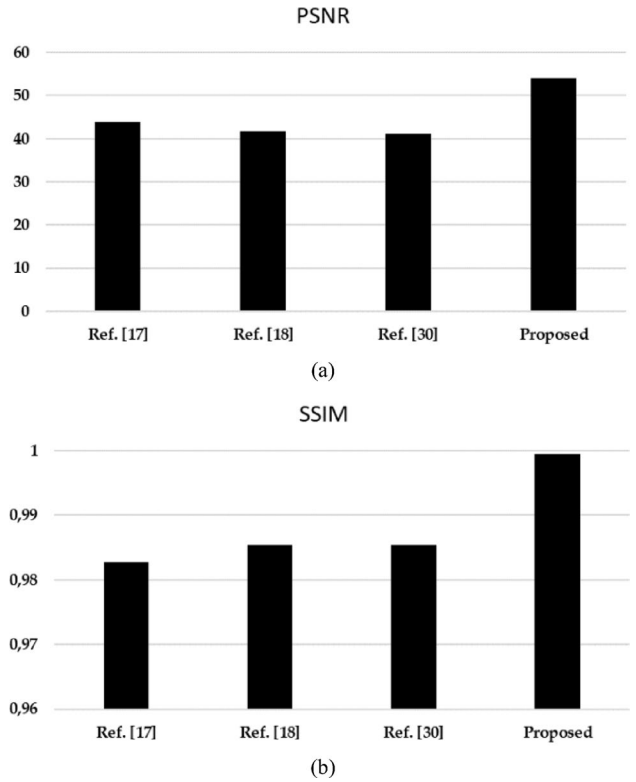

Fig. 9 Evaluation of performance values of gradient-based methods and proposed method: (**a**) PSNR, (**b**) SSIM

Table 10 Comparison of NC values of the gradient based approaches for Lena under various attack scenarios

	[17]	[18]	[30]	Proposed
a	0.9866	0.9825	0.9786	0.9993
b	0.9981	0.9947	0.9968	0.9990
c	0.9425	0.9712	0.9927	0.9996
d	0.9788	0.9798	0.9875	0.9998
e	0.9814	0.9899	0.9853	0.9996
f	0.9955	0.9941	0.9992	0.9998
g	0.9936	0.9990	0,9987	0.9998
h	0.9987	0.9972	0,9998	1.0000
i	0.9999	1.0000	1.0000	1.0000
j	0.9999	0.9999	0.9999	0.9997
k	0.9990	0.9899	0.9993	0.9999
l	0.9899	0.9901	0.9989	0.9996
m	0.9707	0.9736	0.9896	0.9770
n	0.9728	0.9699	0.9814	0.9899
o	0.9897	0.9901	0.9912	0.9982
p	0.9601	0.9699	0.9698	0.9796
q	0.9999	0.9999	0.9997	1.0000
r	0.9921	0.9900	0.9901	0.9865
s	0.9598	0.9688	0.9694	0,9845

marked as edges, provided that their number is not smaller than the pixel number of the logo image. Thus, according to the formula in Eq. 15, $bpp \geq 2$ is guaranteed.

By simulating biological neuron behaviors, SNNs can dynamically adapt to intensity variations in different regions of an image. This capability enables a dynamic edge detection process rather than one relying on fixed threshold

Table 11 Comparison of NC values of the gradient based approaches for peppers under various attack scenarios

	[17]	[18]	[30]	Proposed
a	0.9699	0.9700	0.9712	0.9992
b	0.9997	0.9958	1.0000	0.9964
c	0.9999	0.9998	1.0000	0.9996
d	0.9699	0.9878	0.9724	0.9997
e	0.9782	0.9800	0.9799	0.9992
f	0.9801	0.9790	0.9899	0.9994
g	0.9800	0.9902	0.9954	0.9996
h	0.9912	0.9932	0.9991	0.9999
i	0.9899	0.9985	1.0000	0.9998
j	0.9968	0.9832	0.9875	0.9998
k	0.9909	0.9724	0.9814	0.9998
l	0.9860	0.9990	0.9986	0.9997
m	0.9798	0.9847	0.9901	0.9885
n	0.9889	0.9874	0.9899	0.9995
o	0.9888	0.9957	0.9919	0.9970
p	0.9902	0.9971	0.9999	0.9822
q	0.9990	0.9989	0.9999	0.9999
r	0.9890	0.9901	0.9920	0.9931
s	0.9698	0.9682	0.9710	0.9736

values, enhancing generalization across a broader range of images. Due to all these advantages of the proposed SNN-based edge detection approach, it can tested in areas such as image segmentation, interpolation, and retrieval. However, it is anticipated that incorporating a learning process would enhance the performance of the proposed model when used in specific fields such as the medical sector [85] or satellite imagery [86]. Therefore, new studies are planned to include a learning process in different fields to increase the effectiveness of the method further.

In this paper, we propose a Spiking Neural Network (SNN) based approach to watermarking using edge-based and DWT-SVD. To the best of our knowledge, there is no comprehensive study on the use of SNNs in watermarking in the existing literature. Moreover, unlike Machine Learning techniques, the proposed SNN model does not involve a learning process and has a competitive computational time compared to traditional edge detection techniques, which is a great advantage for real-time applications.

Our method is evaluated in comparison with classical gradient-based stamping techniques. Experiments show that our proposed SNN-based approach offers competitive and in some cases superior robustness against attacks. This suggests that biologically inspired models can provide an effective alternative in areas such as media security. Furthermore, this work bridges the gap between existing SNN models and the field of digital watermarking, providing a new research direction.

Since the SNN model proposed in this study is suitable for working on a single color channel, all experiments are performed on gray-level images. Even though it is possible to apply the proposed method independently to 3 different color channels, it would not be an effective solution. Therefore, designing a model that can work on colored images is one of our plans for future studies. The proposed approach produces successful results by simulating the HVS in a simpler way than known Machine Learning (ML) techniques. Nevertheless, it is possible to increase the watermarking performance of the proposed method by adding a training phase. SNN-based edge detection models can work with training data [35]. The proposed SNN-based method for

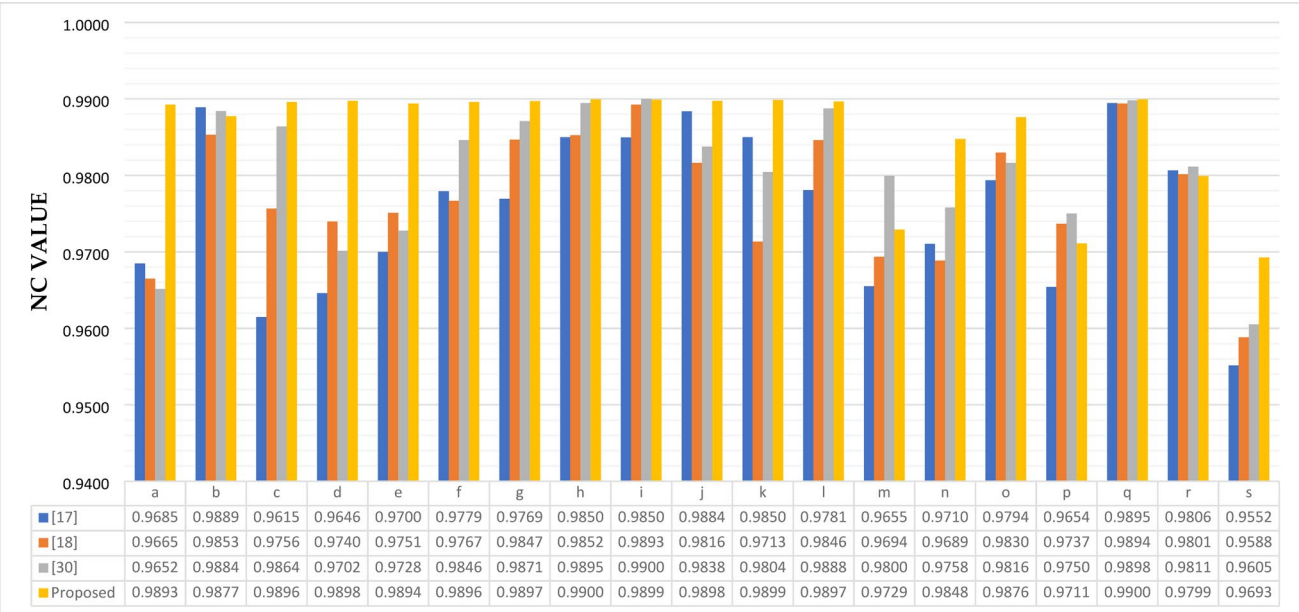

Fig. 10 Average NC values of the gradient-based approaches for the SIPI

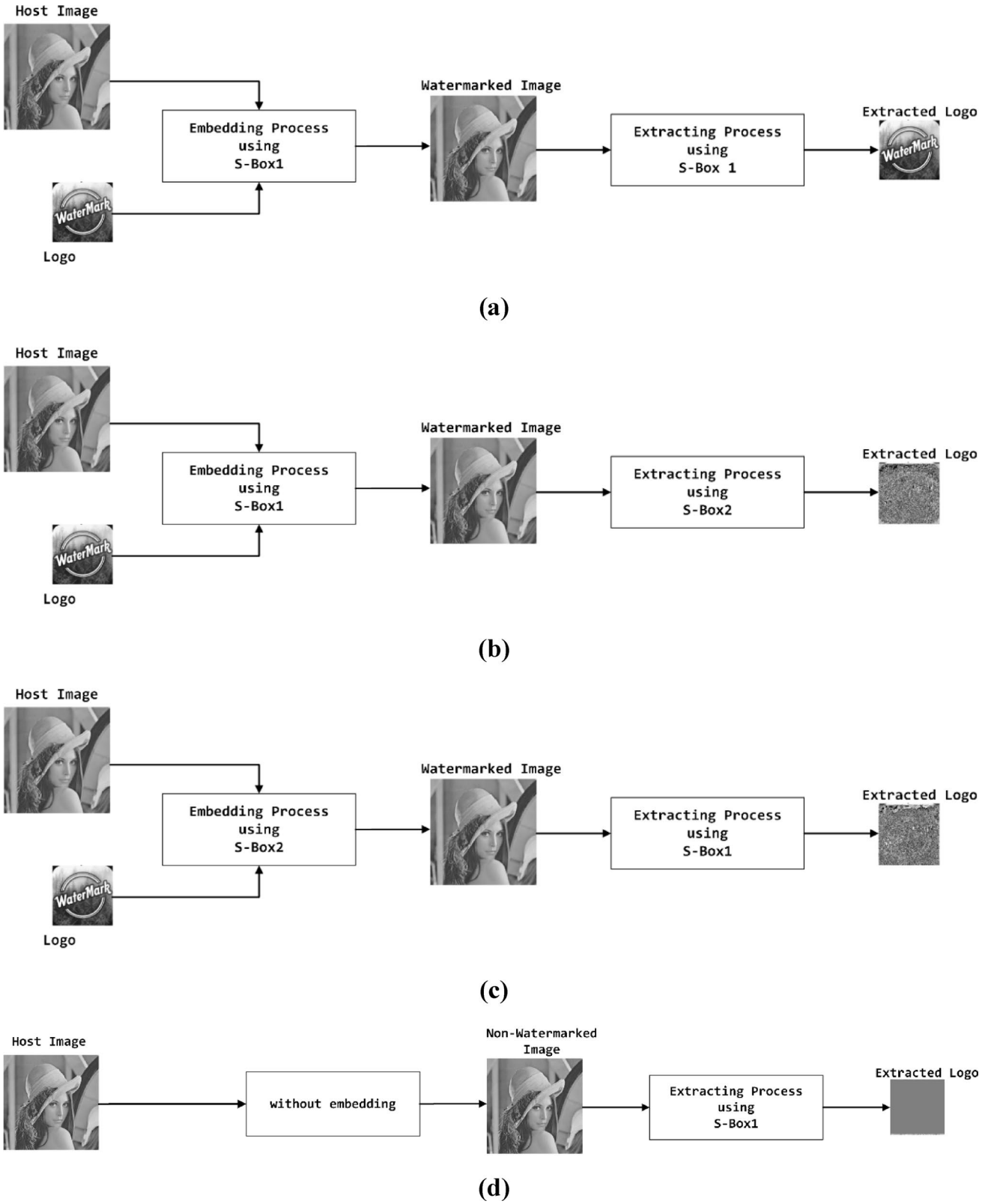

Fig. 11 FPP attack scenarios using different S-Boxes

Table 12 Comparison of mean computational time

Methods	Time (s)
[13]	0.2262
[30]	0.4173
Proposed	0.6812

edge detection can be integrated into watermarking techniques based on ML in the future.

5 Conclusion

In this study, a robust and imperceptible watermarking technique is proposed by detecting edges in digital images using Spiking Neural Networks (SNN). The proposed technique determines ideal regions for watermark placement by taking advantage of the edge detection capabilities of SNNs. Thus, the watermarking process demonstrates high robustness against frequently encountered image processing attacks such as rotation, noise addition, and cropping while preserving image quality. The experimental show that the proposed method has superior performance compared to conventional methods and is effective in preserving both image quality and watermark integrity. The findings emphasize the potential of SNNs in augmenting the security and reliability of digital content protection, setting the stage for future innovations.

The high imperceptibility success of the proposed method does not compromise image quality in the watermarking process. At the same time, being durable to attacks ensures that the added logo can be removed to a large extent despite various manipulations on the image. This shows that the proposed method can be used as an important tool in digital forensics cases such as unauthorized use and distribution, allowing leaks to be traced back to the original source.

The simplicity of the model, its design not requiring a training phase, and its relatively low computational cost are its prominent features. The experimental results reveal that the proposed method can be an alternative to gradient-based edge detection techniques used in image processing applications as well as edge-based watermarking studies. However, since the success of ML-based techniques in the literature is known, it is planned to increase the success of watermarking by integrating the proposed model into ML techniques. Additionally, revising the model to make it suitable for color images is among our plans for future studies.

Acknowledgements We thank the anonymous editor and reviewers for their useful suggestions.

Author contributions MK: conceptualization, methodology, software, writing original draft, validation. MOI: conceptualization, methodology, software, writing original draft.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).
This work has no funding source.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethics approval Not applicable.

Human participants and/or animals Not applicable.

References

1. Zhong, X., et al.: An automated and robust image watermarking scheme based on deep neural networks. IEEE Trans. Multimedia. **23**, 1951–1961 (2020)
2. Dhar, S., Sahu, A.K.: Digital to quantum watermarking: A journey from past to present and into the future. Comput. Sci. Rev. **54**, 100679 (2024)
3. Tiwari, A., Srivastava, V.K.: Image watermarking techniques based on Schur decomposition and various image invariant moments: A review. Multimedia Tools Appl. **83**(6), 16447–16483 (2024)
4. Su, Q., Chen, B.: An improved color image watermarking scheme based on Schur decomposition. Multimedia Tools Appl. **76**, 24221–24249 (2017)
5. Zhang, H., Wang, C., Zhou, X.: A robust image watermarking scheme based on SVD in the Spatial domain. Future Internet. **9**(3), 45 (2017)
6. Saboori, A., Hosseini, S.A.: *A novel non-blind watermarking scheme for color image using pca transform and histogram matching technique*. in *2016 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*. IEEE. (2016)
7. Su, Q., Yuan, Z., Liu, D.: An approximate Schur decomposition-based Spatial domain color image watermarking method. IEEE Access. **7**, 4358–4370 (2018)
8. Das, C., et al.: A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation. AEU-International J. Electron. Commun. **68**(3), 244–253 (2014)

9. Ko, H.-J., et al.: Robust and blind image watermarking in DCT domain using inter-block coefficient correlation. Inf. Sci. **517**, 128–147 (2020)
10. Roy, S., Pal, A.K.: A blind DCT based color watermarking algorithm for embedding multiple watermarks. AEU-International J. Electron. Commun. **72**, 149–161 (2017)
11. Jemal, M.A., Sallami, M.M., Ghorbel, F.: Robust watermarking method based on the analytical Clifford fourier mellin transform. Multimedia Tools Appl. **83**(9), 25901–25922 (2024)
12. Tavakoli, A., Honjani, Z., Sajedi, H.: Convolutional neural network-based image watermarking using discrete wavelet transform. Int. J. Inform. Technol. **15**(4), 2021–2029 (2023)
13. Naffouti, S.E., Kricha, A., Sakly, A.: A sophisticated and provably grayscale image watermarking system using DWT-SVD domain. Visual Comput. **39**(9), 4227–4247 (2023)
14. Zhang, X., et al.: A robust and high-efficiency blind watermarking method for color images in the Spatial domain. Multimedia Tools Appl. **82**(18), 27217–27243 (2023)
15. Karakış, R., et al.: A novel fuzzy logic-based image steganography method to ensure medical data security. Comput. Biol. Med. **67**, 172–183 (2015)
16. Faheem, Z.B., et al.: Image watermarking scheme using LSB and image gradient. Appl. Sci. **12**(9), 4202 (2022)
17. Faheem, Z.B., et al.: Image watermarking using least significant bit and canny edge detection. Sensors. **23**(3), 1210 (2023)
18. Faheem, Z.B., et al.: An edge inspired image watermarking approach using compass edge detector and LSB in cybersecurity. Comput. Electr. Eng. **111**, 108979 (2023)
19. Anitha, K.: *Digital Watermarking: Pathway To Secure!* In *Multimedia Watermarking: Latest Developments and Trends*, pp. 29–47. Springer (2024)
20. Baziyad, M., et al.: A robust DCT-based scheme for watermarking deep neural networks. Procedia Comput. Sci. **231**, 397–402 (2024)
21. Gul, E., Toprak, A.N.: Contourlet and discrete cosine transform based quality guaranteed robust image watermarking method using artificial bee colony algorithm. Expert Syst. Appl. **212**, 118730 (2023)
22. AlShaikh, M., Alzaqebah, M., Jawarneh, S.: Robust watermarking based on modified pigeon algorithm in DCT domain. Multimedia Tools Appl. **82**(2), 3033–3053 (2023)
23. Mohammed, A.O., et al.: A blind and robust color image watermarking scheme based on DCT and DWT domains. Multimedia Tools Appl. **82**(21), 32855–32881 (2023)
24. Naima, S., et al.: *Secure and imperceptible frequency-based watermarking for medical images.* Circuits, Systems, and Signal Processing,: pp. 1–22. (2024)
25. Yasmeen, F., Uddin, M.S.: An efficient watermarking approach based on LL and HH edges of DWT–SVD. SN Comput. Sci. **2**(2), 82 (2021)
26. Azizoglu, G., Toprak, A.N.: A novel reversible fragile watermarking method in DWT domain for tamper localization and digital image authentication. Biomed. Signal Process. Control. **84**, 105015 (2023)
27. Araghi, T.K., Megías, D.: Analysis and effectiveness of deeper levels of SVD on performance of hybrid DWT and SVD watermarking. Multimedia Tools Appl. **83**(2), 3895–3916 (2024)
28. Gong, L.-H., et al.: Robust and imperceptible watermarking scheme based on canny edge detection and SVD in the contourlet domain. Multimedia Tools Appl. **80**, 439–461 (2021)
29. Zhang, L., et al.: A novel zero-watermarking algorithm based on DWT and edge detection. In: Qui, P., Xiang, Y., Ding, Y., Li, D., Wang, L. (eds.) In 2011 4th International Congress on Image and Signal Processing. IEEE (2011)
30. Gürkahraman, K., Karakis, R., Takci, H.: A novel color image watermarking method with adaptive scaling factor using Similarity-Based edge region. Comput. Syst. Sci. Eng. **47**(1), 55–77 (2023)
31. Manjunath, B., Chellappa, R.: A unified approach to boundary perception: Edges, textures, and illusory contours. IEEE Trans. Neural Networks. **4**(1), 96–108 (1993)
32. Buhmann, J.M., Lange, T., Ramacher, U.: Image segmentation by networks of spiking neurons. Neural Comput. **17**(5), 1010–1031 (2005)
33. Ghosh-Dastidar, S., Adeli, H.: Spiking neural networks. Int. J. Neural Syst. **19**(04), 295–308 (2009)
34. Wu, Q., et al.: *Edge detection based on spiking neural network model*. in *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence: Third International Conference on Intelligent Computing, ICIC* Qingdao, China, August 21–24, 2007. Proceedings 3. 2007. Springer. (2007)
35. Wu, Q., et al.: A visual attention model based on hierarchical spiking neural networks. Neurocomputing. **116**, 3–12 (2013)
36. Kerr, D., et al.: A biologically inspired spiking model of visual processing for image feature detection. Neurocomputing. **158**, 268–280 (2015)
37. Kerr, D., Coleman, S., McGinnity, T.M.: Biologically inspired intensity and depth image edge extraction. IEEE Trans. Neural Networks Learn. Syst. **29**(11), 5356–5365 (2018)
38. Kerr, D., et al.: *Biologically inspired edge detection*. in. *11th International Conference on Intelligent Systems Design and Applications*. 2011. IEEE. (2011)
39. Yedjour, H., et al.: Edge detection based on Hodgkin–Huxley neuron model simulation. Cogn. Process. **18**, 315–323 (2017)
40. Vemuru, K.V.: Image edge detector with Gabor type filters using a spiking neural network of biologically inspired neurons. Algorithms. **13**(7), 165 (2020)
41. İncetaş, M.O.: Anisotropic diffusion filter based on spiking neural network model. Arab. J. Sci. Eng. **47**(8), 9849–9860 (2022)
42. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. **117**(4), 500 (1952)
43. Kerr, D., et al.: *Biologically inspired intensity and range image feature extraction*. in *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE. (2013)
44. Abd El-Latif, A.A., et al.: *Quantum-inspired cascaded discrete-time quantum walks with induced chaotic dynamics and cryptographic applications.* Scientific reports, 10(1): p. 1930. (2020)
45. Shannon, C.E.: Communication theory of secrecy systems. Bell Syst. Tech. J. **28**(4), 656–715 (1949)
46. Bin Faheem, Z., et al.: Highly dispersive substitution box (S-box) design using chaos. ETRI J. **42**(4), 619–632 (2020)
47. Ansari, I.A., Pant, M., Ahn, C.W.: Robust and false positive free watermarking in IWT domain using SVD and ABC. Eng. Appl. Artif. Intell. **49**, 114–125 (2016)
48. Hemalatha, J., et al.: *Contemporary Improvements in Enhancing the Watermarking Techniques: A Deep Perspective*, in *Multimedia Watermarking: Latest Developments and Trends*, pp. 125–136. Springer (2024)
49. Hosseini, S.A., Farahmand, P.: An attack resistant hybrid blind image watermarking scheme based on combination of DWT, DCT and PCA. Multimedia Tools Appl. **83**(7), 18829–18852 (2024)
50. Anand, A., Singh, A.K.: An improved DWT-SVD domain watermarking for medical information security. Comput. Commun. **152**, 72–80 (2020)
51. Liu, J., et al.: An optimized image watermarking method based on HD and SVD in DWT domain. IEEE Access. **7**, 80849–80860 (2019)
52. Singh, A.K., Dave, M., Mohan, A.: Wavelet based image watermarking: Futuristic concepts in information security. Proc. Natl. Acad. Sci. India Sect. A: Phys. Sci. **84**, 345–359 (2014)

53. Wan, W., et al.: A comprehensive survey on robust image watermarking. Neurocomputing. **488**, 226–247 (2022)
54. Muhammad, N., Bibi, N.: Digital image watermarking using partial Pivoting lower and upper triangular decomposition into the wavelet domain. IET Image Proc. **9**(9), 795–803 (2015)
55. Ye, X., et al.: A SIFT-based DWT-SVD blind watermark method against geometrical attacks. In: Wan, Y., Shao, L., Wang, L., Sun, J., Nan, J., Zhang, Q. (eds.) In: 2014 7th International Congress on Image and Signal Processing. IEEE (2014)
56. Mehta, R., Rajpal, N., Vishwakarma, V.P.: LWT-QR decomposition based robust and efficient image watermarking scheme using lagrangian SVR. Multimedia Tools Appl. **75**, 4129–4150 (2016)
57. Jane, O., Elbaşi, E.: A new approach of nonblind watermarking methods based on DWT and SVD via LU decomposition. Turkish J. Electr. Eng. Comput. Sci. **22**(5), 1354–1366 (2014)
58. Gaur, S., et al.: An adaptive block-based watermarking scheme using RDWT-SVD and particle swarm optimization. SN Comput. Sci. **4**(5), 654 (2023)
59. Wang, T.: *Digital image watermarking using dual-scrambling and singular value decomposition*. in *IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*. 2017. IEEE. (2017)
60. Bull, D.: Communicating Pictures: A Course in Image and Video Coding. Academic (2014)
61. Bhinder, P., Singh, K., Jindal, N.: Robust Image-Adaptive watermarking using hybrid strength factors. Wireless Pers. Commun. **135**(1), 201–231 (2024)
62. Chekira, C., et al.: Join security and block watermarking-based evolutionary algorithm and Racah moments for medical imaging. Biomed. Signal Process. Control. **96**, 106554 (2024)
63. Ayuba, S., Zainon, W.M.N.W.: Medical image watermarking: A survey on applications, approach and performance requirement compliance. Int. J. Multimedia Inform. Retr. **12**(2), 33 (2023)
64. İncetaş, M.O.: Image interpolation based on spiking neural network model. Appl. Sci. **13**(4), 2438 (2023)
65. Bagheri, B., Ahmadi, S., et al.: An intelligent and blind image watermarking scheme based on hybrid SVD transforms using human visual system characteristics. Visual Comput. **37**(2), 385–409 (2021)
66. Agoyi, M., Çelebi, E., Anbarjafari, G.: A watermarking algorithm based on chirp z-transform, discrete wavelet transform, and singular value decomposition. Signal. Image Video Process. **9**, 735–745 (2015)
67. Mohamed, M., El-Mohandes, A.: Hybrid DCT-DWT watermarking and IDEA encryption of internet contents. Int. J. Comput. Sci. Issues (IJCSI). **9**(1), 394–401 (2012)
68. Begum, M., Uddin, M.S.: Implementation of secured and robust DFT-based image watermark through hybridization with decomposition algorithm. SN Comput. Sci. **2**(3), 221 (2021)
69. Verma, V.S., Jha, R.K., Ojha, A.: Digital watermark extraction using support vector machine with principal component analysis based feature reduction. J. Vis. Commun. Image Represent. **31**, 75–85 (2015)
70. Wang, B., Zhao, P.: An adaptive image watermarking method combining SVD and Wang-Landau sampling in DWT domain. Mathematics. **8**(5), 691 (2020)
71. Khare, P., Srivastava, V.K.: A novel dual image watermarking technique using homomorphic transform and DWT. J. Intell. Syst. **30**(1), 297–311 (2020)
72. Tagesse Takore, T., Kumar, P.R., Lavanya Devi, G.: *A robust and oblivious grayscale image watermarking scheme based on edge detection, SVD, and GA*. in *Proceedings of 2nd International Conference on Micro-Electronics, Electromagnetics and Telecommunications: ICMEET*. 2018. Springer. (2016)
73. Hu, K., et al.: Robust and efficient image watermarking via EMD and dimensionality reduction. Visual Comput. **38**(6), 2153–2170 (2022)
74. Kang, X., et al.: A novel hybrid of DCT and SVD in DWT domain for robust and invisible blind image watermarking with optimal embedding strength. Multimedia Tools Appl. **77**, 13197–13224 (2018)
75. Kazemivash, B., Moghaddam, M.E.: A robust digital image watermarking technique using lifting wavelet transform and firefly algorithm. Multimedia Tools Appl. **76**, 20499–20524 (2017)
76. Kazemivash, B., Moghaddam, M.E.: A predictive model-based image watermarking scheme using regression tree and firefly algorithm. Soft. Comput. **22**, 4083–4098 (2018)
77. Moeinaddini, E., Afsari, F.: Robust watermarking in DWT domain using SVD and opposition and dimensional based modified firefly algorithm. Multimedia Tools Appl. **77**, 26083–26105 (2018)
78. Saha, A., et al.: On unique framework-based implementation of a novel image watermarking scheme. Multimedia Tools Appl. **83**(33), 78861–78878 (2024)
79. Ernawan, F., Kabir, M.N.: A block-based RDWT-SVD image watermarking method using human visual system characteristics. Visual Comput. **36**(1), 19–37 (2020)
80. Makbol, N.M., Khoo, B.E., Rassem, T.H.: Security analyses of false positive problem for the SVD-based hybrid digital image watermarking techniques in the wavelet transform domain. Multimedia Tools Appl. **77**, 26845–26879 (2018)
81. Suresh, G., et al.: False-positive-free SVD based audio watermarking with integer wavelet transform. Circuits Syst. Signal. Process. **41**(9), 5108–5133 (2022)
82. Gao, H., Chen, Q.: A robust and secure image watermarking scheme using SURF and improved artificial bee colony algorithm in DWT domain. Optik. **242**, 166954 (2021)
83. Li, X., et al.: A lightweight convolutional spiking neural network for fires detection based on acoustics. Electronics. **13**(15), 2948 (2024)
84. Kerr, D., et al.: *Spiking hierarchical neural network for corner detection*. in *International Conference on Neural Computation Theory and Applications*. SciTePress. (2011)
85. Anand, A., et al.: Authenticating and Securing healthcare records: A deep learning-based zero watermarking approach. Image Vis. Comput. **145**, 104975 (2024)
86. Xu, D., Ren, N., Zhu, C.: High-Resolution remote sensing image Zero-Watermarking algorithm based on blockchain and SDAE. IEEE J. Sel. Top. Appl. Earth Observations Remote Sens. **17**, 323–339 (2023)