IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.DOI

Exploring Cellular Automata Learning: An Innovative Approach for Secure and Imperceptible Digital Image Watermarking

IRAM KHURSHID BHAT[1], FASEL QADIR [1], MEHDI NESHAT [2,3] AND AMIR H. GANDOMI.[2,4*], (Senior, IEEE)

[1]P.G Department of Computer Sciences University of Kashmir, North Campus Delina, Baramulla, Jammu & Kashmir, 193103, India (e-mail: bhat.eram29@gmail.com, faselqadir@uok.edu.in)
[2]Faculty of Engineering and Information Technology, the University of Technology Sydney, Ultimo, NSW, Australia (e-mail: gandomi@uts.edu.au)
[3]Centre for Artificial Intelligence Research & Optimisation, Torrens University Australia, Brisbane, Australia(e-mail: mehdi.neshat@torrens.edu.au)
[4]University Research and Innovation Center, Obuda University, 1034 Budapest, Hungary

*Corresponding author: Amir H. Gandomi (e-mail: gandomi@uni-obuda.hu) .

ABSTRACT As technology and multimedia production have advanced, there has been a significant rise in attacks on digital media, resulting in duplicated, fraudulent, and altered data as well as the infringement of copyright laws. This paper presents a robust and secure digital image watermarking technique that has been implemented in the spatial domain and exploits the erratic and chaotic behavior of the powerful elementary cellular automata Rule 30. This research is motivated by the potential to incorporate dynamic computational models into the field of image security. We aim to strike a balance between the crucial characteristics of the watermarking system, i.e., imperceptibility, capacity, and robustness, in the suggested blind watermarking technique. In this approach, prior to embedding, the grayscale watermark image is downsized to its two Most Significant Bits (MSBs). In the following, the 2-MSBs watermark is encrypted using an ECA Rule 30 to level up the system's security attributes. Then, the host image is scrambled using ECA Rule 30 to distribute the watermark pixels throughout the host image and thus achieve the highest robustness against geometrical attacks. Finally, the encrypted watermark data is embedded into the scrambled host image using the ECA Rule 30-based embedding key. The proposed method performs better in terms of imperceptibility, capacity, and robustness when compared to several systems with similar competencies. The simulation's findings demonstrate strong imperceptibility, as evaluated by the Peak Signal-to-Noise Ratio (PSNR), which has an average value of 58.3735 dB and a high payload. The experimental outcomes, observed across a diverse range of standardized attack scenarios, unequivocally establish the ascendancy of the proposed algorithm over competing methodologies in the realm of image watermarking.

INDEX TERMS Watermarking, Elementary Cellular Automata, Rule 30, Security, Copyright Protection, Authentication, Robustness.

I. INTRODUCTION

MULTIMEDIA security has gained prominence due to the internet's tremendous development and widespread use. The easy access to the internet, increased use of social media, and influential developments in digital data modification and editing software have made unauthorized copying, editing, and manipulation of digital content so much easier, resulting in the necessity for digital content security and authentication. The watermarking of digital data is performed to accomplish objectives like the protection of copyright, the preservation of originality, the protection from illegal duplication, and the authentication of digital content [1]. Digital watermarking is a process that embeds digital data called watermark into the host document. The watermark is embedded in a manner that does not alter the host document and is invisible to the viewer. The embedded watermark is retrieved from the watermarked document using the extraction process. Digital watermarking ensures fortified security, copyright protection, tamper resistance plus

detection, verification of integrity, controlled copying, and monitored broadcasting of digital data [2].

There are many different ways, such as human perception, resistance, type of document, reversibility, and working domain, to categorize digital watermarking techniques into various groups [3]. According to human perception, watermarking techniques can be classified into two categories: visible watermarking and invisible watermarking [4]. The watermarks embedded using visible watermarking techniques can be perceived easily from the host image. Examples include company logos, QR codes, signatures, initials, etc., that are placed at the corners of the images or TV channels/videos. The watermarks embedded using invisible watermarking techniques are hidden in the host image to preserve imperceptibility and, therefore, are unnoticeable to the human eyes. Their applications include ownership confirmation, integrity management, and digital document authentication. Since invisible watermarks are made to be hard to spot and remove, they are often more robust than visible ones. However, their invisibility may make them less helpful at proving ownership or preventing unauthorized usage.

Based on the resistance level, invisible watermarking techniques can further be categorized into four groups: fragile, semi-fragile, robust, and hybrid approaches [3]. Fragile watermarking involves embedding a watermark in a host document in such a way that any slight modification to the host would cause the watermark to be destroyed or altered. Fragile watermarking schemes are not resistant at all; their goal is to detect tampering, verify the integrity, and authenticate the digital content. The watermarks embedded using semi-fragile watermarking approaches can resist some basic attacks/modifications like compression or resizing; nevertheless, they remain fragile to major attacks/modifications. Semi-fragile watermarks are frequently employed in cases where some content modifications are likely to happen, but any malicious or illegal modifications ought to be identified and prevented [5]. The robust watermarking approaches involve embedding a watermark in a host document in such a manner that it becomes challenging to alter or remove it even if the host undergoes standard operations like cropping, compression, filtering, scaling, or other types of modification. Robust watermarks are highly resistant and are intended to survive deliberate or accidental modifications to the watermarked content. The hybrid watermarking approach combines robust and fragile approaches to simultaneously offer verification of data integrity, data authentication, and copyright enforcement [3].

The digital watermarking procedure can be applied to different types of digital host documents like images, texts [6], audio, and videos to manage digital rights, protect copyright, and identify content. Digital image watermarking is a process that embeds information into the host image in a manner that the embedded information does not alter the host image and is imperceptible to the viewer, which is then retrieved from the watermarked image during the extraction phase. In the realm of digital watermarking, reversibility can be defined as the capability of the watermarking system to fully restore the original unwatermarked form of the host document after watermark embedment. Watermarking schemes are often categorized as irreversible watermarking and reversible watermarking [7] based on reversibility. The irreversible watermarking schemes cannot completely restore the host document's original form once the watermark has been embedded. In contrast, reversible watermarking schemes can fully restore the original form of the host document after watermark embedment and extraction.

Cellular automata (CA) are massively parallel computational architectures with the maximum level of granularity. It is a mathematical model that consists of the D-dimensional lattice of finite-state cells with local interaction. The system's evolution is dependent on the evolution of each of its individual components. Simple rules and structures can result in a wide range of unpredictable patterns, rendering cellular automata theory highly intriguing. For instance, the behavior of a Turing machine or any other CA can be simulated using the CA known as universal cellular automata. The CA is beneficial in a variety of applications due to its fundamental characteristics; the parallelism characteristic of CA makes it suitable for large-scale modeling and simulations. The complex structures, patterns, and behaviors produced by the global behavior of the cells in the cellular space are impossible to predict based on the individual behavior of cells. This characteristic of the CA is termed emergent behavior. The CA is scalable and can be easily scaled down or up according to the dimensions of the system being simulated; as a result, the CA is suitable for a variety of applications. It serves as a flexible and adaptable tool for simulation and modeling as it can simulate various systems by customizing the update rule and the initial values of the parameters to start up CA. The watermarking systems that are based on CA offer high privacy and security as they make watermark detection and removal extremely challenging, thereby making them ideal for preserving delicate and valuable data.

The motivation for exploring digital image watermarking with CA lies in its potential to significantly enhance core principles of image security. This research is driven by the aspiration to integrate advanced computational models into the field of image security, offering robustness, efficiency, and adaptability to diverse content. Leveraging CA's dynamic behavior and parallelism, alongside its chaotic nature, promises to provide robust watermark security against adversarial attacks. This study aims to investigate ECA Rule 30 for digital image watermarking. By leveraging the unique properties of Rule 30, we seek to develop a watermarking method that not only provides effective protection against unauthorized modifications but also ensures the integrity and authenticity of the embedded watermark. An extensive investigation that considered numerous pertinent sources and analytical techniques was conducted to accomplish this goal. The major contributions of this work are as follows:

- This paper presents a robust and secure digital image watermarking technique implemented in the spatial do-

main. It leverages the erratic and chaotic behavior of the elementary cellular automata Rule 30 to enhance the security and robustness of the watermarking system.
- The suggested blind watermarking technique tries to achieve an equilibrium amidst the pivotal watermarking system attributes, i.e., imperceptibility, capacity, and robustness, by downsizing the grayscale watermark image to its two Most Significant Bits (MSBs). By employing ECA Rule 30 to encrypt the 2 most significant bits (MSBs) of the watermark, the security aspect of the system is augmented. Scrambling the host image with ECA Rule 30 also distributes the watermark pixels, maximizing robustness against geometrical attacks.

The remaining portions of the article will be divided into multiple sections, each of which will focus on a different facet of the given problem. The following section of the literature review will discuss a detailed background of digital image watermarking techniques based on the working domain in which they are implemented. Section 3 introduces cellular automata, elementary cellular automata, and Rule 30. Section 4 discusses the methodology of the proposed scheme. This will be followed by section 5, where performance analysis and experimental discussion are presented. In section 6, we will discuss the results of our technique, compare and analyze these results, and draw conclusions based on our findings. The summary of the main findings and suggestions for further study will be included in the part that concludes the article. With this framework, we want to present a vivid and insightful exploration of cellular automata-based digital image watermarking.

II. RELATED WORK

The performance of the different digital image watermarking approaches depends on the domain in which they are being implemented. Based on the working domain, digital image watermarking algorithms can be classified into three groups: spatial domain watermarking, transform or frequency domain watermarking, and hybrid domain watermarking [8].

A. SPATIAL DOMAIN ALGORITHMS

In the spatial domain, the watermark information bits are embedded straight into the host image pixel values using various approaches like modification of Least Significant Bits (LSBs) [9] [10] [11] [12] [13] or Intermediate Significant Bits (ISBs) [14] of the host image, patchwork approach, Local Binary Pattern (LBP) approach [15], histogram modification approach [16] [17], and approaches based on correlation [18] [19] [20] and spread spectrum [21] [22] [23]. The spatial domain techniques are more straightforward, more effective, and execute more quickly. In terms of capacity, these techniques allow the embedding of significant data. However, these methods only work effectively when the image has not been altered by humans or subjected to noise. A significant flaw with spatial domain watermarking is that the watermark can be removed through image cropping. Furthermore, a tiny watermark may be embedded repeatedly. Therefore, despite losing the majority of the image data due to numerous attacks, a single watermark remaining will be viewed as a success.

A digital image scrambling technique-based image watermarking approach using CA has been presented in [24] by Ye and Li. In the reference [24], they embark upon the exploration of fractal box dimensions, unravelling the enigmatic properties of the CA. Employing an innovative technique, they harness the power of these dimensions to curate a selection of chaotic CA rules. With this selection in hand, they proceed to manipulate the host image, imbuing it with a sense of disarray using the chosen rule. In the culmination of the artistic endeavour, they delicately embed the watermark within the transformed image, leaving behind a lasting impression. The scrambled watermarked image is then descrambled to obtain the watermarked image. According to experimental findings, this technique is resistant to different attacks like compression, cropping, and noise; therefore, it is robust. A novel blind watermarking approach based on the game of life CA has been propounded by Adwan et al. in [9]. The Game-of-Life is the most well-known and robust rule of a CA. The proposed approach is implemented in the spatial domain, and it employs the LSB substitution technique for inserting the two most significant bits of the grayscale watermark into the host image. The suggested approach generates the k number of game-of-life generations and uses the locations of the live cells to insert data into the LSBs of the host image. According to the imperceptibility evaluation of the suggested scheme, the embedded watermark in the watermarked image is not perceptibly visible. The experimental findings demonstrate that the suggested approach is secure, simple, and robust enough against passive attacks. However, the method primarily addresses passive attacks and may not be as effective against more aggressive or complex manipulations.

A highly secure image watermarking approach based on the logistic map, the 2D Arnold's cat map, and the 2D Game of Life cellular automata has been suggested by Moniruzzaman et al. [10]. In the proposed scheme, the security mechanism is ensured by scrambling both the grayscale host image and the binary watermark image. The host image is scrambled using the 2D Arnold's cat map before the embedment of the watermark data into it. The initial configuration to start the Game of Life CA is set using the logistic map, and then the Game of Life CA is evolved to scramble the binary watermark image. At last, the scrambled watermark bits are inserted into the LSBs of the scrambled host image pixels. The suggested solution is used for image authentication and can avert unauthorized changes. Comparing the experimental findings of the proposed method to three other current chaos-based watermarking schemes depicts that the suggested approach significantly outperforms other approaches. The solitary constraint of this approach lies in its applicability exclusively to square images, as the Arnold transform's functionality is confined to image dimensions of uniform size, limiting its usability in practical scenarios where images may

not always be of uniform size. Yang et al. [25] introduced a novel approach that combines a deep blind model and a watermarking strategy, a robust deep blind watermarking encoder called CFC+CONCAT, which maintains imperceptibility while encoding watermarks. They put forward a clever approach to watermarking that boosts resilience by expanding the watermarking capacity with the help of patch-based image splitting and harnessing the power of different types of distortion. The results of the experiments revealed that this innovative technique could achieve an average PSNR that is 5.46 units higher than that of traditional methods, all the while maintaining a similar level of resilience against a range of distortions. The intricate nature of this approach and its demand for computational resources may limit its practicality and adoption in environments with constrained computational capabilities or in real-time applications. Additionally, the sophisticated nature of the technique could pose challenges for widespread use and integration into existing systems.

In another primary study, Tjokorda et al. put forth a remarkable approach to medical image watermarking, one that is truly reversible. This groundbreaking method, as presented in their seminal work [11], employs the clever technique of LSB modification to both detect and restore any tampering that may occur within the region of interest (ROI) of medical images. The experiments' results indicate good performance and imply that the suggested watermarking approach more cleverly handles the attacks that target a particular area of the pictures, i.e., block tampering attacks. This reversible medical image watermarking approach may have limited effectiveness beyond block tampering attacks, potentially necessitating careful consideration of its applicability to diverse tampering scenarios and computational requirements. A secret image data hiding technique by Manjula and Danti [12] employs the (2-3-3) LSB approach to embed confidential data into the host image. The proposed method embeds secret image data having eight bits per pixel into the LSBs of the RGB host image. The (2-3-3) LSB approach embeds the first 2 bits in the red channel, the next 3 bits in the green channel, and the last 3 bits in the blue channel of the RGB host image. This approach offers promising results, considerably improving PSNR and MSE values compared to the prior technique.

ROI lossless watermarking affects diagnosis accuracy, reversible watermarking lacks continuous verification, and zero watermarking requires third-party storage. To address these issues, Dai et al. [26] proposed a hybrid reversible-zero watermarking (HRZW) approach. The scheme combines reversible and zero-watermarking components, generating ownership shares using nearest neighbor grayscale residual (NNGR) mapping. The ownership shares are then embedded reversibly using Slantlet Transform, Singular Value Decomposition, and Quantization Index Modulation (SLT-SVD-QIM). Experimental results show that the proposed HRZW scheme achieves high watermarking quality, distinguishability, and robustness.

An imperceptible digital image watermarking scheme for color images in the spatial domain has been propounded by Abraham and Paul [13]. The objective of the reference [13] is to create an innovative technique for embedding watermarks in color images that avoids significant degradation in picture quality and preserves the original perception of colors. Moreover, to ensure the utmost resilience against attacks and facilitate tamper detection and recovery capabilities, the watermark data is inserted in each image block. This methodology incorporates the utilization of M1 and M2 masks during the watermark embedding process. These masks distribute the watermark information to the nearby pixels in the chosen area. The embedding channel uses the M1 mask, and to account for the differences introduced in the embedding channel, the other color channels are adjusted using mask M2. After being tested on various images, the suggested algorithm assures excellent-quality watermarked images that can withstand many attacks.

Pal et al. [27] introduced a novel watermarking approach for tamper detection and image authentication using a unique CA attractor. In this approach, authors first generate an authentication code (AC) from the watermark image by applying the SHA-512 (secure hash algorithm). Then, the secret watermark bits and AC are embedded into the four sub-sampled interpolated image blocks that make up the cover image. The output of this approach is four sub-sampled watermarked images, using which extraction of the secret watermark bits and reconstruction of the cover image is performed. Furthermore, the suggested method holds the potential to accurately detect any kind of distortion that could be brought about by different steganographic attacks in the watermarked images. When compared to other popular schemes of a similar nature, more effective results were observed concerning both quality and capacity. While this novel watermarking approach demonstrates significant improvements in security, quality, and capacity, it has certain limitations. A key limitation is that the method produces four sub-sampled watermarked images instead of a single output image. This can complicate the extraction process and might not be ideal for applications that require a singular, cohesive, watermarked image. The necessity to handle multiple images can also increase storage requirements and processing overhead, potentially making the approach less efficient and practical for certain real-world applications where simplicity and speed are paramount. Additionally, the scheme's reliance on the accurate reconstruction of the cover image from the four sub-sampled watermarked images might be sensitive to high levels of noise or distortion, which could affect its robustness under severe conditions.

The research on digital image watermarking techniques within the spatial domain elucidates a rich spectrum of methodologies, each exhibiting distinct strengths and limitations. Basic techniques, such as LSB and ISB modifications, provide high capacity and speed yet remain vulnerable to conventional attacks and manipulations. Conversely, advanced approaches like histogram modification

and correlation-based methods enhance robustness and imperceptibility, albeit with increased complexity. Cutting-edge techniques leveraging deep learning models offer substantial advancements in security and resilience, requiring specialized expertise and significant computational resources. In the realm of medical imaging, reversible watermarking ensures tamper detection and restoration, though it may fall short in continuous verification capabilities. Hybrid methodologies that integrate reversible and zero-watermarking techniques achieve superior quality and robustness, despite the complexity of implementation. Ultimately, the selection of an appropriate technique necessitates a careful balance among robustness, capacity, imperceptibility, and practical implementation considerations.

B. TRANSFORM DOMAIN ALGORITHMS

In the transform domain, the watermark data is not inserted directly into the host image but rather into the frequency coefficients of the host image. A number of transformational approaches have been devised for frequency coefficient generation, such as Discrete Wavelet Transform (DWT) [28], Discrete Cosine Transform (DCT) [29] [4] [30] [31], Singular Value Decomposition (SVD) [4], Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT) [32], Cellular Automata Transform (CAT), Polar Harmonic Transform (PHT), and Hadamard. The transform domain watermarking techniques first transform the host image from the spatial domain to the frequency domain using any image transformation method in order to generate the coefficients and then alter these coefficients to embed the watermark data. Finally, the inverse transformation method is applied to obtain the watermarked image.

Laouamer and Tayan have proposed a robust semi-blind sensitive text image watermarking approach [30] that is DCT and linear interpolation-based. The prime objective of this methodology is to address the concerns associated with detecting tampering, proving authenticity, verifying integrity, and safeguarding digitally sensitive textual images. In this approach, the host image and the watermark image are initially transformed into the YUV color space, after which both images are partitioned into blocks measuring 8×8. Each 8×8 block of both images undergoes Discrete Cosine Transform (DCT), and subsequently, the quantized DCT coefficients are subjected to linear interpolation in order to yield the watermarked image. According to this study, a good trade-off between robustness and imperceptibility can be achieved if the watermark data is inserted into the medium-frequency (MF) components. The main contribution of this work is to extract the watermark perfectly from the attacked watermarked image. While achieving a balance between robustness and imperceptibility, the reliance on linear interpolation and medium-frequency components may limit its effectiveness against certain types of attacks, such as geometric distortions or sophisticated image manipulations. Roy and Pal [31] have employed DCT and repetition code to put forth a blind color watermarking technique for embedding multiple watermarks. The primary purpose of the proposed method is to enable copyright ownership protection and multiple owner authenticity validations. In this approach, the host image's blue and green components are first fragmented into non-overlapping blocks, and then the DCT is applied to each block. This approach embeds two binary watermark logos scrambled using Arnold's chaotic map before embedding. The watermark bits are embedded into the middle-frequency coefficients of blue and green components using repetition codes. The proposed approach shows high robustness, imperceptibility, and better PSNR value but exhibits high computational complexity. The technique's computational complexity could hinder its practical implementation in real-time applications or environments with limited computational resources.

Liu et al. [33] have propounded a DCT and fractal encoding-based digital image watermarking algorithm. The proposed algorithm combines the traditional DCT technique with the fractal encoding method. This approach encrypts the host image twice, first by encoding it with the fractal encoding and second by applying DCT on the encoded parameters. The experiments carried out show that the proposed approach is highly robust and has a better PSNR value. Although the combination of DCT and fractal encoding enhances robustness and PSNR value, the method's reliance on double encryption may introduce additional computational overhead and complexity. Singh and Bhatnagar have presented a robust watermarking technique [34]. The techniques employed by this blind watermarking approach are integer DCT, dynamic stochastic resonance (DSR), and non-linear chaotic maps. In this approach, integer-DCT is applied to the host image to convert it into an integer linear transform, and the resulting coefficients are divided into non-overlapping blocks. Then, the non-linear chaotic map is used to select the random blocks that form the circulant matrix. The embedding of the watermark bits into the circulant matrix is accomplished through the calculation of the singular values, a process that employs the DSR phenomena to facilitate the extraction of watermarks in a highly efficient manner. The verification step is included to deal with the false positive problem caused in SVD systems. The experiments carried out show that the proposed approach is robust and imperceptible against various attacks. The technique's utilization of DSR and non-linear chaotic maps may pose challenges in terms of computational efficiency and practical implementation.

Ernawan et al. have propounded a DCT psycho-visual threshold-based digital image watermarking algorithm [35]. First, the host image is divided into non-overlapping blocks for watermark embedding, and their modified entropy is computed. Then, DCT is applied to the blocks with the lowest entropy values to obtain the middle-frequency coefficients. Some of these middle-frequency coefficient pairs are modified with a psycho-visual threshold to embed the watermark bits. The watermark is scrambled using Arnold Scrambling before embedding to strengthen the security levels. For evaluation of the proposed algorithm, the watermarked image had undergone various attacks like a median filter, low-pass

filter, sharpening, JPEG and JPEG2000, image noise, and geometrical attacks such as image scaling and cropping. The results demonstrated that the proposed method is invisible and robust compared to the existing methods. However, the suggested technique's reliance on psycho-visual thresholds and entropy values may limit its effectiveness in scenarios where the image content varies significantly.

A novel color image watermarking scheme based on Discrete Cosine Transform (DCT) and Cellular Automata (CA) has been put forth by M Jana and B Jana [36]. This study focuses on developing an image watermarking system with enhanced security and high embedding capacity without compromising the visual quality of the host image. The RGB host image is first separated into red, green, and blue channels. After that, each channel is partitioned into non-overlapping 8x8 blocks; subsequently, DCT is applied on each block, and Zigzag scanning is performed. To increase the security and robustness of the proposed system, CA rule-15 is used to encrypt the watermark data prior to embedding, and CA rule-85 is used for decryption purposes. The CA rule-340 and mapping table are used to modify the DCT coefficients and incorporate the encrypted watermark image. The propounded method is equated with the existing methods. Furthermore, the simulation findings demonstrate an average Peak Signal Noise Ratio (PSNR) value of 54 dB, which signifies strong imperceptibility and a concomitant embedding capacity of 1.48 bpp.

The exploration of transform domain watermarking methodologies unveils a range of approaches, each characterized by unique advantages and constraints. Foundational methods like DCT and DWT offer efficiency but are vulnerable to targeted attacks, while advanced techniques may introduce complexities. Achieving a balance between robustness and imperceptibility remains a challenge, with some methods prioritizing one over the other. Reversible watermarking ensures tamper detection but may lack continuous verification. Hybrid approaches offer superior quality and resilience but are complex to implement. In summary, the selection of an optimal technique demands a nuanced balance of robustness, imperceptibility, capacity, and practical feasibility, underlining the pivotal role of these factors in shaping the future of transform domain watermarking.

III. METHODS AND MATERIALS

A. CELLULAR AUTOMATA

John von Neumann and Stanislaw Ulam are credited as inventors of cellular automata. Neumann proposed a theoretical model for artificial biological systems called the "Universal Constructor" that had the ability of self-reproduction and was comprised of a two-dimensional infinite grid of square cells; each cell had five neighbors, including itself, with 29 states per cell. With the development of Conway's Game of Life popularity of the subject began to spread outside of academics. The Game of Life, developed by mathematician John Conway and hyped up by an article in Scientific American by Martin Gardner, is a simple two-dimensional totalistic cellular automaton with two states per cell using the Moore neighborhood.

A Cellular automaton can be defined as a mathematical model having discrete space and time domains. This computational system is dynamic in nature, and simple rules govern its evolution. Formally, CA is defined by the quadruple A = ($\mathbb{Z}^D$, S, N, f) where:

- $\mathbb{Z}^D$ is a D-dimensional lattice, i.e., a regular arrangement of lattice-sites/points in Euclidean space which is discrete and is closed under subtraction and addition, called cellular space.
- **S** is a finite set of states.
- **N** is a neighborhood vector $(\overrightarrow{n_1}, \overrightarrow{n_2}, \overrightarrow{n_3} \ldots \overrightarrow{n_m})$, comprising of m different cells of $\mathbb{Z}^D$ and is identified separately for every cell of the lattice $\mathbb{Z}^D$. For a given cell "x" a set of cells $\{x + \overrightarrow{n_i} | i = 1, 2, 3, 4 \ldots .m\}$ forms its neighborhood and from the property of lattice $(x + \overrightarrow{n_i}) \in \mathbb{Z}^D$.
- $f : S^m \longrightarrow S$ is the local transition function known as the local rule in CA that synchronously updates the state of every cell based on the current state of the cells in its neighborhood.

All the cells in a cellular space are updated simultaneously using the same local update rule. A function/ mapping that assigns cells their states/values from the set of states (S) is called the configuration of the D-dimensional CA ($\mathbb{Z}^D$) and all configurations are contained in the set $S^{\mathbb{Z}^D}$. Besides, the evolution of CA generations can be described as the hopping of the cellular space from one configuration to another according to its global transition function $G : S^{\mathbb{Z}^D} \longrightarrow S^{\mathbb{Z}^D}$. One-dimensional cellular automata (1D CA) and two-dimensional cellular automata (2D CA) are the two most prevalent forms of CA. A finite or infinite set of identical cells/sites containing discrete variables arranged in the form of a linear array is referred to as 1D CA [37]. Figure 1 depicts the 1D CA with a five-cell neighborhood for updating the state of the cell "n" where the local rule depends on the current states of the two nearest left $(n-2, n-1)$ and right $(n+1, n+2)$ neighbors of the cell "n" and the current state of cell "n" itself. The transition function according to which this CA evolves is defined using Equation (1).

$$S_n(t+1) = f(S_{n-2}(t), S_{n-1}(t), S_n(t), S_{n+1}(t), S_{n+2}(t)) \quad (1)$$

where $S_{\text{n-2}}(t)$, $S_{\text{n-1}}(t)$, $S_{\text{n+1}}(t)$, and $S_{\text{n+2}}(t)$ are the current states of the neighbors at time t, $S_{\text{n}}(t)$ is the current state of the cell "n" at time t, $S_{\text{n}}(t+1)$ is the new state of the cell "n" at time $t+1$, and f is the local update rule.

A 2D CA can be defined as a finite or infinite set of identical cells/sites containing discrete variables arranged in the form of a matrix/grid. For 2D CA, a number of neighborhood structures are possible. Still, the von Neumann neighborhood and the Moore neighborhood are the two most prevalent neighborhood structures illustrated in Figure 2 (a) and 2(b), respectively. The von Neumann/diamond-shaped/five-cell neighborhood is defined as the set of cells that includes

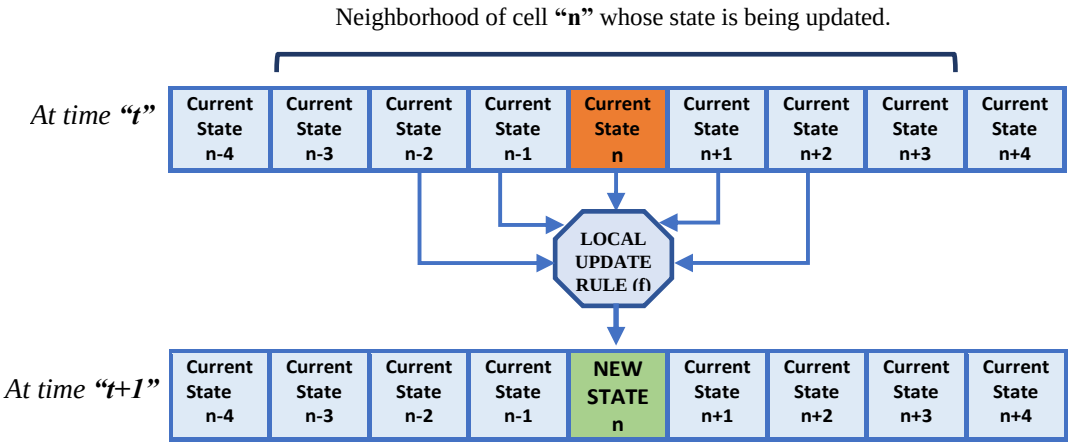


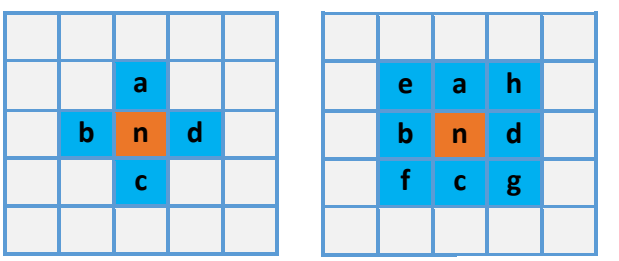
FIGURE 1: 1D Cellular Automata with five-cell Neighborhood.

the central cell and its four adjoining perpendicular cells. Equation (2) illustrates the transition function of the five-cell neighborhood.

$$S_n(t+1) = f(S_n(t), S_a(t), S_b(t), S_c(t), S_d(t)) \quad (2)$$

The Moore/square-shaped/nine-cell neighborhood is defined as the set of cells containing the central cell, its four adjoining perpendicular cells, and four adjoining diagonal cells. Equation (3) illustrates the transition function of the nine-cell neighborhood.

$$S_n(t+1) = f(S_n(t), S_a(t), S_b(t), S_c(t), ..., S_h(t)) \quad (3)$$



(a) The Von Neumann Neighborhood ***(b) The Moore Neighborhood***

FIGURE 2: 2D CA Neighborhood Structures.

1) Elementary Cellular Automata (ECA)

An elementary cellular automaton is the most basic form of 1D CA, extensively studied by Stephen Wolfram since the 1980s. Given the defining characteristics of a CA, the potential scenario in its most basic form is the linear array of cells with two states (0 or 1) per cell and the neighborhood set S = (the cell, it's the immediate right neighbor, it's immediate left neighbor). This basic scenario is termed elementary cellular automata, illustrated using the transition function in Equation (4).

$$S_n(t+1) = f(S_{n-1}(t), S_n(t), S_{n+1}(t)) \quad (4)$$

Since there are two states and three neighbors for each cell, this leads to 2^3 = 8 possible ways to configure the neighborhood, and therefore there are only 2^{2^3} = 256 total ECA rule-sets possible. The Wolfram code, developed by Stephen Wolfram, presented a method for assigning numbers between 0 and 255 to each rule, and this method has now become a norm. Interestingly, Wolfram has categorized these 256 rulesets into four possible classifications based on the increasing complexities of their behaviors [38].

- Class (I) Uniformity: The evolution of almost all the initial configurations rapidly leads to stable, uniform structures, thus completely losing randomness, if any.
- Class (II) Oscillation: The evolution of almost every initial configuration results in patterns that are either stable after a large number of generations or tend to repeat themselves. The initial configuration may lose some of its randomness, but some remains.
- Class (III) Random: The evolution of most of the initial configuration results in chaotic or completely pseudo-random sequences.
- Class (IV) Complexity: Almost all the initial configurations evolve into complex structures with intriguing ways of interaction.

Rule 30: is an elementary cellular automata rule that belongs to a class (III) of Wolfram's classification because of its chaotic and erratic behavior. The ruleset for ECA Rule 30 is depicted in Figure 3, and Figure 4 illustrates the first fifty evolutions of the ECA, which is updated using Rule 30 with the initial configuration being a single middle black cell. Rule 30 generates random configurations from simple initial states, which makes it remarkable, and that is why Mathematica has also employed it as a random number generator [39]. It is left permutative, thus highly sensitive to the initial states, as a tiny difference in one state had a

t	111	110	101	100	011	010	001	000
t+1	0	0	0	1	1	1	1	0

FIGURE 3: The Ruleset for Elementary CA Rule 30.

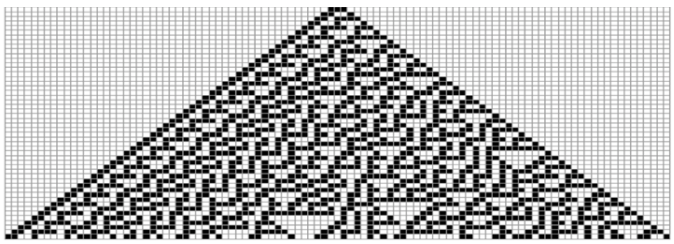


FIGURE 4: First Fifty Evolutions of ECA Evolved Using Rule 30 from [43].

significant impact on subsequent results [40]. The whole set of NIST statistical tests was applied to the output of Rule 30 to analyze and evaluate its randomness, and it was found that except for one, all the tests were passed [41]. It was also shown that for better performance, it is important to have an effective window size, and a size of 200 bits is believed to be sufficient for better randomness [42]. The main reason to use the ECA Rule 30 for embedding and encryption in our system is for its special characteristics, such as its chaotic behavior and sensitivity to initial conditions, which make it ideal for producing high-entropy pseudo-random sequences.

The computational irreducibility of Rule 30 is one of its remarkable characteristics. This implies that computational prediction of its behavior over time or determining a brief representation of its evolution is equal to computing each step separately. It is difficult to minimize or predict the pattern's intricacy. Rule 30's computational universality is well-known. In the larger context of complexity theory and the study of cellular automata, Rule 30 is of great interest because it provides a classic illustration of how simple local rules can give rise to complex and seemingly unpredictable patterns.

B. PROPOSED IMAGE WATERMARKING SCHEME.

The proposed digital image watermarking method is a spatial domain approach in which watermark bits are embedded into the LSBs of the host image. Spatial domain methodologies strike a delicate balance between imperceptibility, robustness, and capacity through refined adjustments in embedding strength, strategic selection of embedding methodologies, integration of redundancy and error correction mechanisms, implementation of adaptive embedding approaches, utilization of multi-resolution techniques, exploitation of perceptual models, and the deployment of encryption protocols. These techniques are simpler, more effective, and execute more quickly.

In this approach, inputs include a grayscale host image of size ($Hm \times Hn$), a grayscale watermark image of size ($Wm \times Wn$), and secret keys (embedding-key and scrambling-key) generated using CA. The proposed scheme is based on powerful ECA Rule 30, which is an ideal source of randomness. Confusion can result from the fact that we are viewing a 2D representation of the 1D CA. It should be noted that the proposed scheme constructs the two-dimensional representation out of a large number of evolutions of one-dimensional output data. Nevertheless, the CA system is purely one-dimensional.

This scheme is divided into five phases: the Secret Keys Generation Phase, the Watermark Pre-processing Phase, the Host Image Scrambling Phase, the Watermark Embedding Phase, and the Watermark Extraction Phase.

1) The Secret Keys Generation Phase.

This is a prime part of the proposed watermarking scheme, as the three distinct secret keys are generated in this phase using ECA. The first and second are the embedding key and the encryption key, which are sized to match the watermark image ($Wm \times Wn$), and the third is the scrambling key, sized to match the host image ($Hm \times Hn$), as shown in Figure 5 (a).

For the embedding-key generation, this scheme employs the ECA as the linear array with a window size of 201 bits/cells contained within the periodic boundaries. Our CA starts off with all the white cells (i.e., all bits equal to zero) except the middle cell at index 101 (array index starts at 1), which is black (i.e., one). The ECA with the above-mentioned parameters is evolved using local update 'Rule 30 ' for the "n" number of evolutions, where "n" depends on the size of the watermark image and the number of generations. From each of these n evolutions, we collect the middle bits as shown in Figure 6. This bit sequence is taken as the output, which forms the embedding-key matrix after layout modification from 1D to 2D. To increase the scheme's chaoticity while embedding, k such matrices, termed generations of the embedding key, are configured.

This method generates a thoroughly random two-bit encryption key by a strategic combination process. Specifically, it involves taking the bits at the final generation (k^{th}) and the penultimate generation ($k - 1^{th}$) within the embedding key. These bits, which reside at identical indexes in their respective generations, are combined to form the encryption key. This approach ensures a high level of randomness and security, as the key is derived from the unique and variable bits at these critical positions in the embedding key sequence. The construction of the encryption key is numerically depicted in Figure 5 (b).

To initiate the generation of the scrambling-key, the original parameters utilized for initializing the ECA remain consistent with those employed for the embedding-key generation. However, a pivotal modification is made to the window size of the linear array, aligning it with the number of columns in the host image. This reconfigured ECA ar-

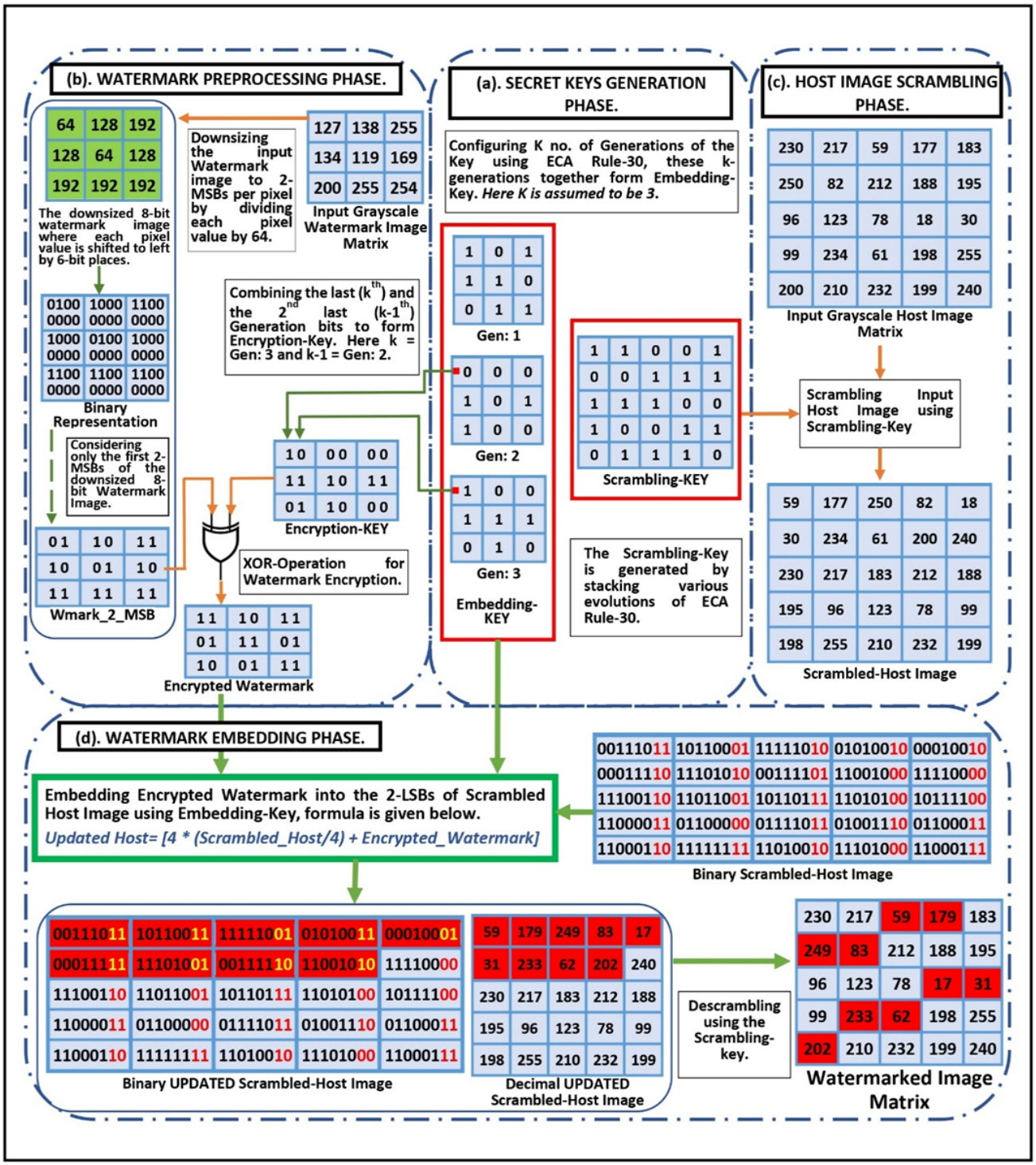


FIGURE 5: Numerical Illustration of the First Four Phases of the Watermarking Process.

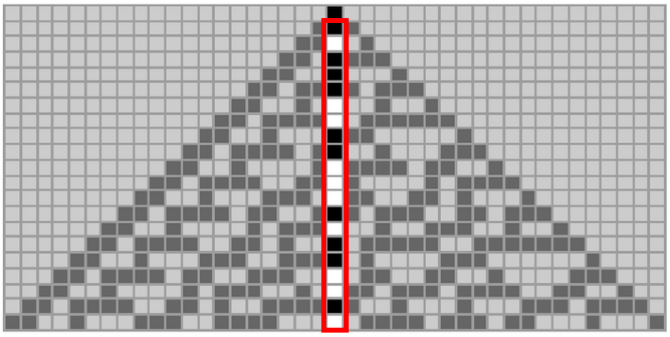


FIGURE 6: The Middle Bit Sequence from [44].

rangement also evolves using Rule 30. Once the formation of the triangular structure is complete, we start stacking the output of every evolution in the downward direction for the number of iterations to construct the scrambling key, where the 'number of iterations'/ 'height of the stack' is equal to the number of rows in the host image.

Randomness and length are two important criteria that affect the security of the keys generated during the secret key generation phase. A randomly generated series of bits highly resistant to prediction is known as a secure key. Its strength lies in its unpredictability and length. Longer keys exponentially increase the number of possible combinations, making brute-force attacks impractical. Attackers engage in these types of attacks by methodically trying every key until they discover the right one. However, when the keys are lengthy enough, the time and resources needed become unaffordable. Thus, to ensure the security of the data, safe keys should be both randomly generated and long enough to resist brute-force attacks. The ECA Rule 30 is employed to generate all the keys used in our image watermarking approach. This rule provides the basis for creating secure and unpredictable keys for encryption, scrambling, and embedding. It is well-known for its intricacy and pseudo-random behavior. ECA Rule 30 operates on a simple set of rules, yet produces highly intricate and seemingly random patterns, making it an ideal candidate for cryptographic applications. Extensive research has shown that Rule 30 exhibits properties such as chaos, ergodicity, and sensitivity to initial conditions, which are desirable traits for key generation. Furthermore, as the longer keys offer a heightened level of security, the iterative process of generating k generations of the key further enhances the robustness and resistance to cryptographic attacks due to increasing key length.

It's critical to note that the aforementioned CA arrangement is not a 2D CA at all but rather the output of 1D CA organized in the form of a matrix.

2) The Watermark Pre-processing Phase.

Before watermark information is inserted into the host image, a preliminary watermark preprocessing step is imperative. The motive of this preprocessing stage is twofold: to downsize the grayscale watermark image to its two Most Significant Bits (MSBs), and level up the suggested system's security attribute by encrypting the watermark information before it is embedded. The watermark preprocessing phase is further divided into two distinct processes: Downsizing the Watermark and Encrypting the Watermark as depicted in Figure 5 (b).

- *Downsizing The Watermark:*
  To address the imperceptibility requirements of the digital image watermarking system, the suggested method does not embed the entire grayscale watermark image but rather embeds the two most significant bits of each pixel of the grayscale watermark image. To make that happen, a grayscale watermark image having ($Wm \times Wn$) number of pixels with 8 bits per pixel is downsized to 2-MSBs per pixel by simply dividing each pixel value by 64, as shown in Equation (5).

$$Wmark\_2\_MSB(i,j) = \frac{Wmark(i,j)}{64} \tag{5}$$

  where $Wmark(i,j)$ is the watermark 8-bit-pixel value at index (i,j) and $Wmark\_2\_MSB(i,j)$ is the watermark pixel value reduced to 2-MSBs, as division by 64 repeatedly shifts the pixel bits to the right by 6-bit places and the six rightmost bits get discarded. Also, six places on the left are set to zero.
- *Encrypting Watermark:*
  After downsizing the watermark image to 2-MSBs, it undergoes an encryption process to add another level of security to the proposed algorithm. To accomplish the encryption task, the bitwise XOR operation is applied to the watermark data to conceal it. The perfectly balanced nature, straightforward implementation, and low computation costs are the significant properties of the XOR operator, making it a better option for encryption purposes. This approach forms an entirely random two-bit encryption key by simply combining every last (k^{th}) and the second last ($k\text{-}1^{th}$) generation bits at similar indexes of the embedding key. The bitwise XOR operation is then applied to the 2-bits of the watermark data at index (i,j) and the 2-bits of the encryption key at index (i,j), as shown in Equation (6). This operation is performed on every element of the matrix.

$$Encrypted\_Watermark(i,j) = Wmark\_2\_MSB(i,j) \oplus Encryption\_Key(i,j) \tag{6}$$

3) The Host Image Scrambling Phase.

In the proposed approach, the encrypted watermark is not embedded directly into the host image pixels. Instead, the host image first gets scrambled using the scrambling key, as shown in Figure 5 (c). The rationale behind scrambling the host image is to distribute the watermark pixels throughout the host image, thereby achieving high robustness against geometrical attacks like cropping. In the proposed approach,

the process of embedding the encrypted watermark data is performed on the LSBs of the scrambled host image. After embedding, the host image is descrambled, resulting in the distribution of the encrypted watermark pixels throughout the host image. This eliminates the correlation between the pixels embedded with watermark data, thereby enhancing resistance against potential geometrical attacks. The host image scrambling process is numerically depicted in Figure 5 (c) and works according to the principle given below.

```
STEP 1:
for  i=1 to RowSize do
    for j=1 to colSize do
        if scrambling_key(i,j) == 0 then
            scrambled_host(row, col) = Host_Image (i,j);
        end if
    end for
end for
STEP 2:
for  i=1 to RowSize do
    for j=1 to colSize do
        if scrambling_key(i,j) == 1 then
            scrambled_host(row, col) = Host_Image (i,j);
        end if
    end for
end for
```

Row and col are initialized to 1 and accordingly incremented after every element is assigned with the pixel value in the scrambled_host matrix.

4) The Watermark Embedding Phase.

The insertion of the preprocessed watermark data into the scrambled host image using the above-generated Rule 30-based embedding key is explained in this section. The numerical illustration of the watermark embedding phase is shown in Figure 5 (d). The encrypted watermark pixel values to be embedded are selected randomly based on the locations of the live cells (1s) in all the generations of the embedding-key starting from the first to the K^{th} generation. After covering all the live cells if there are still watermark pixel values left to be embedded, then these values are also selected randomly based on the locations of the dead cells (0s) in the embedding-key; starting again from the first to K^{th} generation.

In the proposed scheme, the selected two-bit watermark data is inserted into the two LSBs of the host image pixel values using the addition operation. Therefore, it becomes obligatory to clear the previous values of the two LSBs of the selected host image pixels and set them to zero, which is achieved by first dividing and then multiplying each of them by four. The scrambled host image pixel values into which watermark data is to be embedded are selected in a sequential manner, starting from the first pixel to the 'n^{th}' pixel, where n = (Wm $\times$ Wn), i.e., the length of the watermark image. The pixels forming the selected sequence are not correlated at all, as the host image is already scrambled and thus robust against the potential geometrical attacks. Subsequent to the insertion of all the two-bit watermark pixel values, the updated scrambled host image is descrambled using the scrambling key, which finally produces the watermarked image. The complete embedding process is described using pseudo-code notation in the next section.

5) The Pseudo-code Description of the Watermark Embedding Process.

The Algorithm 1 presents the pseudo-code of the watermark embedding process of the proposed CA-based digital image watermarking scheme.

6) The Watermark Extraction Phase.

To retrieve the watermark image that was embedded using a specific technique, the inverse procedure of that technique is carried out, called the watermark extraction process. In the proposed scheme, the extraction process begins by scrambling the watermarked image using the same scrambling key that was used during embedding. Then, from the scrambled watermarked image, the sets of 2-LSBs of the first n pixels are extracted, where n = (Wm$\times$Wn). These extracted n-sets of 2-LSBs are arranged in a (Wm $\times$ Wn) matrix based on the locations of all the live cells (1s) first, followed by the locations of the dead cells (0s) in all the generations of the embedding key.

Nevertheless, the watermark data that has been obtained remains obscured by encryption, thus necessitating a decryption procedure. The extracted watermark data is subjected to the elaborate Bitwise XOR operation with the encryption key. Following decryption, each 2-bit pixel is shifted to the left by 6-bit places in order to get converted into the 8-bit pixel. As a result, extracted LSBs become the MSBs of the extracted watermark image after six left bit-shifts. The numerical illustration of the complete watermark extraction phase is shown in Figure 7.

IV. EXPLORING THE CHALLENGES OF BALANCING ROBUSTNESS, IMPERCEPTIBILITY, AND CAPACITY.

Achieving an ideal balance among robustness, imperceptibility, and capacity is a central challenge in various fields, particularly in areas like digital watermarking, steganography, and information hiding. Let's delve into the discussion on the challenges and limitations associated with this balance:

- Achieving high robustness is crucial for ensuring that the embedded data remains intact even after processing or transmission. However, enhancing robustness often comes at the expense of imperceptibility and capacity. The challenge lies in finding techniques that strike the right balance between robustness and other factors.
- Preserving imperceptibility is crucial to avoid the discovery of concealed information by unintended parties or adversaries. However, enhancing imperceptibility may restrict the quantity of embeddable data or compromise robustness against specific attacks.

Algorithm 1 Watermark Embedding Algorithm

Input: Host (Hm×Hn), Scrambling_key (Hm×Hn), Embedding_key (Wm×Wn), and Watermark(Wm×Wn).
Output: Watermarked_Image.

procedure EMBEDDING($HostImage, Scrambling\_key, Embedding\_key, Watermark$)
  $Wmark\_2\_MSB \leftarrow Downsizing(Watermark)$;
  $Encrypted\_Watermark \leftarrow bitxor(Wmark\_2\_MSB, Encryption\_key)$;
  $Scrambled\_Host \leftarrow Scrambling(Host, Scrambling\_key)$;
    ▷ The (Wm×Wn) Embedding_key consists of K (Wm×Wn) generations (Gen1, Gen2, Gen3 ..... GenK).
  $row \leftarrow 1; col \leftarrow 1$;
    ▷ Covering the locations of the live cells (1s) in Gen1.
  for $i \leftarrow 1$ to Wm **do**
    for $j \leftarrow 1$ to Wn **do**
      if (Gen1[i][j] = 1) **then**
        $temp := (floor(Scrambled\_Host[row][col]/4) \times 4)$;
    ▷ The two MSBs of the encrypted watermark at index (i, j) are added to the two LSBs of the scrambled host image pixel at index (row, col).
        $Updated\_Scrambled\_Host[row][col] := temp + Encrypted\_Watermark[i][j]$;
        $col := col + 1$;
      end if
    end for
  end for
    ▷ Covering the locations of the live cells in GenT, where T = 2, 3, 4...K and S = 1, 2, 3... T-1.
  for $i \leftarrow 1$ to Wm **do**
    for $j \leftarrow 1$ to Wn **do**
      if $(GenT[i][j] = 1) and (GenS[i][j] \neq 1)$ **then**
        $temp := (floor(Scrambled\_Host[row][col]/4) \times 4)$;
        $Updated\_Scrambled\_Host[row][col] := temp + Encrypted\_Watermark[i][j]$;
        $col := col + 1$;
      end if
    end for
  end for
    ▷ After covering all live cells, the pixel values at dead cells (0s) are selected
  for $i \leftarrow 1$ to Wm **do**
    for $j \leftarrow 1$ to Wn **do**
      if $(GenT[i][j] = 0) and (GenS[i][j] \neq 1)$ **then**
        ▷ $Where T = 1, 2, 3...K; S = 1, 2, 3... K and S \neq T.$
        $temp := (floor(Scrambled\_Host[row][col]/4) \times 4)$;
        $Updated\_Scrambled\_Host[row][col] := temp + Encrypted\_Watermark[i][j]$;
        $col := col + 1$;
      end if
    end for
  end for
    ▷ Till all the watermark image bits are embedded.
  Watermarked_Image := Descrambling (Updated_Scrambled_Host, Scrambling_key)
  return $Watermarked\_Image$;
end procedure

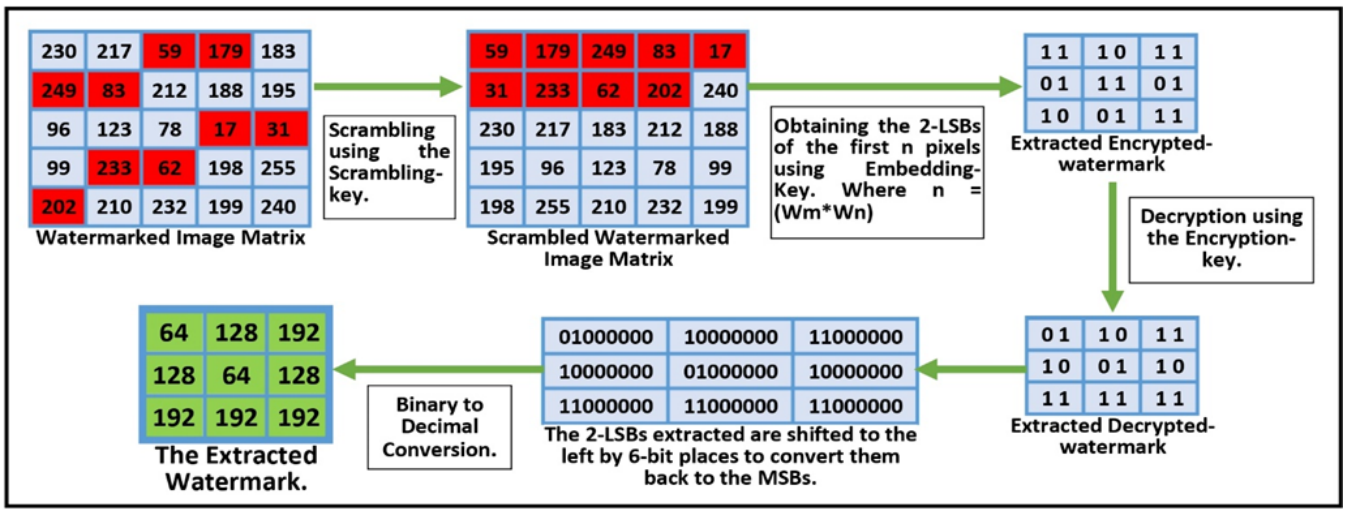


FIGURE 7: Numerical Illustration of the Watermark Extraction Phase.

- Expanding capacity enables the embedding of larger amounts of data, facilitating applications like data authentication, copyright protection, and covert communication. Nonetheless, increasing capacity typically involves a trade-off with imperceptibility and robustness. High-capacity methods can introduce detectable distortions or vulnerabilities to attacks.
- A significant challenge involves managing the trade-offs among these conflicting factors. Achieving the ideal balance demands meticulous attention to the particular application needs, security considerations, and the attributes of the underlying data and cover medium.
- Striking the right balance also entails assessing the trade-offs between security and usability and adjusting embedding techniques accordingly. This may involve fine-tuning parameters or choosing alternative methods tailored to the application's requirements. Ultimately, the aim is to achieve an ideal compromise that ensures adequate security without sacrificing usability or data fidelity.
- Technological advancements continually introduce new challenges and opportunities in achieving the balance between robustness, imperceptibility, and capacity. Keeping pace with these advancements requires ongoing research and innovation to develop robust and efficient techniques that address the evolving landscape of threats and constraints.

In summary, achieving an ideal balance among robustness, imperceptibility, and capacity is a complex and multifaceted challenge that requires careful consideration of trade-offs, application requirements, and technological constraints. Despite the inherent limitations, ongoing research and advancements offer promising avenues for improving the effectiveness and efficiency of watermarking techniques in various domains.

V. PERFORMANCE ANALYSIS AND EXPERIMENTAL DISCUSSION.

In this section, the effectiveness of the proposed algorithm is analyzed. The performance of the suggested scheme is evaluated by computing the various robustness and imperceptibility quality metrics and comparing the results with several existing algorithms [24] [9] [10].

A. EXPERIMENTAL SETUP.

The proposed watermarking scheme is implemented on the computer system with the following configuration: Intel Core i5-2410M CPU @ 2.30 GHz, 4.00 GB RAM, and Windows 7 Ultimate 64-bit SP1 operating system. It is designed, programmed, and analyzed using the MATLAB R2020a [52] platform. The grayscale logo image, which is 128×64 in size, is considered the watermark while testing the proposed approach. The performance testing of the suggested scheme is performed on the dataset of 42 test host images shown in Figure 8. These test images are collected from various standard image databases, which are USC-SIPI [45], The Cell Image Library [46], [47], The Cancer Imaging Archive (TCIA) [48], STARE [49], National Archives Catalog [50], NASA [51], and MATLAB Toolbox [52]. A comprehensive description and the dimensions of these test images in our dataset, meticulously sourced from diverse databases, are presented in Table 1. Some of the images in the dataset were originally color images and, therefore, required conversion to grayscale with a bit depth of eight, and that was achieved using MATLAB built-in image type conversion functions. The proposed algorithm was tested on the test host images of different dimensions and formats like png, tif, tiff, jpg, etc. The original size of the test images was not altered at all except for those from the NASA library, in which the size of huge images was reduced. To determine the efficiency of the suggested watermark system, its test results were compared with several existing systems.

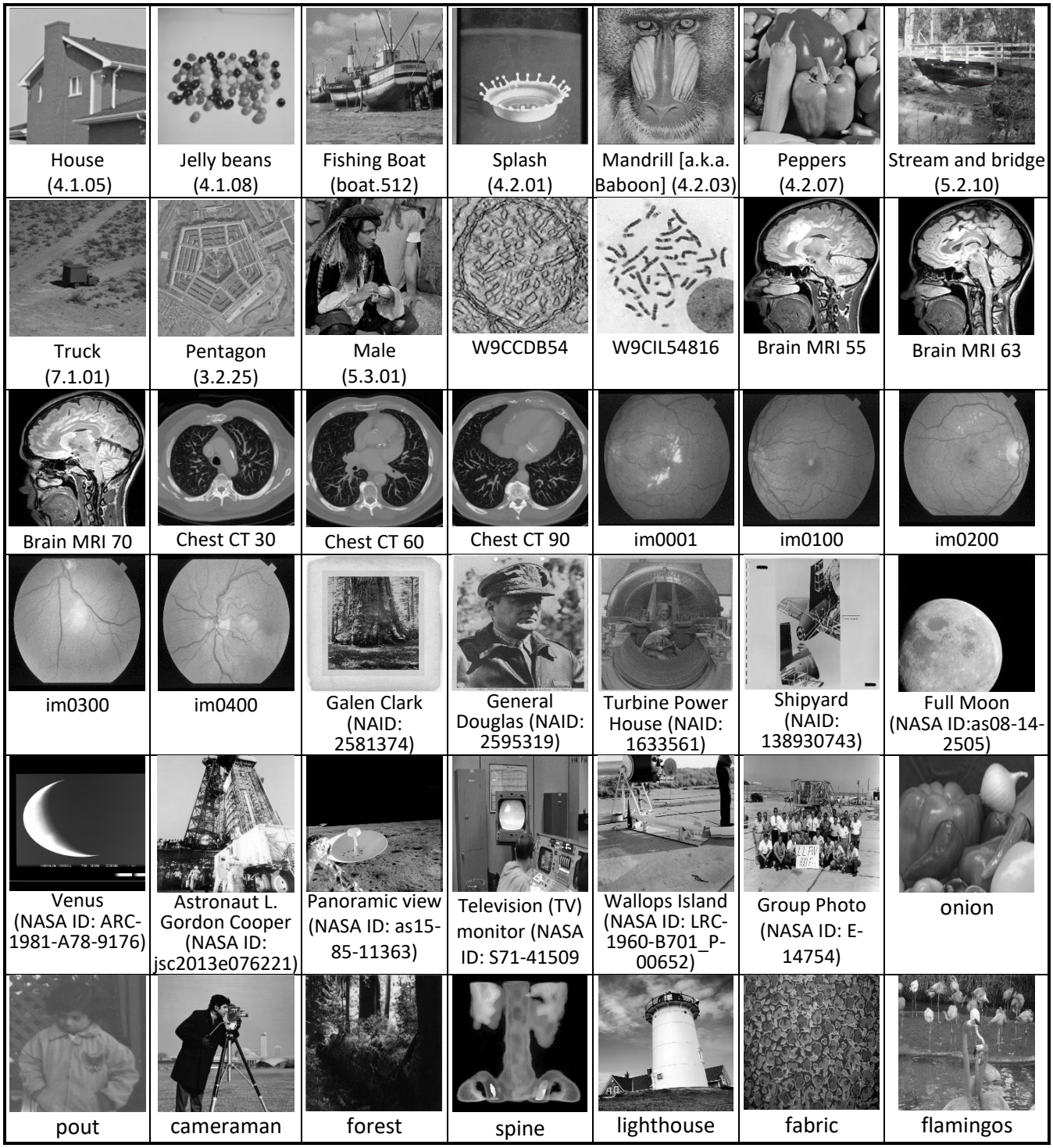


FIGURE 8: The dataset of 42 test host images was used for performance evaluation.

TABLE 1: Description and dimensions of the various test images in our dataset, sourced from diverse databases.

DATABASE	Grayscale Test Image Name and ID.	Test Image Size
USC-SIPI [45]	House (4.1.05)	256x256
	Jelly beans (4.1.08)	256x256
	Fishing Boat (boat.512)	512x512
	Splash (4.2.01)	512x512
	Mandrill [a.k.a. Baboon] (4.2.03)	512x512
	Peppers (4.2.07)	512x512
	Stream and bridge (5.2.10)	512x512
	Truck (7.1.01)	512x512
	Pentagon (3.2.25)	1024x1024
	Male (5.3.01)	1024x1024
Cell Image Library [46], [47]	W9CCDB54	512x512
	W9CIL54816	524x581
Cancer Imaging Archive [48]	Brain MRI 55	256x256
	Brain MRI 63	256x256
	Brain MRI 70	256x256
	Chest CT 30	256x256
	Chest CT 60	256x256
	Chest CT 90	256x256
STARE [49]	im0001	700x605
	im0100	700x605
	im0200	700x605
	im0300	700x605
	im0400	700x605
National Archives Catalog [50]	Galen Clark (NAID: 2581374)	576x706
	General Douglas (NAID: 2595319)	576x712
	Turbine Power House (NAID: 1633561)	576x718
	Shipyard (NAID: 138930743)	1024x1024
NASA [51]	Full Moon (NASA ID: as08-14-2505)	256x256
	Venus (NASA ID: ARC-1981-A78-9176)	256x256
	Astronaut L. Gordon Cooper (NASA ID: jsc2013e076221)	512x512
	Panoramic view (NASA ID: as15-85-11363)	512x512
	Television (TV) monitor (NASA ID: S71-41509)	512x512
	Wallops Island (NASA ID: LRC-1960-B701_P-00652)	512x512
	Group Photo (NASA ID: E-14754)	1024x1024
MATLAB Toolbox [52]	onion	198x135
	pout	240x291
	cameraman	256x256
	forest	447x301
	spine	490x367
	lighthouse	480x640
	fabric	640x480
	flamingos	1296x972

B. PERFORMANCE EVALUATION METRICS.

The most significant attributes, imperceptibility, robustness, and payload of the digital image watermarking system are computed by various image quality metrics, which are given below. In contrast, when calculating the evaluation metrics, the original host image is used as a reference image to determine the quality of the watermarked images.

1) Imperceptibility.

The concept of imperceptibility revolves around the idea that once we insert watermark data into the main image, the resulting quality should be indistinguishable from the original image. To assess the imperceptibility aspect of the suggested watermarking system, we rely on five distinct performance evaluation metrics mentioned below.

- Mean-Squared Error (MSE): The averaged intensity between the original host and the watermarked image is calculated using Mean Square Error. It gauges the degree to which a pixel varies from its original state. The smaller MSE value signifies that the watermarked image resembles the original host image. Equation (7) is used to determine MSE.

$$MSE = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}(HI_{(m,n)} - WI_{(m,n)})^2 \quad (7)$$

Where $HI_{(m,n)}$ and $WI_{(m,n)}$ denote the pixel values at index (m,n) in the original host image and the watermarked image, respectively, and MxN is the size of the images.

- Root Mean-Squared Error (RMSE): It is a quality assessment metric that is used for the error magnitude evaluation. It is derived by simply square rooting the MSE as illustrated in Equation (8).

$$RMSE = \sqrt{MSE} \quad (8)$$

- Peak Signal-to-Noise Ratio (PSNR): The well-known image quality metric widely used to evaluate the perceptual quality of the watermarked images with reference to the original host images is PSNR. It is derived from

the MSE and is expressed as the ratio of the maximum pixel intensity to the power of the distortion. The PSNR value should be at least greater than 35 dB; the higher PSNR value denotes better imperceptibility. Equation (9) is used to determine PSNR.

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \tag{9}$$

- Structural Similarity (SSIM) Index: The perceptual quality assessment metric that is used to measure the similarity between the original host image and the watermarked image is SSIM. The watermarked image has great perceptual quality if the SSIM value is close to 1. The SSIM of the watermarked image with reference to the original host image is determined using Equation (10).

$$SSIM(HI, WI) = \frac{(2\mu_{HI}\mu_{WI} + C1)(2\sigma_{HIWI} + C2)}{(\mu_{HI}^2 + \mu_{WI}^2 + C1)(\sigma_{HI}^2 + \sigma_{WI}^2 + C2)} \tag{10}$$

Where μ_{HI} and μ_{WI} are the averages, σ_{HI}^2 and σ_{WI}^2 are the variances, and σ_{HIWI} is the covariance of the original host image and watermarked image, respectively. C1 = $(k1L)^2$ and C2 = $(k2L)^2$; L = $(2^{Bits/Pixel} - 1)$, k1 = 0.01 and k2 = 0.03.

- Universal Quality Index (Q-Index): The distortion within an image is determined by the Q-Index. The range of the Q-Index is [-1 to 1], and its best possible value can be 1, indicating that the images are identical. The three parameters required to calculate the Q-Index are correlation, luminance, and contrast, which are calculated using Equation (11).

$$Q = \frac{\sigma_{HIWI}}{\sigma_{HI}.\sigma_{WI}} \cdot \frac{2\overline{HI}.\overline{WI}}{(\overline{HI})^2 + (\overline{WI})^2} \cdot \frac{(2\sigma_{HI}.\sigma_{WI}}{(\sigma_{HI}^2.\sigma_{WI}^2)} \tag{11}$$

2) Robustness.

The robustness of the image watermarking system measures the ability of the embedded watermark to resist and remain unaffected by various intentional and unintentional attacks, thus implying the system's reliability. To evaluate the robustness attribute of the proposed watermarking system, the original watermark is compared to the extracted watermark using the performance evaluation metrics below.

- Correlation Coefficient (CC): When evaluating a watermarking scheme's robustness to various attacks and transformations, the correlation coefficient is an essential statistical measure to consider. It gives information about how well the system will maintain the watermark's integrity and tolerate changes while retaining the ability to allow accurate watermark extraction. The correlation coefficient quantifies the intensity and direction of the linear relationship between the original watermark and the extracted watermark. Equation (12) can be used to determine the correlation coefficient value, which ranges from 0 to 1. where $OW_{(m,n)}$ and $EW_{(m,n)}$ denote the pixel values at index (m,n) in the original watermark image and the extracted watermark image, respectively. The $\overline{OW}$ is the mean of the original watermark image, and the $\overline{EW}$ is the mean of the extracted watermark image.

- Bit Error Ratio (BER): The ratio of the total number of errored/corrupted bits to the total number of bits in the image is referred to as the bit error ratio and is calculated using Equation (13).

$$BER = \frac{(TotalErroredBits)}{(TotalBits)} \tag{13}$$

C. IMPERCEPTIBILITY ANALYSIS.

The imperceptibility results of the proposed watermarking scheme are compared to the current state-of-the-art cellular automata-based methods to analyze the relative performance of the proposed scheme. Table 2 includes the multiple image quality metrics to determine the effectiveness of different image watermarking methods in terms of their imperceptibility. The values of these metrics are compared across multiple techniques, namely Ye and Li [24], Adwan et al. [9], Moniruzzaman et al. [10], and the Proposed Scheme. The lower MSE and RMSE values and the higher PSNR, SSIM, and Q-Index values are indicators of the superior perceptual quality of an image. It has been determined that among the evaluated techniques, the proposed scheme consistently outperforms the other assessed schemes in all metrics. It repeatedly results in the lowest MSE values for all the test images, reflecting higher accuracy in watermarked images. The proposed scheme also emphasizes its potential to reduce the average difference between the original and watermarked images by having the lowest RMSE values. Regarding PSNR, the proposed scheme consistently performs better than the other techniques, with an average PSNR value of 58.3735 dB. It can be observed that greater PSNR values are constantly obtained for all the test images in the dataset by the proposed Scheme, which implies greater retention of image details and less quality loss than the other techniques under evaluation. The higher SSIM results further demonstrate the positive aspects of the proposed scheme in retaining the image's structural information and its capacity to preserve the visual composition and attributes of the original images. Finally, the higher values of the perceptual quality metric Q-Index indicate that the proposed scheme generates watermarked images with better overall quality.

After accounting for all the given parameters, we can certainly infer that the proposed scheme demonstrates strong performance in the grayscale image watermarking process. It consistently performs better than the other techniques (Ye and Li [24], Adwan et al. [9], and Moniruzzaman et al. [10]) in terms of lower MSE and RMSE values, and higher PSNR, SSIM, and Q-Index values thus assuring the superior perceptual quality of the watermarked image. The proposed scheme

$$CC(OW, EW) = \frac{\sum_{(m=1)}^{M} \sum_{(n=1)}^{N} (OW_{(m,n)} - \overline{OW})(EW_{(m,n)} - \overline{EW}))}{\sqrt{(\sum_{(m=1)}^{M} \sum_{(n=1)}^{N} (OW_{(m,n)} - \overline{OW})^2)(\sum_{(m=1)}^{M} \sum_{(n=1)}^{N} (EW_{(m,n)} - \overline{EW})^2)}} \quad (12)$$

can be seen as a reliable and high perceptual quality approach for performing digital grayscale image watermarking tasks.

D. ROBUSTNESS ANALYSIS.

This section provides a comprehensive assessment of the robustness of the proposed scheme when subjected to an extensive spectrum of deliberate attacks. The proposed watermarking algorithm has undergone a meticulous evaluation on an extensive dataset comprising 42 diverse test images, aimed at assessing its robustness against a wide spectrum of potential attacks, with the evaluation criterion being the Correlation Coefficient (CC), the Number of Erroneous Bits (NEB), and the Bit Error Ratio (BER). The consequential findings, stemming from the execution of multifarious attack scenarios upon the watermarked images, are comprehensively documented in Tables 3 and 4 for detailed examination and analysis.

1) Robustness Against Cropping Attacks.

An intentional removal or trimming of portions of a digital image with the goal of changing its content or context—possibly leading to misinterpretation or removal of copyright information and watermarks, known as image cropping attacks. It is a kind of modification that compromises the integrity of an image and can vary in scale and intent.

Here are the mathematical representations of the cropping attacks at different segments of the watermarked image *I* with dimensions $M \times N$:

The coordinates representing the top-left corner can be denoted as (x_{start}, y_{start}) and the dimensions of the cropping region as W×H. The cropped image I_{crop} can be obtained using Equation (14):

$$I_{crop} = I(x_{start} : W,\ y_{start} : H) \quad (14)$$

The coordinates representing the top-right corner can be denoted as (x_{end}, y_{start}) and dimensions as W×H. The cropped image I_{crop} crop can be obtained from Equation (15):

$$I_{crop} = I(x_{end} - W + 1 : x_{end},\ y_{start} : H) \quad (15)$$

The coordinates representing the bottom-left corner can be denoted as (x_{start}, y_{end}) and dimensions as W×H. The cropped image I_{crop} crop can be derived using Equation (16):

$$I_{crop} = I(x_{start} : W,\ y_{end} - H + 1 : y_{end}) \quad (16)$$

For the bottom-right corner, the coordinates can be denoted as (x_{end}, y_{end}) and the cropped region could be extracted using Equation (17):

$$I_{crop} = I(x_{end} - W + 1 : x_{end},\ y_{end} - H + 1 : y_{end}) \quad (17)$$

When cropping an image uniformly from all four sides, we usually specify a rectangular region for extraction. This rectangle is defined by providing two coordinates: the top-left corner and the bottom-right corner. Given the top-left corner's coordinates as (x_{start}, y_{start}), and the bottom-right corner's coordinates as (x_{end}, y_{end}), the width of the cropping region is $W = x_{end} - x_{start} + 1$, and the height is $H = y_{end} - y_{start} + 1$. The formula to extract the cropped region is given in the Equation (18):

$$I_{crop} = I(x_{start} : W,\ y_{start} : H) \quad (18)$$

where $W > 0$ and $H > 0$.

This analysis aims to evaluate the robustness of a proposed watermarking technique under various cropping scenarios. The watermarked images are manipulated by applying different cropping percentages such as 6%, 10%, 20%, and 35% at different segments like top-left, top-right, bottom-left, bottom-right, center and all the sides of the watermarked images. The impact of these cropping scenarios on the extracted watermark quality is assessed, specifically considering correlation coefficients and bit error ratios. The analysis, according to Table 3, underscores that the proposed watermarking technique is highly robust as it maintains a high correlation coefficient even with substantial cropping. It also demonstrates that the proposed scheme is effective at preserving watermark integrity across different cropping scenarios.

2) Robustness Against Noise Attacks.

A noise attack denotes the unwanted introduction of stochastic perturbations, or "noise," into watermarked images, leading to a degradation in image fidelity and perceptual acuity. Noise manifests diversely, manifesting as erratic pixel fluctuations within images, the emergence of granular patterns, sporadic conspicuous anomalies in luminosity, or the distortion of chromatic properties. The incursion of noise can be attributed to various sources, encompassing electromagnetic interference, the propagation of signals with imperfections or distortions, or the inherent limitations of data acquisition apparatuses. The watermarked images were subjected to the addition of Speckle and Salt and pepper noises in order to assess the proposed watermarking scheme's robustness against noise attacks. The salt & pepper noises with varying noise densities and multiple variances of the speckle noise were introduced into the watermarked images to evaluate the proposed approach. Table 4 shows the results for the salt & pepper noise with noise densities = 0.01 and 0.1 and speckle noise with variances = 0.01 and 0.1, and the outcomes imply that the proposed scheme is very robust to these kinds of attacks.

The mathematical model that was used for adding 'salt & pepper' noise with density *"d"* to the watermarked images is described in Equation (19). Let *WI* be the watermarked image and *NI* be the noisy watermarked image. Generate a matrix P of the same size as WI, where each element is a random probability value drawn from a standard uniform distribution on the open interval (0, 1). For each pixel in WI:

- $If\ P(x,y) \in (0, \frac{d}{2}),\ set\ NI(x,y)\ to\ 0.$
- $If\ \ P(x,y)\ \in\ (\frac{d}{2}, d),\ \ set\ \ NI(x,y)$ *to the highest possible value within the range of the image's data type.*
- *Otherwise, leave NI(x,y) unchanged.*

(19)

Speckle noise is often described as a multiplicative phenomenon affecting images. In mathematical terms, it's represented by the Equation (20).

$$NI\ =\ WI\ +\ n \cdot WI \tag{20}$$

where NI denotes the resulting noisy image, WI represents the original watermarked image, and n stands for the speckle noise. This noise, characterized by a uniform distribution with a mean of 0 and a variance σ^2, is added to each pixel in WI.

3) Robustness Against Sharpening Attacks.

The image processing procedure designed to heighten image acuity and refine its visual details is sharpening. When it comes to digital image alteration, a "sharpening attack" is an intentional attempt to utilize image sharpening techniques to make a watermark on an image less visible or effective. Such an attack usually aims to hide, modify, or remove a watermark (such as a logo or copyright notice) that has been superimposed on an image in order to preserve its ownership or copyright. Enhancing the contrast and sharpness of the image might make the watermark less visible or perhaps unreadable. To determine how robust the proposed strategy is, the impact of sharpening attacks on the extracted watermark quality is evaluated. We employed the unsharp masking technique to perform the sharpening attacks on the watermarked images. The fundamental equation for unsharp masking is outlined below:

$$SI\ =\ WI\ +\ \alpha \cdot (WI - WI_{blur}) \tag{21}$$

Where WI denotes the input watermarked image and SI represents the sharpened image, WI_{blur} stands for the image post-application of a smoothing filter, such as Gaussian blur, to WI. Additionally, α serves as an amplification factor regulating the potency of the sharpening impact.

The sharpening attacks of varying strengths, i.e., multiple values for the amount variable at multiple radii, are applied to the watermark images. The study based on Table 4 highlights how robust the suggested watermarking method is since it keeps a high correlation coefficient even when there is significant sharpening. It also shows that the suggested scheme works well to maintain watermark integrity under various sharpening conditions.

TABLE 2: Comparison in terms of imperceptibility of the proposed image watermarking scheme with several existing Cellular Automata-based image watermarking schemes.

Host Image Grayscale (8 bits/pixel)	Ye and Li. [24]					Adwan et al. [9]					Moniruzzaman et al. [10]					Proposed Scheme.				
	MSE	RMSE	PSNR	SSIM	Q-Index	MSE	RMSE	PSNR	SSIM	Q-Index	MSE	RMSE	PSNR	SSIM	Q-Index	MSE	RMSE	PSNR	SSIM	Q-Index
House	706.788	26.5855	19.6379	0.6311	0.9946	3.5433	1.8824	42.6368	0.988	0.9998	0.4315	0.6569	51.7808	0.9965	1	0.3165	0.5626	53.1273	0.9963	1
Jelly beans	807.4062	28.4149	19.0599	0.6258	0.9938	3.4273	1.8513	42.7813	0.9884	0.9999	0.4323	0.6575	51.7727	0.9956	1	0.3075	0.5545	53.2526	0.9957	1
Fishing Boat	107.4458	10.3656	27.8189	0.9345	0.9994	3.5374	1.8808	42.6439	0.992	0.9978	0.1081	0.3289	57.7907	0.9994	0.9999	0.0798	0.2824	59.1122	0.9993	1
Splash	219.654	14.8207	24.7134	0.8769	0.9843	3.4796	1.8654	42.7155	0.9871	0.9898	0.1076	0.328	57.8126	0.9989	0.9987	0.0785	0.2801	59.1843	0.9994	1
Mandrill	142.4066	11.9334	26.5955	0.9513	0.9984	3.4968	1.87	42.6941	0.9967	0.9998	0.1094	0.3308	57.7394	0.9997	1	0.0787	0.2805	59.1716	1	1
Peppers	134.3881	11.5926	26.8472	0.9049	0.9965	3.4876	1.8675	42.7056	0.9901	0.9994	0.1067	0.3266	57.8505	0.9992	1	0.0776	0.2785	59.2346	0.9995	1
Stream & bridge	154.5128	12.4303	26.2412	0.9341	0.9985	2.3214	1.5236	44.4734	0.9997	0.9999	0.0948	0.3079	58.3621	0.9997	1	0.0501	0.2238	61.1339	0.9999	1
Truck	123.524	11.1141	27.2133	0.9025	0.9982	3.1032	1.7616	43.2127	0.9928	0.9997	0.1098	0.3314	57.7247	0.9993	1	0.0742	0.2725	59.4246	0.9997	1
Pentagon	24.4735	4.9471	34.2438	0.9845	0.9999	3.3315	1.8252	42.9044	0.9936	0.9999	0.0281	0.1678	63.6373	0.9999	1	0.0196	0.1401	65.2026	0.9999	1
Male	49.4194	7.0299	31.1918	0.9724	0.9968	3.3181	1.8216	42.922	0.9857	0.9742	0.0273	0.1652	63.7709	0.9998	0.9995	0.0191	0.1381	65.3259	0.9999	1
W9CCDB54	141.9224	11.9131	26.6103	0.9414	0.9992	3.5038	1.8718	42.6854	0.9974	0.9999	0.1065	0.3264	57.8561	0.9998	1	0.0773	0.2781	59.2483	0.9999	1
W9CIL54816	418.3748	20.4542	21.9151	0.8796	0.9982	3.4951	1.8695	42.6962	0.9879	0.9999	N/A	N/A	N/A	N/A	N/A	0.0665	0.2579	59.9031	0.9992	1
Brain MRI 55	1321.8654	36.3575	16.9189	0.6654	0.8857	3.5439	1.8825	42.636	0.9683	0.9053	0.4535	0.6734	51.5653	0.9969	0.9716	0.307	0.554	53.26	0.9965	0.9654
Brain MRI 63	1371.5278	37.0341	16.7588	0.6728	0.8771	3.5334	1.8797	42.6489	0.9673	0.9039	0.4483	0.6695	51.6152	0.997	0.9732	0.3125	0.559	53.1827	0.9968	0.9673
Brain MRI 70	1240.2223	35.2168	17.1958	0.6783	0.8869	3.4719	1.8633	42.7252	0.9656	0.8965	0.4599	0.6782	51.5037	0.9968	0.9675	0.3096	0.5564	53.2224	0.9969	0.973
Chest CT 30	1195.4414	34.5752	17.3555	0.6182	0.8647	3.4924	1.8688	42.6995	0.9861	0.9708	0.4261	0.6528	51.8355	0.9973	0.9935	0.3117	0.5583	53.1929	0.9969	0.9836
Chest CT 60	1008.4899	31.7567	18.0941	0.6612	0.9002	3.4553	1.8588	42.746	0.9886	0.9861	0.4167	0.6456	51.9321	0.9975	0.9931	0.3088	0.5557	53.2341	0.9977	0.9951
Chest CT 90	906.1802	30.1028	18.5587	0.6429	0.926	3.4841	1.8666	42.7099	0.9898	0.9867	0.4278	0.654	51.8186	0.9976	0.9981	0.3123	0.5588	53.1855	0.9976	0.9929
im0001	116.7653	10.8058	27.4577	0.9203	0.9884	3.5229	1.8769	42.6618	0.9808	0.9886	N/A	N/A	N/A	N/A	N/A	0.0498	0.2232	61.1577	0.9994	0.9944
im0100	142.0792	11.9197	26.6055	0.9165	0.983	3.4819	1.866	42.7126	0.9815	0.9882	N/A	N/A	N/A	N/A	N/A	0.0477	0.2183	61.3478	0.9994	0.9949
im0200	110.341	10.5043	27.7034	0.9375	0.9912	3.4785	1.8651	42.7169	0.9814	0.9779	N/A	N/A	N/A	N/A	N/A	0.0485	0.2203	61.2691	0.9995	0.9987
im0300	165.518	12.8654	25.9424	0.9307	0.9831	3.5117	1.874	42.6756	0.9795	0.9819	N/A	N/A	N/A	N/A	N/A	0.0484	0.2199	61.2846	0.9995	0.9995
im0400	187.5904	13.6964	25.3987	0.9216	0.9767	3.5083	1.873	42.6799	0.9814	0.9882	N/A	N/A	N/A	N/A	N/A	0.0483	0.2199	61.2875	0.9994	0.9967
Galen Clark	219.0054	14.7988	24.7263	0.9173	0.9988	3.4882	1.8677	42.7048	0.9903	0.9996	N/A	N/A	N/A	N/A	N/A	0.0493	0.222	61.2026	0.9995	1
General Douglas	126.4171	11.2435	27.1127	0.9361	0.9987	3.5261	1.8778	42.6579	0.9899	0.9997	N/A	N/A	N/A	N/A	N/A	0.0495	0.2225	61.1849	0.9995	1
Turbine Power-house	81.485	9.0269	29.02	0.9484	0.9996	3.5431	1.8823	42.637	0.991	0.9997	N/A	N/A	N/A	N/A	N/A	0.0491	0.2215	61.2241	0.9996	1
Shipyard	82.3662	9.0756	28.9733	0.9646	0.9986	3.4589	1.8598	42.7414	0.9863	0.9991	0.027	0.1644	63.8128	0.9997	0.9999	0.0187	0.1368	65.4111	0.9997	1
Full Moon	1932.9835	43.9657	15.2685	0.5331	0.5945	3.4339	1.8531	42.7729	0.8521	0.4952	0.3377	0.5811	52.846	0.9951	0.6737	0.1859	0.4312	55.437	0.9959	0.7825
Venus	1418.0432	37.6569	16.6139	0.5307	0.8388	3.5109	1.8737	42.6767	0.9267	0.8234	0.4609	0.6789	51.4945	0.9917	0.8314	0.3257	0.5707	53.0022	0.9906	0.901
Astronaut L. Gordon Cooper	280.3742	16.7444	23.6534	0.8875	0.9914	3.5046	1.8721	42.6844	0.9922	0.9986	0.1089	0.3301	57.7586	0.9992	0.9991	0.0781	0.2794	59.205	0.9994	0.9998
Panoramic view	543.8697	23.321	20.7759	0.8449	0.8448	1.9811	1.4075	45.1617	0.9827	0.9566	0.1719	0.4146	55.7774	0.9971	0.564	0.1085	0.3293	57.7778	0.9939	0.9368
Television (TV) monitor	103.2832	10.1628	27.9905	0.9299	0.9993	3.5	1.8708	42.6901	0.9913	0.9986	0.1057	0.3252	57.8891	0.9993	1	0.0776	0.2785	59.2338	0.9995	1
Wallops Island	283.8586	16.8481	23.5998	0.9104	0.9834	3.4603	1.8602	42.7397	0.9859	0.9795	0.1058	0.3253	57.8862	0.9994	0.9984	0.077	0.2774	59.2672	0.9993	0.9999

TABLE 2 Continued: Comparison in terms of imperceptibility of the proposed image watermarking scheme with several existing Cellular Automata-based image watermarking schemes.

Group Photo	*23.5546*	*4.8533*	*34.4101*	*0.9848*	*0.9999*	*3.4784*	*1.8651*	*42.717*	*0.9893*	*0.9966*	*0.0269*	*0.164*	*63.8354*	*0.9998*	*0.9998*	*0.0198*	*0.1408*	*65.1603*	*0.9997*	*1*
onion	*1826.7066*	*42.74*	*15.5141*	*0.2221*	*0.9379*	*3.4554*	*1.8589*	*42.7458*	*0.9884*	*0.9996*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.7466*	*0.8641*	*49.3998*	*0.9944*	*1*
pout	*390.0869*	*19.7506*	*22.2192*	*0.6407*	*0.9948*	*3.6359*	*1.9068*	*42.5246*	*0.9881*	*0.9998*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.2862*	*0.5349*	*53.5648*	*0.9969*	*1*
cameraman	*865.088*	*29.4124*	*18.7602*	*0.6505*	*0.9449*	*3.4359*	*1.8536*	*42.7703*	*0.9872*	*0.966*	*0.4228*	*0.6502*	*51.8699*	*0.9967*	*0.9919*	*0.3079*	*0.5549*	*53.2466*	*0.9962*	*0.9994*
forest	*585.955*	*24.2065*	*20.4522*	*0.8133*	*0.9434*	*4.2363*	*2.0582*	*41.861*	*0.9919*	*0.9903*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.1577*	*0.3971*	*56.1529*	*0.9996*	*1*
spine	*432.4444*	*20.7953*	*21.7715*	*0.8352*	*0.9192*	*2.499*	*1.5808*	*44.1532*	*0.9767*	*0.9487*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.1506*	*0.3881*	*56.3529*	*0.9949*	*0.9621*
lighthouse	*143.2877*	*11.9703*	*26.5687*	*0.9186*	*0.9962*	*3.4674*	*1.8621*	*42.7308*	*0.9884*	*0.9994*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.0662*	*0.2573*	*59.9236*	*0.9993*	*1*
fabric	*113.3036*	*10.6444*	*27.5884*	*0.9412*	*0.9983*	*3.4939*	*1.8692*	*42.6977*	*0.996*	*0.9986*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.0661*	*0.2571*	*59.9279*	*0.9998*	*1*
flamingos	*45.3042*	*6.7308*	*31.5694*	*0.9788*	*0.9984*	*3.5002*	*1.8709*	*42.6899*	*0.992*	*0.9996*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*0.0161*	*0.1268*	*66.0691*	*1*	*1*

TABLE 3: Robustness results of the proposed scheme under different cropping attacks across a dataset of forty-two distinct test host images, with the Correlation Coefficient (CC), NO. of Erroneous Bits (NEB), and Bit Error Ratio (BER) being the evaluation criterion.

Host Image Grayscale (8 bits/pixel)	*Crop All Sides (6%)*			*Crop Top-Left (10%)*			*Crop Top-Right (10%)*			*Crop Bottom-Left (20%)*			*Crop Bottom-Right (20%)*			*Crop Center (35%)*		
	CC	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*
House	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Jelly beans	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Fishing Boat	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Splash	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Mandrill	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Peppers	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Stream & bridge	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Truck	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Pentagon	*0.6258*	*2335*	*0.0356*	*0.9169*	*437*	*0.0067*	*0.9206*	*434*	*0.0066*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Male	*0.6258*	*2335*	*0.0356*	*0.9169*	*437*	*0.0067*	*0.9206*	*434*	*0.0066*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
W9CCDB54	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
W9CIL54816	*0.8645*	*809*	*0.0123*	*0.9441*	*313*	*0.0048*	*0.941*	*342*	*0.0052*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Brain MRI 55	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Brain MRI 63	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Brain MRI 70	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Chest CT 30	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Chest CT 60	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Chest CT 90	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
im0001	*0.8285*	*1044*	*0.0159*	*0.9196*	*440*	*0.0067*	*0.915*	*458*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
im0100	*0.8285*	*1044*	*0.0159*	*0.9196*	*440*	*0.0067*	*0.915*	*458*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
im0200	*0.8285*	*1044*	*0.0159*	*0.9196*	*440*	*0.0067*	*0.915*	*458*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
im0300	*0.8285*	*1044*	*0.0159*	*0.9196*	*440*	*0.0067*	*0.915*	*458*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*

TABLE 3 Continued: Robustness results of the proposed scheme under different cropping attacks, across a dataset of forty-two distinct test host images, with the Correlation Coefficient (CC), NO. of Erroneous Bits (NEB), and Bit Error Ratio (BER) being the evaluation criterion.

im0400	*0.8285*	*1044*	*0.0159*	*0.9196*	*440*	*0.0067*	*0.915*	*458*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Galen Clark	*0.8133*	*1125*	*0.0172*	*0.9065*	*491*	*0.0075*	*0.9154*	*456*	*0.007*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
General Douglas	*0.8537*	*876*	*0.0134*	*0.912*	*463*	*0.0071*	*0.9103*	*467*	*0.0071*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Turbine Power House	*0.8537*	*876*	*0.0134*	*0.912*	*463*	*0.0071*	*0.9103*	*467*	*0.0071*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Shipyard	*0.6258*	*2335*	*0.0356*	*0.9169*	*437*	*0.0067*	*0.9206*	*434*	*0.0066*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Full Moon	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Venus	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
Astronaut L. Gordon Cooper	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Panoramic view	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Television (TV) monitor	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Wallops Island	*0.9084*	*535*	*0.0082*	*0.9506*	*263*	*0.004*	*0.9488*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
Group Photo	*0.6258*	*2335*	*0.0356*	*0.9169*	*437*	*0.0067*	*0.9206*	*434*	*0.0066*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
onion	*0.9766*	*114*	*0.0017*	*0.9956*	*24*	*3.66e-4*	*0.9941*	*32*	*4.88e-4*	*1*	*0*	*0*	*1*	*0*	*0*	*0.7833*	*1399*	*0.0213*
pout	*0.9631*	*186*	*0.0028*	*0.9863*	*71*	*0.0011*	*0.9868*	*65*	*9.92e-4*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
cameraman	*0.9632*	*195*	*0.003*	*0.9854*	*75*	*0.0011*	*0.9844*	*86*	*0.0013*	*1*	*0*	*0*	*1*	*0*	*0*	*0.9637*	*148*	*0.0023*
forest	*0.9037*	*548*	*0.0084*	*0.9717*	*155*	*0.0024*	*0.9695*	*163*	*0.0025*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
spine	*0.9112*	*512*	*0.0078*	*0.9642*	*197*	*0.003*	*0.9563*	*231*	*0.0035*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
lighthouse	*0.8781*	*719*	*0.011*	*0.9425*	*331*	*0.0051*	*0.9422*	*354*	*0.0054*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
fabric	*0.8907*	*649*	*0.0099*	*0.9414*	*328*	*0.005*	*0.9478*	*302*	*0.0046*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*
flamingos	*0.6203*	*2390*	*0.0365*	*0.9096*	*498*	*0.0076*	*0.9143*	*480*	*0.0073*	*1*	*0*	*0*	*1*	*0*	*0*	*1*	*0*	*0*

TABLE 4: Robustness Analysis of the proposed methodology against perturbations induced by noise and sharpening attacks, across a dataset of forty-two distinct test host images, with the evaluation criterion being the Correlation Coefficient (CC), NO. of Erroneous Bits (NEB), and Bit Error Ratio (BER).

Host Image Grayscale (8 bits/pixel)	*Salt & Pepper Noise Density = (0.01)*			*Salt & Pepper Noise Density = (0.1)*			*Speckle Variance = (0.01)*			*Speckle Variance = (0.1)*			*Sharpening (Radius = 0.3 & Amount = 0.5)*			*Sharpening (Radius = 0.4 & Amount = 0.1)*		
	CC	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*	*CC*	*NEB*	*BER*
House	*0.9964*	*18*	*2.75e-4*	*0.9679*	*164*	*0.0025*	*0.9826*	*87*	*0.0013*	*0.9506*	*255*	*0.0039*	*1*	*0*	*0*	*1*	*0*	*0*
Jelly beans	*0.9981*	*10*	*1.53e-4*	*0.9667*	*164*	*0.0025*	*0.9861*	*73*	*0.0011*	*0.9598*	*215*	*0.0033*	*1*	*0*	*0*	*1*	*0*	*0*
Fishing Boat	*0.9977*	*10*	*1.53e-4*	*0.9692*	*162*	*0.0025*	*0.985*	*77*	*0.0012*	*0.9561*	*232*	*0.0035*	*0.9999*	*1*	*1.53e-5*	*0.9884*	*66*	*0.001*
Splash	*0.9993*	*7*	*1.07e-4*	*0.9642*	*171*	*0.0026*	*0.9874*	*68*	*0.001*	*0.9563*	*225*	*0.0034*	*1*	*0*	*0*	*0.9995*	*2*	*3.05e-5*
Mandrill	*0.9985*	*7*	*1.07e-4*	*0.9685*	*171*	*0.0026*	*0.9858*	*76*	*0.0012*	*0.9613*	*213*	*0.0033*	*0.9939*	*31*	*4.73e-4*	*0.7494*	*1589*	*0.0242*
Peppers	*0.9984*	*8*	*1.22e-4*	*0.9713*	*155*	*0.0024*	*0.9853*	*79*	*0.0012*	*0.9566*	*233*	*0.0036*	*1*	*0*	*0*	*0.9989*	*6*	*9.16e-5*
Stream and bridge	*0.9984*	*8*	*1.22e-4*	*0.9623*	*185*	*0.0028*	*0.9858*	*75*	*0.0011*	*0.954*	*249*	*0.0038*	*0.998*	*16*	*2.44e-4*	*0.8942*	*618*	*0.0094*
Truck	*0.9986*	*8*	*1.22e-4*	*0.9703*	*152*	*0.0023*	*0.9871*	*70*	*0.0011*	*0.9506*	*245*	*0.0037*	*1*	*0*	*0*	*0.9994*	*3*	*4.58e-5*

TABLE 4 Continued: Robustness Analysis of the proposed methodology against perturbations induced by noise and sharpening attacks, across a dataset of forty-two distinct test host images, with the evaluation criterion being the Correlation Coefficient (CC), NO. of Erroneous Bits (NEB), and Bit Error Ratio (BER).

Pentagon	*0.9982*	*7*	*1.07e-4*	*0.9663*	*172*	*0.0026*	*0.983*	*90*	*0.0014*	*0.9541*	*238*	*0.0036*	*1*	*0*	*0*	*0.9897*	*50*	*7.63e-4*
Male	*0.9964*	*17*	*2.59e-4*	*0.9709*	*154*	*0.0023*	*0.985*	*76*	*0.0012*	*0.9574*	*228*	*0.0035*	*1*	*0*	*0*	*0.9995*	*2*	*3.05e-5*
W9CCDB54	*0.9958*	*20*	*3.05e-4*	*0.9725*	*141*	*0.0022*	*0.9881*	*68*	*0.001*	*0.9524*	*251*	*0.0038*	*1*	*0*	*0*	*1*	*0*	*0*
W9CIL54816	*0.9991*	*6*	*9.16e-5*	*0.9681*	*167*	*0.0025*	*0.9855*	*73*	*0.0011*	*0.9587*	*231*	*0.0035*	*1*	*0*	*0*	*1*	*0*	*0*
Brain MRI 55	*0.9982*	*7*	*1.07e-4*	*0.9633*	*193*	*0.0029*	*0.9849*	*80*	*0.0012*	*0.9551*	*234*	*0.0036*	*0.9995*	*2*	*3.05e-5*	*0.9688*	*171*	*0.0026*
Brain MRI 63	*0.9989*	*7*	*1.07e-4*	*0.9692*	*162*	*0.0025*	*0.9857*	*70*	*0.0011*	*0.9545*	*237*	*0.0036*	*0.9974*	*11*	*1.68e-4*	*0.9353*	*349*	*0.0053*
Brain MRI 70	*0.9989*	*6*	*9.16e-5*	*0.9664*	*178*	*0.0027*	*0.9848*	*89*	*0.0014*	*0.9557*	*231*	*0.0035*	*0.9998*	*4*	*6.10e-5*	*0.9619*	*215*	*0.0033*
Chest CT 30	*0.9978*	*12*	*1.83e-4*	*0.9671*	*170*	*0.0026*	*0.9846*	*85*	*0.0013*	*0.9488*	*257*	*0.0039*	*1*	*0*	*0*	*0.9849*	*65*	*9.92e-4*
Chest CT 60	*0.9986*	*8*	*1.22e-4*	*0.97*	*156*	*0.0024*	*0.9854*	*83*	*0.0013*	*0.9538*	*242*	*0.0037*	*1*	*0*	*0*	*0.992*	*44*	*6.71e-4*
Chest CT 90	*0.9987*	*7*	*1.07e-4*	*0.9708*	*156*	*0.0024*	*0.9857*	*72*	*0.0011*	*0.9522*	*249*	*0.0038*	*1*	*0*	*0*	*0.9918*	*47*	*7.17e-4*
im0001	*0.9978*	*9*	*1.37e-4*	*0.9704*	*164*	*0.0025*	*0.9856*	*78*	*0.0012*	*0.9491*	*265*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*
im0100	*0.9993*	*3*	*4.58e-5*	*0.9657*	*179*	*0.0027*	*0.9854*	*78*	*0.0012*	*0.953*	*250*	*0.0038*	*1*	*0*	*0*	*1*	*0*	*0*
im0200	*0.9978*	*11*	*1.68e-4*	*0.9711*	*155*	*0.0024*	*0.982*	*85*	*0.0013*	*0.9477*	*282*	*0.0043*	*1*	*0*	*0*	*1*	*0*	*0*
im0300	*0.999*	*7*	*1.07e-4*	*0.9685*	*162*	*0.0025*	*0.9871*	*69*	*0.0011*	*0.9574*	*233*	*0.0036*	*1*	*0*	*0*	*1*	*0*	*0*
im0400	*0.9976*	*10*	*1.53e-4*	*0.9719*	*151*	*0.0023*	*0.9874*	*70*	*0.0011*	*0.9511*	*266*	*0.0041*	*1*	*0*	*0*	*1*	*0*	*0*
Galen Clark	*0.9989*	*6*	*9.16e-5*	*0.9656*	*179*	*0.0027*	*0.9836*	*82*	*0.0013*	*0.9508*	*242*	*0.0037*	*1*	*0*	*0*	*1*	*0*	*0*
General Douglas	*0.9984*	*8*	*1.22e-4*	*0.9695*	*155*	*0.0024*	*0.9829*	*78*	*0.0012*	*0.9503*	*261*	*0.004*	*1*	*0*	*0*	*0.9988*	*7*	*1.07e-4*
Turbine Power House	*0.9985*	*10*	*1.53e-4*	*0.9709*	*160*	*0.0024*	*0.9858*	*76*	*0.0012*	*0.954*	*247*	*0.0038*	*1*	*0*	*0*	*0.9995*	*2*	*3.05e-5*
Shipyard	*0.9993*	*5*	*7.63e-5*	*0.9651*	*180*	*0.0027*	*0.9879*	*69*	*0.0011*	*0.965*	*206*	*0.0031*	*1*	*0*	*0*	*0.9977*	*14*	*2.14e-4*
Full Moon	*0.9989*	*6*	*9.16e-5*	*0.9709*	*150*	*0.0023*	*0.9825*	*86*	*0.0013*	*0.9557*	*236*	*0.0036*	*0.9973*	*13*	*1.98e-4*	*0.9875*	*68*	*0.001*
Venus	*0.999*	*5*	*7.63e-5*	*0.9643*	*184*	*0.0028*	*0.9865*	*74*	*0.0011*	*0.9466*	*273*	*0.0042*	*0.9923*	*47*	*7.17e-4*	*0.983*	*94*	*0.0014*
Astronaut L. Gordon Cooper	*0.9986*	*9*	*1.37e-4*	*0.9703*	*158*	*0.0024*	*0.9813*	*95*	*0.0014*	*0.9534*	*252*	*0.0038*	*0.9671*	*164*	*0.0025*	*0.8148*	*1130*	*0.0172*
Panoramic view	*0.9986*	*7*	*1.07e-4*	*0.9698*	*158*	*0.0024*	*0.9825*	*88*	*0.0013*	*0.9564*	*239*	*0.0036*	*1*	*0*	*0*	*1*	*0*	*0*
Television (TV) monitor	*0.9985*	*7*	*1.07e-4*	*0.9682*	*163*	*0.0025*	*0.9839*	*82*	*0.0013*	*0.9528*	*250*	*0.0038*	*1*	*0*	*0*	*0.956*	*239*	*0.0036*
Wallops Island	*0.9974*	*13*	*1.98e-4*	*0.9677*	*169*	*0.0026*	*0.9844*	*83*	*0.0013*	*0.952*	*249*	*0.0038*	*0.9899*	*56*	*8.54e-4*	*0.9294*	*388*	*0.0059*
Group Photo	*0.9991*	*4*	*6.10e-5*	*0.9669*	*180*	*0.0027*	*0.9847*	*81*	*0.0012*	*0.9597*	*233*	*0.0036*	*1*	*0*	*0*	*1*	*0*	*0*
onion	*0.9972*	*15*	*2.29e-4*	*0.9719*	*156*	*0.0024*	*0.9871*	*72*	*0.0011*	*0.9569*	*238*	*0.0036*	*1*	*0*	*0*	*0.9951*	*23*	*3.51e-4*
pout	*0.9973*	*14*	*2.14e-4*	*0.9679*	*164*	*0.0025*	*0.9855*	*75*	*0.0011*	*0.953*	*247*	*0.0038*	*1*	*0*	*0*	*1*	*0*	*0*
cameraman	*0.9991*	*5*	*7.63e-5*	*0.9697*	*157*	*0.0024*	*0.9857*	*72*	*0.0011*	*0.9473*	*269*	*0.0041*	*0.9908*	*44*	*6.71e-4*	*0.9308*	*406*	*0.0062*
forest	*0.9997*	*2*	*3.05e-5*	*0.963*	*186*	*0.0028*	*0.9853*	*81*	*0.0012*	*0.9614*	*214*	*0.0033*	*0.9969*	*13*	*1.98e-4*	*0.9045*	*548*	*0.0084*
spine	*0.998*	*8*	*1.22e-4*	*0.9679*	*168*	*0.0026*	*0.9859*	*73*	*0.0011*	*0.9504*	*263*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*
lighthouse	*0.9981*	*10*	*1.53e-4*	*0.9704*	*156*	*0.0024*	*0.9855*	*73*	*0.0011*	*0.9548*	*264*	*0.004*	*1*	*0*	*0*	*1*	*0*	*0*
fabric	*0.9998*	*4*	*6.10e-5*	*0.9651*	*178*	*0.0027*	*0.9835*	*85*	*0.0013*	*0.9486*	*277*	*0.0042*	*1*	*0*	*0*	*0.9955*	*23*	*3.51e-4*
flamingos	*0.9991*	*6*	*9.16e-5*	*0.9685*	*159*	*0.0024*	*0.9819*	*92*	*0.0014*	*0.9491*	*261*	*0.004*	*0.9995*	*2*	*3.05e-5*	*0.9959*	*23*	*3.51e-4*

E. COMPUTATIONAL TIME ANALYSIS.

Table 5 and Table 6 present a comparative analysis of embedding and extraction times (in seconds), respectively, for watermarking methods across different host images and watermark sizes. Four methods are evaluated with embedding and extraction times provided for two watermark sizes: 64×128 and 128×164 pixels. The analysis reveals significant variation in embedding and extraction times based on the method and image characteristics. Notably, the proposed scheme consistently demonstrates lower outcomes for both embedding as well as extraction times than other methods across almost all scenarios listed in the tables. This observation underscores its remarkable prowess and effectiveness in the processes of embedding and extracting watermarks from images, showcasing unmatched performance and efficiency. Factors influencing embedding and extraction times include algorithm complexity, computational efficiency, and host image and watermark characteristics. Ultimately, the choice of watermarking method should consider a balance between embedding and extraction times, robustness, and other relevant factors to meet the requirements of the intended application.

Table 7 presents detailed data on key generation times (in seconds) for various watermarking methods across different host images and watermark sizes. The proposed scheme consistently exhibits notably longer key generation times compared to other methods, particularly for larger host image sizes and watermark dimensions. This extended duration results from a careful and thorough approach taken during the key generation process aimed at enhancing the security of the watermarking system. The scheme invests more time and effort into generating keys with a high level of complexity and randomness, making them more resistant to various attacks and unauthorized access. Despite the longer key generation times, the proposed scheme offers heightened security benefits due to the thoroughness of its CA key generation process. However, the trade-off between longer key generation times and enhanced security should be carefully weighed against the requirements of specific applications.

This prolonged duration is also attributable to our system being implemented on a sequential architecture which means that the parallelism property of CA is not being taken advantage of at all. Incorporating CAD systems for concurrent implementation [53] to leverage the parallelism property of cellular automata presents a promising solution to mitigate the longer time required for key generation in our system. By harnessing the computational power of parallel processing, we can distribute the workload across multiple computing units, thereby accelerating the key generation process. This optimization not only addresses the current bottleneck but also enhances the efficiency and scalability of our system. Leveraging CAD systems for key generation underscores our commitment to optimizing performance while ensuring robust security measures.

VI. CONCLUSION AND FUTURE WORK

Cellular automata (CA) are a notable and sophisticated model that has shown considerable potential in improving the security and reliability of digital assets when it comes to digital image watermarking. CA-based watermarking techniques have demonstrated several impressive qualities, such as robustness to frequent attacks, the embedded watermarks' imperceptibility, and effective data embedding in images. Through this research, we have explored a CA-based watermarking strategy governed by Rule 30, which demonstrated its efficacy in protecting digital images against unauthorized use, unauthorized modification, and infringement on intellectual property. The experimental outcomes across a diverse range of standardized attack scenarios establish the ascendancy of the proposed algorithm over competing methodologies in the field of image watermarking.

Even if there have been a lot of noteworthy advances in the field of CA-based digital image watermarking, there are still a lot of uncharted territories that want more research and development. Fortifying CA-based watermarking approaches against a larger range of threats should be the primary focus of future research endeavors. Making these techniques more resistant to a wider range of attacks, such as geometric transformations, compression-induced distortions, and complex signal processing techniques, may involve discovering more complex rule sets and carefully integrating machine learning paradigms. The security component must be given prominent attention. The vulnerability of spatial domain watermarking techniques to complex and advanced attacks should be thoroughly examined in future investigations. One of the key goals is to maximize the watermarking capacity while preserving the image's visual quality. Subsequent research endeavors may explore and develop algorithms that augment data-hiding efficacy, guaranteeing that watermarks remain undetectable while concurrently optimizing the amount of information that may be incorporated. A promising direction for future research is to modify CA-based watermarking techniques for real-time use. This could apply to the watermarking of videos or the protection of streaming media, where quick data processing is critical. Meanwhile, one of our future plans is to consider modern deep-learning techniques [54] that have revolutionized the field of digital image watermarking, offering new possibilities for robust and imperceptible watermark embedding and extraction. Deep learning models, such as convolutional neural networks (CNNs), can learn complex representations and patterns from large amounts of image data, enabling them to effectively encode and decode watermarks in images.

To sum up, the integration of cellular automata with digital image watermarking has shown considerable potential and cracked the gate to further study and development. Dedicated efforts in this area have the potential to offer safe and effective ways to protect digital assets in an increasingly digital environment as the digital world grows and develops over time.

TABLE 5: Embedding Time in Seconds of Watermarking Methods for Different Host Images and Watermarks

Host Image Grayscale (8 bits/pixel)	Host Image Size (pixels)	Watermark Size (64×128)				Watermark Size (128×164)			
		Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.	Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.
onion	198×135	0.159809	0.036489	N/A	**0.025728**	0.128354	0.107455	N/A	**0.030702**
House	256×256	0.124157	0.072592	0.899457	**0.020502**	0.123899	0.088318	0.730469	**0.036032**
Splash	512×512	0.632889	0.038331	2.915754	**0.033725**	0.559585	0.052262	2.582008	**0.039807**
Pentagon	1024×1024	1.300323	0.087177	10.707748	**0.021180**	1.323873	0.099821	10.901481	**0.037367**

TABLE 6: Extraction Time in Seconds of Watermarking Methods for Different Host Images and Watermarks

Host Image Grayscale (8 bits/pixel)	Host Image Size (pixels)	Watermark Size (64×128)				Watermark Size (128× 164)			
		Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.	Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.
onion	198×135	0.018147	0.041306	N/A	0.031554	0.060712	0.078757	N/A	**0.02669**
House	256×256	0.032714	0.055386	0.657102	**0.032163**	0.052649	0.091742	0.653864	**0.041124**
Splash	512×512	0.124647	0.069233	2.505042	**0.035088**	0.108714	0.118865	2.357764	**0.028482**
Pentagon	1024×1024	0.28955	0.036978	11.034892	**0.026006**	0.318926	0.070721	11.237124	**0.056624**

TABLE 7: Key Generation Time in Seconds of Watermarking Methods for Different Host Images and Watermarks

Host Image Grayscale (8 bits/pixel)	Host Image Size (pixels)	Watermark Size (64×128)				Watermark Size (128× 164)			
		Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.	Ye and Li [24]	Adwan et al. [9]	Moniruzzaman et al. [10]	Proposed Scheme.
onion	198×135	0.052517	0.140891	N/A	109.8306	0.125685	0.376386	N/A	244.3651
House	256×256	0.074302	0.105079	1.031523	121.8449	0.084188	0.220986	0.704578	257.2078
Splash	512×512	0.380358	0.200555	3.06983	126.5283	0.231141	0.340498	2.750723	225.1555
Pentagon	1024×1024	2.42462	0.241485	11.469756	159.2139	3.140535	0.288682	11.853642	207.0514

REFERENCES

[1] X. Zhong, A. Das, F. Alrasheedi, and A. Tanvir, "A brief, in-depth survey of deep learning-based image watermarking," *Applied Sciences*, vol. 13, no. 21, p. 11852, 2023.

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

[3] S. M. Mousavi, A. Naghsh, and S. Abu-Bakar, "Watermarking techniques used in medical images: a survey," *Journal of digital imaging*, vol. 27, pp. 714–729, 2014.

[4] S. L. Gomez-Coronel, E. Moya-Albor, J. Brieva, and A. Romero-Arellano, "A robust and secure watermarking approach based on hermite transform and svd-dct," *Applied Sciences*, vol. 13, no. 14, p. 8430, 2023.

[5] A. F. Qasim, F. Meziane, and R. Aspin, "Digital watermarking: Applicability for developing trust in medical imaging workflows state of the art review," *Computer Science Review*, vol. 27, pp. 45–60, 2018.

[6] F. N. Al-Wesabi, F. Alrowais, H. G. Mohamed, M. A. Duhayyim, A. M. Hilal, and A. Motwakel, "Heuristic optimization algorithm based watermarking on content authentication and tampering detection for english text," *IEEE Access*, vol. 11, pp. 86 104–86 111, 2023.

[7] A. Menendez-Ortiz, C. Feregrino-Uribe, R. Hasimoto-Beltran, and J. J. Garcia-Hernandez, "A survey on reversible watermarking for multimedia content: A robustness overview," *IEEE Access*, vol. 7, pp. 132 662–132 681, 2019.

[8] B. M. S. Hasan, S. Y. Ameen, and O. M. S. Hasan, "Image authentication based on watermarking approach," *Asian Journal of Research in Computer Science*, vol. 9, no. 3, pp. 34–51, 2021.

[9] O. Adwan, A. A. Awwad, A. Sleit, and A. L. A. Alhoum, "A novel watermarking scheme based on two dimensional cellular automata," in Proceedings of the International Conference on Computers and Computing, World Scientific and Engineering Academy and Society (WSEAS). Canary Islands, Spain, 2011, pp. 88–94.

[10] M. Moniruzzaman, M. A. K. Hawlader, and M. F. Hossain, "Watermarking scheme based on game of life cellular automaton," in *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*. IEEE, 2014, pp. 1–6.

[11] T. A. BW, F. P. Permana et al., "Medical image watermarking with tamper detection and recovery using reversible watermarking with lsb modification and run length encoding (rle) compression," in *2012 IEEE International Conference on Communication, Networks and Satellite (ComNetSat)*. IEEE, 2012, pp. 167–171.

[12] G. Manjula and A. Danti, "A novel hash based least significant bit (2-3-3) image steganography in spatial domain," *arXiv preprint arXiv:1503.03674*, 2015.

[13] J. Abraham and V. Paul, "An imperceptible spatial domain color image watermarking scheme," *Journal of King Saud University-Computer and Information Sciences*, vol. 31, no. 1, pp. 125–133, 2019.

[14] A. Zeki, A. Abubakar, and H. Chiroma, "An intermediate significant bit (isb) watermarking technique using neural networks," *SpringerPlus*, vol. 5, pp. 1–25, 2016.

[15] H. Zhang, C. Wang, and X. Zhou, "Fragile watermarking based on lbp for blind tamper detection in images," *Journal of Information Processing Systems*, vol. 13, no. 2, pp. 385–399, 2017.

[16] V. Kelkar, K. Tuckley, H. Nemade et al., "Novel variants of a histogram shift-based reversible watermarking technique for medical images to improve hiding capacity," *Journal of healthcare engineering*, vol. 2017, 2017.

[17] M. Nasir, W. Jadoon, I. A. Khan, N. Gul, S. Shah, M. ELAffendi, and A. Muthanna, "Secure reversible data hiding in images based on linear prediction and bit-plane slicing," *Mathematics*, vol. 10, no. 18, p. 3311, 2022.

[18] F. Zhang, T. Luo, G. Jiang, M. Yu, H. Xu, and W. Zhou, "A novel robust color image watermarking method using rgb correlations," *Multimedia Tools and Applications*, vol. 78, pp. 20 133–20 155, 2019.

[19] H.-J. Ko, C.-T. Huang, G. Horng, and W. Shiuh-Jeng, "Robust and blind image watermarking in dct domain using inter-block coefficient correlation," *Information Sciences*, vol. 517, pp. 128–147, 2020.

[20] Y. Zhang, Z. Wang, Y. Zhan, L. Meng, J. Sun, and W. Wan, "Jnd-aware robust image watermarking with tri-directional inter-block correlation," *International Journal of Intelligent Systems*, vol. 36, no. 12, pp. 7053–7079, 2021.

[21] D. S. Chauhan, A. K. Singh, A. Adarsh, B. Kumar, and J. P. Saini, "Combining mexican hat wavelet and spread spectrum for adaptive watermarking and its statistical detection using medical images," *Multimedia Tools and Applications*, vol. 78, pp. 12 647–12 661, 2019.

[22] T. Nguyen-Thanh and T. Le-Tien, "Study on improved cooperative spread

spectrum based robust blind image watermarking," *Journal of Advances in Information Technology*, vol. 11, no. 3, 2020.

[23] L. Novamizanti, A. B. Suksmono, D. Danudirdjo, and G. Budiman, "Robust reversible watermarking using stationary wavelet transform and multibit spread spectrum in medical images." *International Journal of Intelligent Engineering & Systems*, vol. 15, no. 3, 2022.

[24] R. Ye and H. Li, "A novel image scrambling and watermarking scheme based on cellular automata," in *2008 International Symposium on Electronic Commerce and Security*. IEEE, 2008, pp. 938–941.

[25] B. Yang, G. Lim, and J. Hur, "Toward practical deep blind watermarking for traitor tracing," *IEEE Access*, vol. 11, pp. 72 836–72 847, 2023.

[26] Z. Dai, C. Lian, Z. He, H. Jiang, and Y. Wang, "A novel hybrid reversible-zero watermarking scheme to protect medical image," *IEEE Access*, vol. 10, pp. 58 005–58 016, 2022.

[27] P. Pal, B. Jana, and J. Bhaumik, "Robust watermarking scheme for tamper detection and authentication exploiting ca," IET Image Processing, vol. 13, no. 12, pp. 2116–2129, 2019.

[28] A. M. Ramos, J. A. Artiles, D. P. Chaves, and C. Pimentel, "A fragile image watermarking scheme in dwt domain using chaotic sequences and error-correcting codes," *Entropy*, vol. 25, no. 3, p. 508, 2023.

[29] B. Zhu, X. Fan, T. Zhang, and X. Zhou, "Robust blind image watermarking using coefficient differences of medium frequency between inter-blocks," *Electronics*, vol. 12, no. 19, p. 4117, 2023.

[30] L. Laouamer and O. Tayan, "A semi-blind robust dct watermarking approach for sensitive text images," *Arabian Journal for Science and Engineering*, vol. 40, pp. 1097–1109, 2015.

[31] S. Roy and A. K. Pal, "A blind dct based color watermarking algorithm for embedding multiple watermarks," *AEU-International Journal of Electronics and Communications*, vol. 72, pp. 149–161, 2017.

[32] H.-T. Hu, S.-T. Wu, and T.-T. Lee, "Fft-based dual-mode blind watermarking for hiding binary logos and color images in audio," *IEEE Access*, vol. 11, pp. 37 612–37 622, 2023.

[33] S. Liu, Z. Pan, and H. Song, "Digital image watermarking method based on dct and fractal encoding," *IET image processing*, vol. 11, no. 10, pp. 815–821, 2017.

[34] S. P. Singh and G. Bhatnagar, "A new robust watermarking system in integer dct domain," *Journal of Visual Communication and Image Representation*, vol. 53, pp. 86–101, 2018.

[35] F. Ernawan and M. N. Kabir, "A robust image watermarking technique with an optimal dct-psychovisual threshold," *IEEE Access*, vol. 6, pp. 20 464–20 480, 2018.

[36] M. Jana and B. Jana, "A new dct based robust image watermarking scheme using cellular automata," *Information Security Journal: A Global Perspective*, vol. 31, no. 5, pp. 527–543, 2022.

[37] N. Pitsianis, P. Tsalides, G. Bleris, A. Thanailakis, and H. Card, "Deterministic one-dimensional cellular automata," *Journal of statistical physics*, vol. 56, pp. 99–112, 1989.

[38] S. Wolfram, "Universality and complexity in cellular automata," *Physica D: Nonlinear Phenomena*, vol. 10, no. 1-2, pp. 1–35, 1984.

[39] "Random number generation—wolfram language documentation," https://reference.wolfram.com/language/tutorial/RandomNumberGeneration.html, (Accessed on 10/19/2023).

[40] G. Cattaneo, M. Finelli, and L. Margara, "Investigating topological chaos by elementary cellular automata dynamics," *Theoretical computer science*, vol. 244, no. 1-2, pp. 219–241, 2000.

[41] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert et al., A statistical test suite for random and pseudorandom number generators for cryptographic applications. Gaithersburg, MD, USA: US Department of Commerce, Technology Administration, National Institute of Standards and Technology., 2001, vol. 22.

[42] D. Gage, E. Laub, and B. McGarry, "Cellular automata: is rule 30 random," in *Proceedings of the Midwest NKS Conference, Indiana University*, 2005.

[43] S.Wolfram-Writings, "r30img2.png (1240×642)," https://content.wolfram.com/sites/43/2020/07/r30img2.png, (Accessed on 11/10/2023).

[44] ——, "r30img7.png (702×366)," https://content.wolfram.com/sites/43/2020/07/r30img7.png, (Accessed on 11/10/2023).

[45] "The university of southern california. "sipi image database"," https://sipi.usc.edu/database/database.php, (Accessed on 10/20/2023).

[46] C.-H. L.-E. K.-J. Tseng, "Cil:54816, homo sapiens linnaeus, 1758, epithelial cell. cil. dataset. cil. dataset," http://www.cellimagelibrary.org/images/54816#cite, 2022, (Accessed on 10/20/2023).

[47] G. P. Don Fox, University of Houston, "The cell image library," http://www.cellimagelibrary.org/images/CCDB_54#cite, 2001, (Accessed on 10/20/2023).

[48] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle et al., "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, pp. 1045–1057, 2013.

[49] "University of california, san diego. the stare project," https://cecas.clemson.edu/~ahoover/stare/, (Accessed on 10/20/2023).

[50] U.S, "National archives nextgen catalog," https://catalog.archives.gov/, (Accessed on 11/09/2023).

[51] "National aeronautics and space administration (nasa), "nasa image and video library", images.nasa.gov," https://images.nasa.gov/, (Accessed on 11/09/2023).

[52] "Mathworks - makers of matlab and simulink - matlab & simulink, version 9.8.0.1323502 release 2020a; the math works, inc., feb 25,2020. computer software," https://in.mathworks.com/, (Accessed on 11/09/2023).

[53] N. F. Jódar, J. C. Boluda, R. G. Gironés, and V. H. Bosch, "Csdl & glider: Cad tools for hardware implementation of cellular automata," in *2008 International Conference on Advances in Electronics and Microelectronics*. IEEE, 2008, pp. 20–25.

[54] S. Dadgar and M. Neshat, "Comparative hybrid deep convolutional learning framework with transfer learning for diagnosis of lung cancer," in *International Conference on Soft Computing and Pattern Recognition*. Springer, 2022, pp. 296–305.

IRAM KHURSHID BHAT A dynamic and emerging scholar in the realm of Computer Sciences. Currently positioned as a Research Scholar in the P.G Department of Computer Science at the University of Kashmir, North Campus, India. A product of academic excellence at the Central University of Kashmir, Iram Khurshid Bhat earned a Master's Degree in Information and Technology in 2016. Following the completion of her master's degree, she contributed significantly as a contractual lecturer for two consecutive years, imparting her knowledge and expertise to eager minds. Since then, she has been ardently involved in cutting-edge research, focusing particularly on Cellular Automata, Image Processing, and Image Watermarking. She has presented her findings at prominent conferences. In the vibrant landscape of computer science, Iram Khurshid Bhat stands as a promising scholar, bringing fresh perspectives and innovation to the forefront of academic discourse.

FASEL QADIR Fasel Qadir is currently a Senior Assistant Professor in the Department of Computer Science at the University of Kashmir, North Campus, India. With an illustrious career marked by academic excellence and a passion for advancing knowledge, Dr. Fasel Qadir is a guiding force in both research and education. Completing his doctoral studies in Computer Science at the University of Kashmir in 2013, Dr. Qadir swiftly ascended the academic ranks, earning his current position as a Senior Assistant Professor. Dr. Qadir's research endeavors have left an indelible mark on the field. His work, published in reputable journals such as the Journal of Information Multimedia Tools and Applications, Security and Applications, and Advances in Applied Science Research, reflects his dedication to the field of image processing and cellular automata. He has been a consistent contributor to the academic discourse, presenting his findings at national and international conferences. A member of the Departmental Research Committee, Dr. Qadir actively engages with peers and scholars, contributing to the broader academic community.

MEHDI NESHAT received a PhD degree in Computer Science, from the University of Adelaide, Australia, in 2020. From 2020 to 2022, he was a Postdoctoral Research Associate in Data Science, Machine learning, and Deep learning with the University of South Australia. He has been a Senior Research Fellow at the Center for Artificial Intelligence Research and Optimisation, Torrens University Australia, since 2022. His primary research interests include Artificial Intelligence, Optimisation, and Machine/Deep learning. He has published more than 100 articles in top international conferences and journals with more than 2600 citations. In 2019 and 2020, he had received two Best Paper Awards from the most prestigious Genetic and Evolutionary Computation Conference (GECCO).

AMIR H. GANDOMI (Senior Member, IEEE) is currently a Professor of Data Science and an ARC DECRA Fellow with the Faculty of Engineering and Information Technology, University of Technology Sydney and Adjunct Professor in University Research and Innovation Center, Obuda University, Budapest, Hungary. Prior to joining UTS, he was an Assistant Professor at Stevens Institute of Technology, Hoboken, NJ, USA, and a Distinguished Research Fellow at BEACON Center, Michigan State University, East Lansing, MI, USA. He has published over 200 journal papers and seven books, which collectively have been cited over 47,000 times (H-index = 97). He has been named as one of the most influential scientific minds and a Highly Cited Researcher (top 1 year, from 2017 to 2023. He also ranked 18th in GP bibliography among more than 12,000 researchers. He is active in delivering keynotes and invited talks. His research interests are global optimization and (big) data analytics using machine learning and evolutionary computations in particular. He has served as an Associate Editor, Editor, and Guest Editor in several prestigious journals, such as AE of SWEVO, IEEE TBD, and IEEE IoTJ.