

A novel (n, n) multi-secret image sharing scheme harnessing RNA cryptography and 1-D group cellular automata

Yasmin Abdul, Venkatesan Ramasamy, Gaverchand Kukaram

Department of Mathematics, College of Engineering and Technology, SRM Institute of Science and Technology,
Kattankulathur, India

Article Info

Article history:

Received Aug 10, 2024

Revised Mar 14, 2025

Accepted Mar 26, 2025

Keywords:

Differential attacks

Entropy test

Group cellular automata

Image encryption

RNA cryptography

Statistical analysis

ABSTRACT

In the modern landscape, securing digital media is crucial, as digital images are increasingly disseminated through unsecured channels. Therefore, image encryption is widely employed, transforming visual data into an unreadable format to enhance image security and prevent unauthorized access. This paper proposes an efficient (n, n) multi-secret image sharing (MSIS) scheme that leverages ribonucleic acid (RNA) cryptography and one-dimensional (1-D) group cellular automata (GCA) rules. The (n, n) MSIS scheme encrypts n images into n distinct shares, necessitating all n shares for decryption to accurately reconstruct the original n images. Initially, a key image is generated using RNA cryptography, harnessing the extensive sequence variability and inherent complexity of RNA. This secret key is then used to encrypt n images in the primary phase. In the secondary phase, pixel values are transformed through multiple processes, with randomness achieved by executing a key function derived from GCA, known for its reversible properties, computational efficiency, and robustness against cryptographic attacks. The proposed model, implemented in Python, is validated through experimental results, demonstrating its effectiveness in resisting a broad spectrum of attacks, including statistical, entropy, differential, and pixel parity analyses. These findings affirm the model's durability, security, and resilience, underscoring its superior performance compared to existing models.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Venkatesan Ramasamy

Department of Mathematics, College of Engineering and Technology

SRM Institute of Science and Technology

Kattankulathur 603203, Tamil Nadu, India

Email: venkater1@srmist.edu.in

1. INTRODUCTION

In the modern digital landscape, the secure transfer of data has become crucial, with numerous internet applications facilitating confidential communication. Consequently, safeguarding information against unauthorized access has emerged as a critical objective. The rapid advancement of computer and internet technologies has underscored information security as a perpetual concern. To address this, various methods such as cryptography [1], [2], steganography [3], [4], and watermarking [5], [6] have been proposed and extensively studied. However, with the proliferation of digital images, including medical, grayscale, color, and binary images, protecting this format of information has become paramount. Digital images are extensively utilized in several domains like space research, medical research, telemedicine, industrial processes, defense sensors, and others, often containing sensitive and private information. These images are crucial for scientific and technological advancements as well as for ensuring the privacy and confidentiality

of personal and proprietary data. Therefore, it is imperative to develop robust image encryption techniques to safeguard against manipulation or unauthorized access by third parties, ensuring the integrity and confidentiality of these critical data assets. Researchers are actively advancing the use of cellular automata (CA) and ribonucleic acid (RNA) in image encryption, leveraging their capabilities to generate complex, pseudo-random patterns and harnessing RNA's vast sequence diversity and intrinsic complexity to enhance visual cryptographic security.

CA developed by Ulam and von Neumann and later refined by Stephen Wolfram in 1986 [6], have demonstrated their efficacy in image encryption through two primary methods such as generating pseudo-random numbers and bit-level encryption. Techniques such as position permutation and value transformation, including confusion and diffusion, allow CA to reposition pixels and alter their values without changing the initial entropy, resulting in robust encryption. The extensive rule set and minimal hardware requirements of CA render key discovery computationally impractical for attackers, ensuring secure data transmission, image cryptography, processing, and authentication with low complexity in both hardware and software implementations.

RNA based encryption schemes leverage the distinctive characteristics of RNA molecules to store data within nucleotide sequences, which are then translated into pixel sequences in images. This approach can potentially overcome traditional encryption vulnerabilities, including quantum computing attacks and brute-force attempts, due to the vast number of possible RNA sequences [7]. Although still in early development, current methods combining RNA and DNA molecules, along with non-coding RNA patterns, require further research and rigorous testing [8].

Wu *et al.* [9] proposed an encryption technique utilizing RNA molecules and genetic codes to enhance security and scalability. A study in [10] compared RNA based encryption systems based on encryption time, decryption time, and image quality. Gilbert *et al.* [11] investigated a DNA and RNA motif-based method for key generation, emphasizing security and computational efficiency. Abbasi *et al.* [12] developed the erratic amino acid technique, combining SHA-256 key generation, pixel permutation, diffusion, and optimization, but the process remained time-consuming despite its high resistance to attacks.

Alvarez *et al.* [13] developed an image encryption technique using second-order CA, where the subsequent state of a cell was based on its prior state and adjacent cells, doubling the encrypted image size. Maleki *et al.* [14] extended this to higher-order CA, but image quality was compromised as the least significant bit-plane couldn't be recovered. Li *et al.* [15] proposed combining a chaotic map with CA for better security, though improvements were needed in noise attack resistance and overall effectiveness.

Secret image sharing was first pioneered using the Shamir–Lagrange technique [16], followed by advancements in single-secret image sharing [17]–[19]. To enhance efficiency for multiple secrets, methods utilizing additional storage were introduced [20], alongside a theoretical framework for sharing two secret images [21]. Subsequent innovations included the dual-ring multi-secret image sharing (MSIS) scheme [22] and an (n, n) MSIS method leveraging Boolean operations for encoding multiple shares [23]. Visual cryptography, originally devised for single-image encryption, evolved to accommodate multiple images, with decryption requiring all shares [24]. The MSIS scheme found diverse applications, including missile launch codes, safety deposit boxes, access control, and e-voting or e-auctions.

The earlier analysis highlighted the pros and cons of CA and RNA in image encryption, with existing methods facing challenges like susceptibility to statistical attacks, high pixel correlation, low entropy, and limited resistance to differential attacks. Considering these issues and the broad applications of MSIS, this paper introduces an (n, n) MSIS scheme that integrates GCA and RNA to effectively tackle these challenges and provide a robust solution. RNA codons enhance security through their extensive sequence variability and inherent complexity during key image generation. Furthermore, the deployment of 1-D GCA rules ensures randomness and robust image encryption by iterating image pixels, thereby enhancing resilience against cryptographic attacks due to their simplicity and effectiveness in computing.

The remaining sections of the paper are organized in the following manner: section 2 delivers an in-depth exploration of RNA cryptography and CA. Section 3 elucidates the proposed image encryption technique, including the relevant algorithms. Section 4 thoroughly examines the observations and findings. Section 5 concludes the concise summary of the paper.

2. FUNDAMENTALS OF CA AND RNA CRYPTOGRAPHY

2.1. RNA cryptography

RNA cryptography harnesses the distinctive traits of RNA sequences to enhance data security. An RNA sequence comprises four nucleic acid bases, adenine (A), guanine (G), cytosine (C), and uracil (U). Adenine forms a base pair with uracil (A-U), and guanine forms a base pair with cytosine (G-C), mirroring the complementary nature of binary digits (0 and 1). Using the four RNA bases to encode binary pairs (00, 11, 01, 10) results in 24 possible coding schemes. Despite this, merely eight of these schemes adhering to the

Watson-Crick complementarity criteria, as depicted in Table 1. The RNA coding table converts the pixel values of images into RNA sequences. The RNA dictionary, presented in Table 2, translates these RNA sequences into decimal values. In the proposed scheme, the RNA encoding table and dictionary are utilized to generate the key image from the original images, ensuring a secure and robust encryption process.

Table 1. RNA encoding table

	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
B	11	11	10	10	01	01	00	00
C	01	10	00	11	00	11	01	10
D	10	01	11	00	11	00	10	01

Table 2. Dictionary for translating RNA sequences to decimal form

Dec.	DNA	Dec.	DNA	Dec.	DNA	Dec.	DNA	Dec.	DNA	Dec.	DNA
0	AAAA	43	AGGC	86	UUUG	129	GAAU	172	GGCA	215	CUUC
1	AAAU	44	AGCA	87	UUUC	130	GAAG	173	GGCU	216	CUGA
2	AAAG	45	AGCU	88	UUGA	131	GAAC	174	GGCG	217	CUGU
3	AAAC	46	AGCG	89	UUGU	132	GAUA	175	GGCC	218	CUGG
4	AAUA	47	AGCC	90	UUGG	133	GAUU	176	GCAA	219	CUGC
5	AAUU	48	ACAA	91	UUGC	134	GAUG	177	GCAU	220	CUCA
6	AAUG	49	ACAU	92	UUCA	135	GAUC	178	GGAG	221	CUCU
7	AAUC	50	ACAG	93	UUCU	136	GAGA	179	GCAC	222	CUCG
8	AAGA	51	ACAC	94	UUCG	137	GAGU	180	GCUA	223	CGCC
9	AAGU	52	ACUA	95	UGCC	138	GAGG	181	GCUU	224	CGAA
10	AAGG	53	ACUU	96	UGAA	139	GAGC	182	GCUG	225	CGAU
11	AAGC	54	ACUG	97	UGAU	140	GACA	183	GCUC	226	CGAG
12	AACA	55	ACUC	98	UGAG	141	GACU	184	GCGA	227	CGAC
13	AACU	56	ACGA	99	UGAC	142	GACG	185	GCGU	228	CGUA
14	AACG	57	ACGU	100	UGUA	143	GACC	186	GCGG	229	CGUU
15	AACC	58	ACGG	101	UGUU	144	GUAA	187	GCGC	230	CGUG
16	AUAA	59	ACGA	102	UGUG	145	GUAU	188	GCCA	231	CGUC
17	AUAU	60	ACCA	103	UGUC	146	GUAG	189	GCCU	232	CGGA
18	AUAG	61	ACCU	104	UGGA	147	GUAC	190	GCCG	233	CGGU
19	AUAC	62	ACCG	105	UGGU	148	GUUA	191	GCCC	234	CGGG
20	AUUA	63	ACCC	106	UGGG	149	GUUU	192	CAAA	235	CGGC
21	AUUU	64	UAAA	107	UGGC	150	GUUG	193	CAAU	236	CGCA
22	AUUG	65	UAAU	108	UGCA	151	GUUC	194	CAAG	237	CGCU
23	AUUC	66	UAAG	109	UGCU	152	GUGA	195	CAAC	238	CGCG
24	AUGA	67	UAAC	110	UGCG	153	GUGU	196	CAUA	239	CGCC
25	AUGU	68	UAUA	111	UGCC	154	GUGG	197	CAUU	240	CCAA
26	AUGG	69	UAUU	112	UCAA	155	GUGC	198	CAUG	241	CCAU
27	AUGC	70	UAUG	113	UCAU	156	GUCA	199	CAUC	242	CCAG
28	AUCA	71	UAUC	114	UCAG	157	GUCU	200	CAGA	243	CCAC
29	AUCU	72	UAGA	115	UCAC	158	GUCG	201	CAGU	244	CCUA
30	AUCG	73	UAGU	116	UCUA	159	GUCC	202	CAGG	245	CCUU
31	AGCC	74	UAGG	117	UCUU	160	GGAA	203	CAGC	246	CCUG
32	AGAA	75	UAGC	118	UCUG	161	GGAU	204	CACA	247	CCUC
33	AGAU	76	UACA	119	UCUC	162	GGAG	205	CACU	248	CCGA
34	AGAG	77	UACU	120	UCGA	163	GGAC	206	CACG	249	CCGU
35	AGAC	78	UACG	121	UCGU	164	GGUA	207	CACC	250	CCGG
36	AGUA	79	UACC	122	UCGG	165	GGUU	208	CUAA	251	CCGC
37	AGUU	80	UUAU	123	UCGC	166	GGUG	209	CUAU	252	CCCA
38	AGUG	81	UUAA	124	UCCA	167	GGUC	210	CUAG	253	CCCU
39	AGUC	82	UUAG	125	UCCU	168	GGGA	211	CUAC	254	CCCG
40	AGGA	83	UUAC	126	UCCG	190	GGGU	212	CUUA	255	CCCC
41	AGGU	84	UUUA	127	UCCC	170	GGGG	213	CUUU		
42	AGGG	85	UUUU	128	GAAA	171	GGGC	214	CUUG		

2.2. Cellular automata

CA are discrete mathematical systems where time, states, and space are all quantized. Cells are systematically arranged in a finite, regular lattice structure. CA can be rigorously defined by the five-tuple (L, Q, N, δ, I) where L denotes the lattice of a regular grid, Q is the finite set of states, N represents the set of neighbors, δ is the transition state function and I signifies the initial state. Elementary CA also called 1-D CA, consist of a linear sequence of cells. Each cell modifies its state by a local transition rule that depends on the present state and the states of its neighboring cells. The state of each cell can only be either 0 or 1. Thus, there are $2 \times 2 \times 2 = 2^3 = 8$ possible neighbourhood configurations: 111, 011, 101, 110, 001, 010,

100,000 resulting in $2^8 = 256$ possible rules for 1-D CA. The transition matrix of a 1-D CA, symbolized by δ , is an $n \times n$ matrix encapsulating the local update rules for all cells. The k^{th} row corresponds to the neighborhood relations of the k^{th} cell. If $\delta(i, j)$ is 1, it indicates that the subsequent state of the i^{th} cell depends upon the current state of the j^{th} cell. Contrarily, the value is 0. State transitions are mathematically represented as:

$$[S_{t+1}(x)] = [\delta] \times [S_t(x)]$$

where $S_{t+1}(x)$ represent the state of cell x at the next time step $t + 1$. At the current time step t , the state of cell x is denoted by $S_t(x)$, while $S_t(x - 1)$ and $S_t(x + 1)$ indicate the states of the left and right neighbouring cells, respectively.

CA is categorized as GCA if the determinant of δ is 1, contrary, it is categorized as non-group CA. In a GCA, applying a specific rule or rule vector to the cells regenerates their initial state after a specific number of iterations. This number is referred as the order of the CA and can be mathematically expressed as:

$$[\delta]^n = I \Rightarrow [S_{t+n}(x)] = I \times [S_t(x)]$$

where n signifies the order of the group and I represents the identity matrix. Examples of GCAs include rules 90, 102, 105 and 204. Their complements are rules 165, 153, 150, and 51, respectively. According to [25], the complement of a CA rule that forms a group is also a GCA. Therefore, all the various logical operators outlined in Table 3 are identified as GCA. Table 4 delineates the next state of these rules under different neighborhood configurations.

In the proposed model, the pixel values of an image are transmuted by a key function through the unique characteristics of GCA rules, iterating through a half-cycle during the encryption phase. The subsequent half-cycle iteration is executed during the decryption phase, thereby restoring the original pixel values. This methodology ensures a robust and efficient cryptographic system by capitalizing on the inherent reversibility of GCA rules.

Table 3. Logical expression of GCA rules

No	Rule	Logical operations
1	51	$S_{t+1}(x) = S_t(x)$
2	90	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x + 1)$
3	102	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x)$
4	105	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x) \oplus S_t(x + 1)$
5	150	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x) \oplus S_t(x + 1)$
6	153	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x)$
7	165	$S_{t+1}(x) = S_t(x - 1) \oplus S_t(x + 1)$
8	204	$S_{t+1}(x) = S_t(x)$

Table 4. Illustration of GCA rules for next state

Rule	111	011	101	110	001	010	100	000
51	0	0	1	0	1	0	1	1
90	1	1	0	1	1	0	1	0
102	0	0	1	1	1	1	0	0
105	0	1	1	1	0	0	0	1
150	1	0	0	0	1	1	1	0
153	1	1	0	0	0	0	1	1
165	1	0	1	0	0	1	0	1
204	1	1	0	1	0	1	0	0

3. PROPOSED METHOD

This section delineates an innovative approach to (n, n) MSIS, utilizing RNA and GCA to significantly enhance the encryption and transmission of multiple images simultaneously. In the proposed model, all n original images are of uniform size with dimension $m \times m$ and comprise four primary components: original images, a key image, a key function, and encrypted images. The key image, generated from the n original images via an RNA encoding table and dictionary, is used to encrypt the original images in the initial phase. To further enhance security and randomness, the key function, derived from GCA rules, is employed to iterate the image pixels during the culmination of the encryption process. This dual-layer encryption scheme ensures heightened security and yields n encrypted images corresponding to the n original images.

3.1. Key generation process

Generating the key image from n original images involve the deployment of an RNA encoding table and RNA dictionary. Initially, the n original images are inputted, and their red, green, and blue (RGB) components are extracted and converted into binary matrices. Sequential Boolean XOR operations are performed on the R, G, and B components, yielding consolidated RGB values. These values are then encoded into RNA codons according to rules determined by the number of images using Table 1. The RNA codons are subsequently translated into decimal values and then into binary values using Table 2. Finally, these binary values are interpreted as RGB components and reconstructed into an image, serving as the key image for enhancing security in the encryption process. The workflow of key generation is depicted in Figure 1, while the detailed key generation process is presented in Algorithm 1.

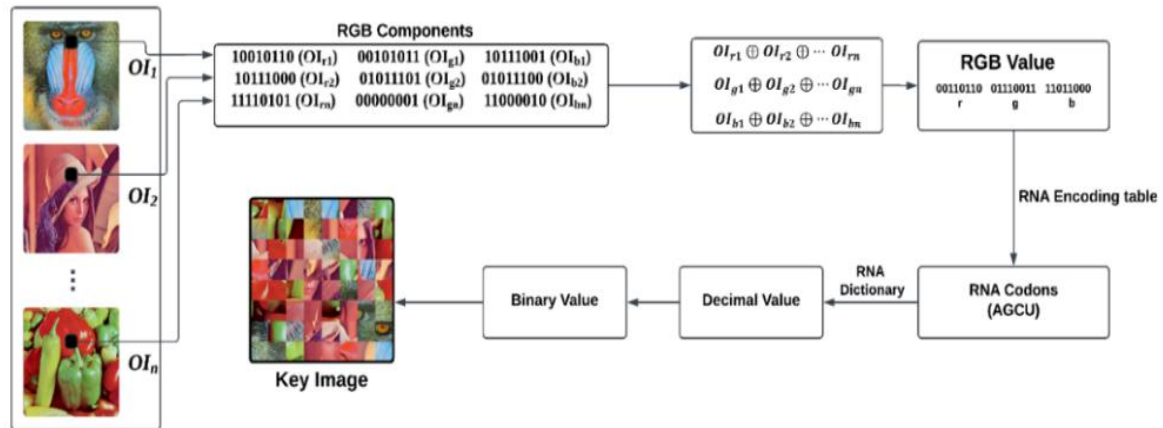


Figure 1. Workflow of key image generation process

Algorithm 1. Key image formation

Input: Original Images (OI_1, OI_2, \dots, OI_n)

Output: Key Image KI

1. Consider OI_1, OI_2, \dots, OI_n as input
2. Extract the RGB components and transform them into matrices
 - $OI_1 = OI_{r1}, OI_{g1}, OI_{b1}$
 - $OI_2 = OI_{r2}, OI_{g2}, OI_{b2}$
 - $OI_n = OI_{rn}, OI_{gn}, OI_{bn}$
3. Perform XOR on the R, G, and B components of all images to yield the consolidated RGB matrix $OI' = [OI_r, OI_g, OI_b]$
 - $OI_r = OI_{r1} \oplus OI_{r2} \oplus \dots \oplus OI_{rn}$
 - $OI_g = OI_{g1} \oplus OI_{g2} \oplus \dots \oplus OI_{gn}$
 - $OI_b = OI_{b1} \oplus OI_{b2} \oplus \dots \oplus OI_{bn}$
4. Convert OI' to OI'' to encode binary values into RNA sequence using RNA rules based on image count
 - $OI'' = \text{RNA Codes } [OI']$
 - RNA Rule = $n \bmod 8$, where n = number of images
5. Convert the RNA sequence OI'' into decimal values using an RNA dictionary to generate key matrix
 - $KM = \text{Decimal values } [OI'']$
6. Transform KM into its equivalent binary values
 - $KM' = \text{Bin}[KM]$
7. The values in KM' act as RGB components, constructing key image KI

3.2. Encryption and decryption process

In the proposed image encryption technique, a series of sophisticated transformations are applied to secure n original images OI_1, OI_2, \dots, OI_n , each with dimensions $m \times m$. Initially, each original image undergoes an XOR operation with a key image of the same size to diffuse the pixel values, generating primary encrypted shares $OI'_1, OI'_2, \dots, OI'_n$. Subsequently, these primary shares are converted into 2-D matrices M_1, M_2, \dots, M_n . Following this step, the rows and columns of these matrices are shuffled to disrupt spatial correlations, producing permuted matrices M'_1, M'_2, \dots, M'_n . The process continues with further modification of pixel values through arithmetic operations, leading to matrices $M''_1, M''_2, \dots, M''_n$.

Subsequently, a key function processes the pixel values according to the GCA rules, determined by n , systematically iterating the final 2-D matrices over half of the cycle length. This iterative process significantly increases randomness, resulting in matrices $M_1'', M_2'', \dots, M_n''$. These final transformed matrices are then converted back into image form, yielding the encrypted images EI_1, EI_2, \dots, EI_n , which correspond to the original images. This encryption methodology robustly obscures the pixel values by employing advanced techniques such as diffusion, permutation, transformation, and iterative processes governed by the GCA rules. Conversely, implementing the reverse technique allows for the precise restoration of the original pixels, thereby regenerating the decrypted images DI_1, DI_2, \dots, DI_n , ensuring the decryption process accurately reconstructs the original images. The workflow of the proposed technique is meticulously elucidated in Figure 2, while the encryption and decryption procedures are described in Algorithm 2 and Algorithm 3, respectively.

Algorithm 2. Encryption procedure

Input: Original Images (OI_1, OI_2, \dots, OI_n)

Output: Encrypted Images (EI_1, EI_2, \dots, EI_n)

1. Read OI_1, OI_2, \dots, OI_n as input
2. Perform XOR operation between n original images and key image
 - $OI'_1 = OI_1 \oplus KI; OI'_2 = OI_2 \oplus KI \dots OI'_n = OI_n \oplus KI$
3. Convert $OI'_1, OI'_2, \dots, OI'_n$ into 2-D matrices M_1, M_2, \dots, M_n
4. Permute M_1, M_2, \dots, M_n to M'_1, M'_2, \dots, M'_n
 - For each M'_i ($i = 1$ to n) interchange rows \leftrightarrow columns
5. Modify the pixel value of matrix M'_i to M''_i
 - $M''_i = ((h \times M'_i) \bmod 256)$ times, where h is an integer
6. Apply key function to iterate each matrix M''_i to obtain the final encrypted matrix M'''_i
 - $M'''_i = KF(M''_i)$
7. Transform matrix M'''_i ($i = 1$ to n) into images and store them as encrypted images EI_1, EI_2, \dots, EI_n

Algorithm 3. Decryption procedure

Input: Encrypted Images (EI_1, EI_2, \dots, EI_n)

Output: Original Images (DI_1, DI_2, \dots, DI_n)

1. Input the encrypted images EI_1, EI_2, \dots, EI_n
2. Convert each encrypted image E_i into its corresponding 2-D matrix form M'''_i for $i = 1$ to n
3. Apply key function on M'''_i , where M'_i denotes the decrypted 2-D matrix
 - $M'_i = KF(M'''_i)$
4. Transform each matrix M'_i to obtain M'_i for modifying the pixel values
 - $M'_i = ((h' \times M'_i) \bmod 256)$, where h' is a multiplicative integer of $h \forall i = 1$ to n
5. Revert the permutation of M'_i to derive matrices M_i ($i = 1$ to n)
 - $M_i = \text{interchange rows} \leftrightarrow \text{columns} (M'_i)$
6. Perform the XOR operation amid KI and M'_i to retrieve the original pixel values
 - Calculate $OI_i = M_i \oplus KI$ for each matrix $M_i, \forall i = 1$ to n
7. Obtain decrypted images DI_1, DI_2, \dots, DI_n from restored pixel values OI_1, OI_2, \dots, OI_n

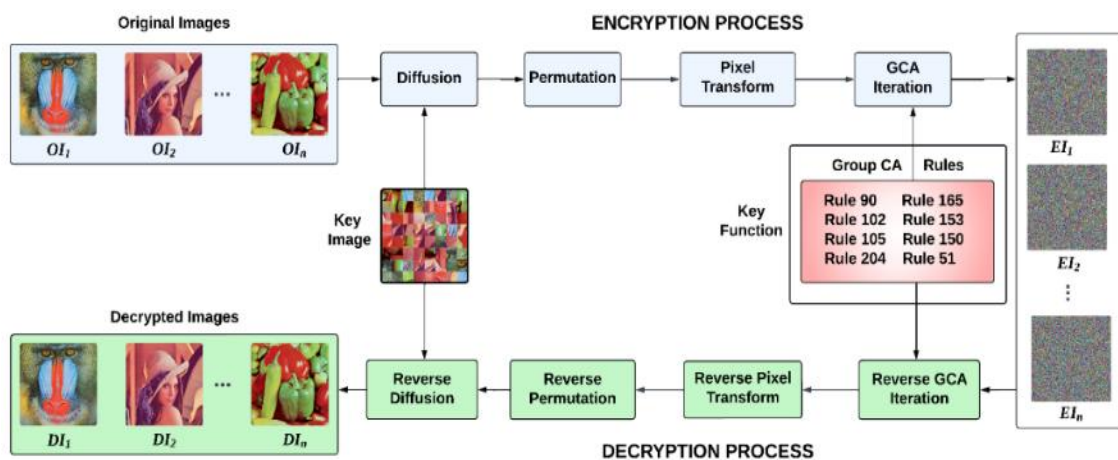


Figure 2. Flowchart of the proposed (n, n) MSIS scheme

3.3. Formation of key function

The key function is invoked during both the encryption and decryption processes, utilizing a 2-D matrix as input. Each RGB value is treated as a block and is encrypted or decrypted using a GCA rule vector. The rules are selected based on $n \bmod 8$, where n be the number of images. This key function significantly enhances randomness by iterating over the matrix, thereby fortifying confidentiality, ensuring data integrity, and bolstering defense against security vulnerabilities. The key function generation is shown in Algorithm 4.

Algorithm 4. Generation of key function

Input: 2-D matrix, GCA rule vector

Output: Iterated matrix (encryption/ decryption)

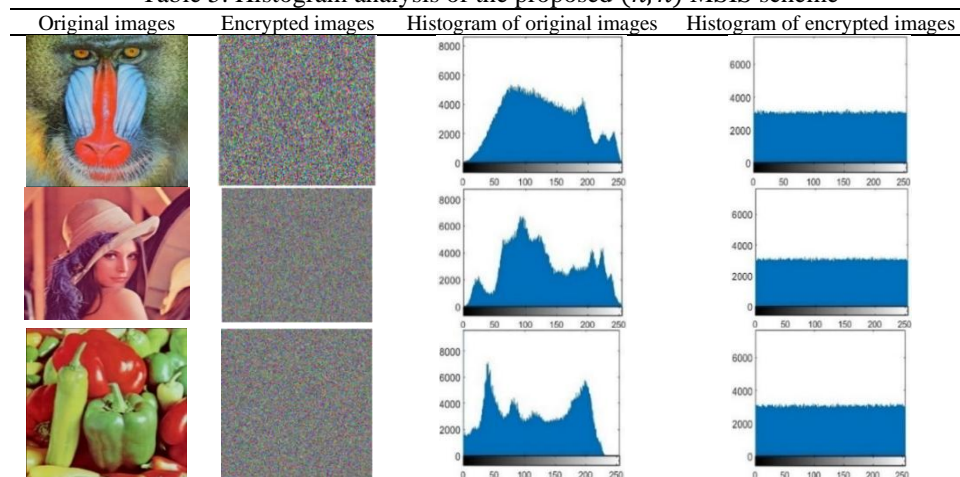
1. Load a block from the 2-D matrices into an array of length 8
2. Initialize the array with a designated GCA rule set
3. Iterate the array using a specific GCA rule
4. The final output array signifies the encrypted or decrypted value of the original block
5. Execute Steps 1 through 4 for each RGB value to encrypt or decrypt the entire matrix.

4. RESULTS AND DISCUSSION

The current section comprises a series of rigorous experiments designed to assess and validate the effectiveness and superiority of the proposed (n, n) MSIS model. RGB images, specifically a) Baboon, b) Lena, and c) Pepper, each sized 512×512 , were sourced from the University of Waterloo Image Repository for standard analysis and comparison [26]. These experiments were meticulously conducted on an HP laptop equipped with a 12th-generation Intel Core i5 processor and a 512 GB SSD. Python software was utilized to run the analysis, and the results will be presented in the following sections.

- 1) Statistical analysis: the evaluation of security in encrypted images is achieved through histogram and correlation analysis. Histogram analysis assesses pixel intensity distributions for uniformity, while the correlation coefficient measures the resemblance between neighboring pixels (horizontal, vertical, and diagonal), aiming for values close to zero. Table 5 displays smoother, flatter histogram distributions for the encrypted images, and Table 6 reveals a negligible association between the original and encrypted images, confirming the proposed model's resistance to statistical attacks.

Table 5. Histogram analysis of the proposed (n, n) MSIS scheme



- 2) Information entropy analysis: entropy quantifies the unpredictability and randomness of data, with values close to 8 indicating a uniform pixel value distribution. This signifies a robust cipher that is less susceptible to security breaches.
- 3) Differential attack analysis: differential attacks assert that even minor modifications to the pixels of an original image should result in significant changes in the corresponding cipher image. This can be evaluated using the number of pixels change rate (NPCR) and the unified average changing intensity (UACI), which provide both quantitative and qualitative assessments of the cipher images. Generally, higher NPCR and UACI scores signify greater resistance to these attacks.
- 4) Pixel disparity analysis: pixel disparity evaluates the relationship amid plain and cipher images. The peak signal-to-noise ratio (PSNR) and mean square error (MSE) are widely recognized methods used to

gauge this relationship. The PSNR value reflects image distortion, where a lower PSNR and greater MSE signify better encryption effectiveness during the assessments.

The results for tests 2 to 4, encompassing entropy, NPCR, UACI, MSE, and PSNR for the proposed model, have been rigorously compared with existing image encryption techniques. Table 7 demonstrates that the proposed model surpasses all others across these metrics and exhibits exceptionally high entropy, nearing 7.9986, indicative of strong encryption randomness. Additionally, it achieves an average UACI of 34.3037 and an NPCR of 99.9236, reflecting robust resistance to differential attacks. Furthermore, the model achieves an average MSE of 74.5233 and PSNR of 24.6033, signifying commendable image quality. Notably, the proposed MSIS scheme not only demonstrates superior results but also effectively encrypts multiple images simultaneously, significantly reducing computational complexity while maintaining high efficiency.

Table 6. Correlation analysis of the proposed model amid adjacent pixels

Image	Original images			Encrypted images		
	H	V	D	H	V	D
Baboon	0.7115	0.8511	0.6839	-0.0037	0.0032	0.0004
Lena	0.9850	0.9782	0.9633	0.0065	0.0052	-0.0003
Pepper	0.9835	0.9715	0.9618	0.0077	0.0068	-0.0053

Table 7. Comparative analysis of entropy, NPCR, UACI, PSNR and MSE

Test	Images	Proposed	Ref [27]	Ref [28]	Ref [29]	Ref [30]
Entropy	Baboon	7.9985	7.9975	7.9981	7.6026	7.9856
	Lena	7.9988	7.9982	7.9983	7.8232	7.9903
	Pepper	7.9987	7.9972	7.9985	7.7986	7.9464
NPCR	Baboon	99.8735	99.6124	99.6037	99.6361	95.9788
	Lena	99.9983	99.6103	99.5961	99.6203	96.6937
	Pepper	99.8991	99.6519	99.6203	98.9261	95.0443
UACI	Baboon	34.4347	33.3973	33.5050	32.7937	32.6018
	Lena	33.9127	33.4891	33.5002	31.9796	33.8962
	Pepper	34.5637	33.4567	33.4962	33.3715	34.1950
PSNR	Baboon	30.51	23.87	27.46	28.40	29.43
	Lena	29.43	24.96	25.65	28.02	25.59
	Pepper	30.74	26.59	27.03	29.13	28.54
MSE	Baboon	89.87	88.03	86.92	96.25	91.24
	Lena	83.76	82.71	79.35	88.51	83.65
	Pepper	91.43	84.63	82.78	90.43	87.39

5. CONCLUSION

Through the results obtained from a series of experiments on various images, it is evident that the proposed method exhibits superior efficiency and effectiveness compared to existing techniques, ensuring high entropy, robust resistance to differential attacks, and commendable image quality. Utilizing RNA cryptography for key image generation establishes a secure encryption process. Moreover, the inherent properties of GCA rules enhance computational efficiency by executing the key function through half-cycle iterations during encryption and completing the remaining iterations during decryption. Future developments could extend this model to higher-dimensional data structures, integrate parallel processing techniques for large data management, and explore quantum cryptography to significantly enhance security.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Yasmin Abdul	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Venkatesan Ramasamy		✓			✓	✓		✓	✓	✓	✓	✓	✓	
Gaverchand Kukaram	✓	✓	✓	✓	✓		✓			✓	✓		✓	✓

C : C onceptualization	I : I nterpretation	Vi : V isualization
M : M ethodology	R : R esources	Su : S upervision
So : S oftware	D : D ata Curation	P : P roject administration
Va : V alidation	O : Writing - O riginal Draft	Fu : F unding acquisition
Fo : F ormal analysis	E : Writing - Review & E ditng	

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, upon reasonable request.




REFERENCES

- [1] William, "Cryptography and network security - principles and practice, 7th edition," *Google Books*. <https://books.google.co.in/books?id=AhDCDwAAQBAJ>
- [2] G. Kukaram and V. Ramasamy, "A novel approach of 1-D cellular automata in cryptosystem," *Mathematical Modelling and Engineering Problems*, vol. 10, no. 6, pp. 2121–2126, Dec. 2023, doi: 10.18280/mmep.100623.
- [3] W. M. Abdullallah and A.M.S Rahma, "A review on steganography techniques," *American Scientific Research Journal for Engineering, Technology, and Sciences*, vol. 24, no. 1, pp. 131–150, 2016.
- [4] H. K. Tayyeh and A. S. A. Al-Jumaili, "A combination of least significant bit and deflate compression for image steganography," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 1, pp. 358–364, Nov. 2021, doi: 10.11591/ijece.v12i1.pp358-364.
- [5] M. Begum and M. S. Uddin, "Digital image watermarking techniques: a review," *Information*, vol. 11, no. 2, p. 110, Feb. 2020, doi: 10.3390/info11020110.
- [6] S. Wolfram, "Random sequence generation by cellular automata," *Advances in applied mathematics*, vol. 7, no. 2, pp. 123–169, 1986, doi: 10.1016/0196-8858(86)90028-X.
- [7] T. M. Fernandez-Carames and P. Fraga-Lamas, "Towards post-quantum blockchain: a review on blockchain cryptography resistant to quantum computing attacks," *IEEE Access*, vol. 8, pp. 21091–21116, Jan. 2020, doi: 10.1109/access.2020.2968985.
- [8] N. Reynolds, A. Diamantopoulos, and B. Schlegelmilch, "Pre-testing in questionnaire design: a review of the literature and suggestions for further research," *Market Research Society Journal*, vol. 35, no. 2, pp. 1–11, Mar. 1993, doi: 10.1177/147078539303500202.
- [9] C. Wu, K.-Y. Hu, Y. Wang, J. Wang, and Q.-H. Wang, "Scalable asymmetric image encryption based on phase-truncation in cylindrical diffraction domain," *Optics Communications*, vol. 448, pp. 26–32, May 2019, doi: 10.1016/j.optcom.2019.05.009.
- [10] B. S. A. Alhayani *et al.*, "Optimized video internet of things using elliptic curve cryptography based encryption and decryption," *Computers & Electrical Engineering*, vol. 101, p. 108022, Apr. 2022, doi: 10.1016/j.compeleceng.2022.108022.
- [11] H. Gilbert, M. J. B. Robshaw, and Y. Seurin, "Increasing the security and efficiency of," in *Springer eBooks*, 2008, pp. 361–378. doi: 10.1007/978-3-540-78967-3_21.
- [12] A. A. Abbasi, M. Mazinani, and R. Hosseini, "Chaotic evolutionary-based image encryption using RNA codons and amino acid truth table," *Optics & Laser Technology*, vol. 132, p. 106465, Aug. 2020, doi: 10.1016/j.optlastec.2020.106465.
- [13] G. Alvarez, L. H. Encinas, and A. M. Del Rey, "A multiset secret sharing scheme for color images based on cellular automata," *Information Sciences*, vol. 178, no. 22, pp. 4382–4395, Jul. 2008, doi: 10.1016/j.ins.2008.07.010.
- [14] F. Maleki, A. Mohades, S. M. Hashemi, and M. E. Shirri, "An image encryption system by cellular automata with memory," in *2008 Third International Conference on Availability, Reliability and Security*, 2008, pp. 1266–1271. doi: 10.1109/ares.2008.121.
- [15] L. Li, Y. Luo, S. Qiu, X. Ouyang, L. Cao, and S. Tang, "Image encryption using chaotic map and cellular automata," *Multimedia Tools and Applications*, vol. 81, no. 28, pp. 40755–40773, May 2022, doi: 10.1007/s11042-022-12621-9.
- [16] P.-Y. Lin, J.-S. Lee, and C.-C. Chang, "Distortion-free secret image sharing mechanism using modulus operator," *Pattern Recognition*, vol. 42, no. 5, pp. 886–895, Oct. 2008, doi: 10.1016/j.patcog.2008.09.014.
- [17] C. Guo, C.-C. Chang, and C. Qin, "A hierarchical threshold secret image sharing," *Pattern Recognition Letters*, vol. 33, no. 1, pp. 83–91, Oct. 2011, doi: 10.1016/j.patrec.2011.09.030.
- [18] X. Wu, D. Ou, Q. Liang, and W. Sun, "A user-friendly secret image sharing scheme with reversible steganography based on cellular automata," *Journal of Systems and Software*, vol. 85, no. 8, pp. 1852–1863, Mar. 2012, doi: 10.1016/j.jss.2012.02.046.
- [19] M. Ulutas, G. Ulutas, and V. V. Nabiyeu, "Invertible secret image sharing for gray level and dithered cover images," *Journal of Systems and Software*, vol. 86, no. 2, pp. 485–500, Sep. 2012, doi: 10.1016/j.jss.2012.09.027.
- [20] C.-C. Chen and Y.-W. Chien, "Sharing numerous images secretly with reduced possessing load," *Fundamenta Informaticae*, vol. 86, no. 4, pp. 447–458, 2008, doi: 10.3233/FUN-2008-86405.
- [21] S.-J. Lin, S.-K. Chen, and J.-C. Lin, "Flip visual cryptography (FVC) with perfect security, conditionally-optimal contrast, and no expansion," *Journal of Visual Communication and Image Representation*, vol. 21, no. 8, pp. 900–916, Aug. 2010, doi: 10.1016/j.jvcir.2010.08.006.
- [22] H. Hsu, J. Chen, T. Chen, and Y. Lin, "Special type of circular visual cryptography for multiple secret hiding," *The Imaging Science Journal*, vol. 55, no. 3, pp. 175–179, Sep. 2007, doi: 10.1179/174313107x176289.
- [23] T.-H. Chen and C.-S. Wu, "Efficient multi-secret image sharing based on Boolean operations," *Signal Processing*, vol. 91, no. 1, pp. 90–97, Jun. 2010, doi: 10.1016/j.sigpro.2010.06.012.




- [24] M. Naor and A. Shamir, "Visual cryptography," in *Lecture notes in computer science*, 1995, pp. 1–12. doi: 10.1007/bfb0053419.
- [25] N. Pries, N. Thanailakis, and N. Card, "Group properties of cellular automata and VLSI applications," *IEEE Transactions on Computers*, vol. C-35, no. 12, pp. 1013–1024, Dec. 1986, doi: 10.1109/tc.1986.1676709.
- [26] "Image repository," *University of Waterloo*. <https://links.uwaterloo.ca/Repository.html> (accessed Jul. 15, 2024).
- [27] X. Wang and N. Guan, "A novel chaotic image encryption algorithm based on extended Zigzag confusion and RNA operation," *Optics & Laser Technology*, vol. 131, p. 106366, Jun. 2020, doi: 10.1016/j.optlastec.2020.106366.
- [28] M. Mahmud, N. Atta-Ur-Rahman, M. Lee, and J.-Y. Choi, "Evolutionary-based image encryption using RNA codons truth table," *Optics & Laser Technology*, vol. 121, p. 105818, Sep. 2019, doi: 10.1016/j.optlastec.2019.105818.
- [29] Y. Li, C. Wang, and H. Chen, "A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation," *Optics and Lasers in Engineering*, vol. 90, pp. 238–246, Nov. 2016, doi: 10.1016/j.optlaseng.2016.10.020.
- [30] J. Jin, "An image encryption based on elementary cellular automata," *Optics and Lasers in Engineering*, vol. 50, no. 12, pp. 1836–1843, Jul. 2012, doi: 10.1016/j.optlaseng.2012.06.002.

BIOGRAPHIES OF AUTHORS






Yasmin Abdul    graduated from the Department of Mathematics in 2019 and received master's degree in mathematics from SRM Institute of Science and Technology in 2021. Currently she pursuing her Ph.D. degree in the same institution. Her current area of research includes automata theory, cryptography, and image processing. She can be contacted at email: ya5805@srmist.edu.in.



Venkatesan Ramasamy    working as an assistant professor in the Department of Mathematics, SRM Institute of Science and Technology, India. His current area of research includes formal languages and automata theory, algebraic automata theory, image processing, and cryptography. He can be contacted at email: venkater1@srmist.edu.in.



Gaverchand Kukaram    graduated from the Department of Mathematics in 2019 and received master's degree in mathematics from SRM Institute of Science and Technology in 2021. Currently he pursuing his Ph.D. degree in the same institution. His current area of research includes automata theory, cryptography, DNA computing, and image processing. He can be contacted at email: gk1617@srmist.edu.in.