Security Analysis of an Image Encryption Based on the Kronecker Xor Product, the Hill Cipher and the Sigmoid Logistic Map

George Teşeleanu[1,2,a]
[1] *Advanced Technologies Institute, 10 Dinu Vintilă, Bucharest, Romania*
[2] *Simion Stoilow Institute of Mathematics of the Romanian Academy, 21 Calea Grivitei, Bucharest, Romania*

Keywords: Image Encryption Scheme, Chaos Based Encryption, Cryptanalysis.

Abstract: In 2023, Mfungo *et al.* introduce an image encryption scheme that employs the Kronecker xor product, the Hill cipher and a chaotic map. Their proposal uses the chaotic map to dynamically generate two out of the three secret keys employed by their scheme. Note that both keys are dependent on the size of the original image, while the Hill key is static. Despite the authors' assertion that their proposal offers sufficient security (149 bits) for transmitting color images over unsecured channels, we found that this is not accurate. To support our claim, we present a chosen plaintext attack that requires 2 oracle queries and has a worse case complexity of $O(2^{32})$. Note that in this case Mfungo *et al.*'s scheme has a complexity of $O(2^{33})$, and thus our attack is two times faster than an encryption. The reason why this attack is viable is that the two keys remain unchanged for different plaintext images of the same size, while the Hill key remains unaltered for all images.

1 INTRODUCTION

The security risks associated with digital images, particularly theft and unauthorized distribution, have been amplified by the widespread use of social media. Consequently, researchers have devoted significant attention to this issue and have developed various techniques to encrypt images. Chaotic maps have emerged as a popular choice due to their high sensitivity to initial conditions and previous states, which makes predicting their behavior difficult. As a result, several novel cryptographic algorithms based on chaos have been developed. However, many image encryption schemes based on chaotic maps suffer from critical security vulnerabilities due to inadequate security analysis and a lack of design guidelines. In fact, numerous compromised schemes exist, which are listed non-exhaustively in Table 1. For further information, please refer to (Zolfaghari and Koshiba, 2022; Muthu and Murali, 2021; Hosny, 2020; Özkaynak, 2018).

In (Mfungo et al., 2023), the authors propose a new image encryption scheme that combines the Kronecker xor product, Hill cipher and sigmoid logistic map. More specifically, their algorithm starts by shifting the values in each row of all 4×4 image blocks using the AES shift row operation. Then, the algorithm performs a bitwise xor between the top value of each odd or even column and all other values in the corresponding even or odd column, excluding the top value. Next, the Hill Cipher encrypts each 4×4 block of the result. The resulting image is then xor-ed with a key generated using the sigmoid logistic map. To further obscure the image's pixels, the result is transformed using the Kronecker xor product. Finally, another key generated using the sigmoid logistic map is xor-ed with the output to obtain the encrypted image. Since the sigmoid logistic map is simply used as a pseudorandom number generator (PRNG) and the scheme's weakness is independent of the employed generator, we omit its description and simply consider the two keys as being randomly generated.

The focus of this paper is to carry out a security analysis of the Mfungo *et al.* scheme (Mfungo et al., 2023). We describe a chosen plaintext attack, which would allow an attacker to decrypt all images of a specific size. To execute such an attack, the adversary would need to access the ciphertexts of 2 chosen plaintexts. Once the attacker has this information in his possession, he can easily extract the secret keys. According to the authors, the largest image size that they were able to handle with their available computational resources was limited to 256×256 pixels. Thus, in this case, the key recovery and the encryption

[a] https://orcid.org/0000-0003-3953-2744

Teşeleanu, G.
Security Analysis of an Image Encryption Based on the Kronecker Xor Product, the Hill Cipher and the Sigmoid Logistic Map.
DOI: 10.5220/0012307800003648

In *Proceedings of the 10th International Conference on Information Systems Security and Privacy (ICISSP 2024)*, pages 467-473
ISBN: 978-989-758-683-5; ISSN: 2184-4356

Table 1: Broken chaos based image encryption algorithms.

Scheme	(Yen and Guo, 2000)	(Matoba and Javidi, 2004)	(Wang et al., 2012)	(Huang et al., 2014)	(Khan, 2015)	(Song and Qiao, 2015)	(Chen et al., 2015)
Broken by	(Li and Zheng, 2002)	(Wang et al., 2019)	(Arroyo et al., 2013)	(Wen et al., 2021)	(Alanazi et al., 2021)	(Wen et al., 2019)	(Hu et al., 2017)
Scheme	(Hu et al., 2017)	(Niyat et al., 2017)	(Hua and Zhou, 2017)	(Pak and Huang, 2017)	(Liu et al., 2018)	(Shafique and Shahid, 2018)	(Sheela et al., 2018)
Broken by	(Li et al., 2019a)	(Li et al., 2018)	(Yu et al., 2021)	(Wang et al., 2018)	(Ma et al., 2020)	(Wen and Yu, 2019)	(Zhou et al., 2019)
Scheme	(Wu et al., 2018)	(Yosefnezhad Irani et al., 2019)	(Khan and Masood, 2019)	(Pak et al., 2019)	(Mondal et al., 2021)	(Essaid et al., 2019)	
Broken by	(Chen et al., 2020)	(Liu et al., 2020)	(Fan et al., 2021)	(Li et al., 2019b)	(Li et al., 2021)	(Teşeleanu, 2023)	

processes have a complexity of $O(2^{32})$ and $O(2^{33})$, respectively. However, if the attacker has already computed the Hill key, then only 1 chosen plaintext is required and the complexity of the recovery process is $O(1)$. Keeping all these in mind, using the attack described in this paper we managed to reduce the security of the scheme from 149 bits to 32, and once the Hill key is recovered to 0. Note that we could not devise an efficient chosen ciphertext attack, due to the repetition code embedded in the encryption scheme.

Structure of the Paper. We provide the necessary preliminaries in Section 2. An alternative description of Mfungo *et al.*'s scheme is outlined in Section 3. In Section 4 we show how an attacker can recover all three secret keys in a chosen plaintext scenario. We conclude in Section 5.

2 PRELIMINARIES

Notations. In this paper, the subset $\{1,\ldots,s-1\} \in \mathbb{N}$ is denoted by $[1,s)$. The action of selecting a random element x from a sample space X is represented by $x \xleftarrow{\$} X$, while $x \leftarrow y$ indicates the assignment of value y to variable x. By H and W we denote an image's height and width.

2.1 Mfungo *et al.* Image Encryption Scheme

In this section we present Mfungo *et al.*'s encryption (Algorithm 2) and decryption (Algorithm 3) algorithms as described in (Mfungo et al., 2023). Note that W and H must be divisible by 4.

The first step of the encryption process consists in breaking the image in 4×4 blocks and then circular shifting row i of each block to the left by i positions. The exact function is provided in Algorithm 1 as $shift\_rows$. Note that the function takes as input one of the following matrices

$$shift \leftarrow \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 2 & 3 & 0 \\ 2 & 3 & 0 & 1 \\ 3 & 0 & 1 & 2 \end{bmatrix}$$

or

$$inv\_shift \leftarrow \begin{bmatrix} 0 & 1 & 2 & 3 \\ 3 & 0 & 1 & 2 \\ 2 & 3 & 0 & 1 \\ 1 & 2 & 3 & 0 \end{bmatrix},$$

one for encryption and the other one for decryption. Then the top values of the resulting matrix are preserved, while all values in even columns[1] are xor-ed with the top value of the previous odd column. In the case of odd columns, the values are xor-ed with the top value of the next column, except their top value. The corresponding function is $xor\_between\_pairwise\_columns$ from Algorithm 1. Using a secret 4×4 matrix h, each row of each 4×4 block is multiplied with h. Hill encryption is presented in Algorithm 1, $Hill$. The resulting image is then xor-ed with $k^{(1)}$. Another diffusion layer is then added, *i.e.* the rows are moved down with 3 positions (see Algorithm 1, $shift\_columns$). The Kronecker xor transformation is then applied. More precisely, the authors apply the Kronecker product between the image and itself, with the following modifications: the product between two elements from two distinct positions is replaced by xor, while the ones from the same position remain unaltered. The pseudo-code is given in the $Kronecker\_xor\_transformation$ function from Algorithm 1. Finally, we perform a final xor with the second key $k^{(2)}$.

To decrypt we simply perform all the inverse operations in reverse order. Note that when reversing the Kronecker xor transformation, we should recover the matrices from all $W \times H$ block and take a majority vote for each byte. This is done in order to provide protection against data loss and noise alteration. Basically, the compression of the Kronecker xor transformation is used as a repetition code. Since, we consider the ideal case when oracle answers are relayed unaltered, we simply recover the image from the first $W \times H$ block.

3 A NEW LOOK AT MFUNGO *et al.*'s SCHEME

In this section we present an alternative description of Mfungo *et al.*'s scheme. More precisely, we show

[1] except their top values

Algorithm 1: Helper Functions.

Function $shift\_rows(P, shift)$
  for $i \in [0,W)$ *and at each step increment i with* 4 **do**
    for $j \in [0,H)$ **do**
      for $k \in [0,4)$ **do**
        $index \leftarrow i + shift_{k,j \bmod 4}$
        $Q_{i+k,j} \leftarrow P_{index,j}$
  return Q
Function $xor\_between\_pairwise\_columns(P)$
  for $i \in [0,W)$ **do** $R_{i,0} \leftarrow P_{i,0}$
  for $i \in [0,W)$ *and at each step increment i with* 2 **do**
    for $j \in [1,H)$ **do**
      $R_{i,j} \leftarrow P_{i,j} \oplus P_{i+1,0}$
      $R_{i+1,j} \leftarrow P_{i+1,j} \oplus P_{i,0}$
  return R
Function $Hill(P,h)$
  for $i \in [0,W)$ *and at each step increment i with* 4 **do**
    for $j \in [0,H)$ **do**
      $S_{i,j} \leftarrow P_{i,j}h_{0,0} + P_{i+1,j}h_{0,1} + P_{i+2,j}h_{0,2} + P_{i+3,j}h_{0,3} \bmod 256$
      $S_{i+1,j} \leftarrow P_{i,j}h_{1,0} + P_{i+1,j}h_{1,1} + P_{i+2,j}h_{1,2} + P_{i+3,j}h_{1,3} \bmod 256$
      $S_{i+2,j} \leftarrow P_{i,j}h_{2,0} + P_{i+1,j}h_{2,1} + P_{i+2,j}h_{2,2} + P_{i+3,j}h_{2,3} \bmod 256$
      $S_{i+3,j} \leftarrow P_{i,j}h_{3,0} + P_{i+1,j}h_{3,1} + P_{i+2,j}h_{3,2} + P_{i+3,j}h_{3,3} \bmod 256$
  return S
Function $shift\_columns(P,n)$
  for $i \in [0,W)$ *and* $j \in [0,H)$ **do**
    $T_{i,j} \leftarrow P_{i,j+n \bmod H}$
  return T
Function $Kronecker\_xor\_transformation(P)$
  for $i \in [0,W)$ *and* $j \in [0,H)$ **do**
    for $k \in [0,W)$ *and* $\ell \in [0,H)$ **do**
      if $i = k$ *and* $j = \ell$ **then** $U_{i \cdot W+k, j \cdot H+\ell} \leftarrow P_{i,j}$
      else $U_{i \cdot W+k, j \cdot H+\ell} \leftarrow P_{i,j} \oplus P_{k,\ell}$
  return U
Function $compress\_Kronecker\_xor\_transformation(P)$
  for $i \in [0,W)$ *and* $j \in [0,H)$ **do**
    if $i = 0$ *and* $j = 0$ **then** $T_{i,j} \leftarrow P_{i,j}$
    else $T_{i,j} \leftarrow P_{i,j} \oplus P_{0,0}$
  return T

how to combine $k^{(1)}$ and $k^{(2)}$ into a single key $k^{(3)}$. The alternative encryption and decryption algorithms are provided in Algorithms 4 and 5.

We further show how we derived the equivalent description of lines 4-7, Algorithm 2. After the *shift_row* operation we obtain

$$T_{i,j} \leftarrow S_{i,j+3 \bmod H} \oplus k^{(1)}_{i,j+3 \bmod H}.$$

Applying the Kronecker transformation we get

$$U_{i \cdot W+k, j \cdot H+\ell} \leftarrow T_{i,j} = S_{i,j+3 \bmod H} \oplus k^{(1)}_{i,j+3 \bmod H}$$

when $i = k$ and $j = \ell$ and

$$\begin{aligned} U_{i \cdot W+k, j \cdot H+\ell} \leftarrow T_{i,j} \oplus T_{k,\ell} &= S_{i,j+3 \bmod H} \oplus k^{(1)}_{i,j+3 \bmod H} \\ &\oplus S_{k,\ell+3 \bmod H} \oplus k^{(1)}_{k,\ell+3 \bmod H} \\ &= (S_{i,j+3 \bmod H} \oplus S_{k,\ell+3 \bmod H}) \\ &\oplus (k^{(1)}_{i,j+3 \bmod H} \oplus k^{(1)}_{k,\ell+3 \bmod H}), \end{aligned}$$

Algorithm 2: Encryption algorithm.

Input: A plaintext P, two secret keys $k^{(1)}$ and $k^{(2)}$, and a secret matrix h
Output: A ciphertext C

1 $Q \leftarrow shift\_rows(P, shift)$
2 $R \leftarrow xor\_between\_pairwise\_columns(Q)$
3 $S \leftarrow Hill(R, h)$
4 **for** $i \in [0,W)$ *and* $j \in [0,H)$ **do**
  $S_{i,j} \leftarrow S_{i,j} \oplus k^{(1)}_{i,j}$
5 $T \leftarrow shift\_columns(S, 3)$
6 $U \leftarrow Kronecker\_xor\_transformation(T)$
7 **for** $i \in [0,W^2)$ *and* $j \in [0,H^2)$ **do**
  $C_{i,j} \leftarrow U_{i,j} \oplus k^{(2)}_{i,j}$
8 **return** C

Algorithm 3: Decryption algorithm.

Input: A ciphertext C, two secret keys $k^{(1)}$ and $k^{(2)}$, and a secret matrix h
Output: A plaintext P

for $i \in [0,W^2)$ *and* $j \in [0,H^2)$ **do**
  $U_{i,j} \leftarrow C_{i,j} \oplus k^{(2)}_{i,j}$
$T \leftarrow compress\_Kronecker\_xor\_transformation(U)$
$S \leftarrow shift\_columns(T, -3)$
for $i \in [0,W)$ *and* $j \in [0,H)$ **do**
  $S_{i,j} \leftarrow S_{i,j} \oplus k^{(1)}_{i,j}$
$R \leftarrow Hill(S, h^{-1})$
$Q \leftarrow xor\_between\_pairwise\_columns(R)$
$P \leftarrow shift\_rows(Q, inv\_shift)$
return P

otherwise. Finally, we get

$$C_{i\cdot W+k, j\cdot H+\ell} \leftarrow U_{i\cdot W+k, j\cdot H+\ell} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell} = S_{i,j+3 \bmod H} \oplus (k^{(1)}_{i,j+3 \bmod H} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell})$$

when $i = k$ and $j = \ell$ and

$$C_{i\cdot W+k, j\cdot H+\ell} \leftarrow U_{i\cdot W+k, j\cdot H+\ell} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell} = (S_{i,j+3 \bmod H} \oplus S_{k,\ell+3 \bmod H}) \oplus (k^{(1)}_{i,j+3 \bmod H} \oplus k^{(1)}_{k,\ell+3 \bmod H} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell}),$$

otherwise. Note that if we compose $Kr = Kronecker\_xor\_transformation$ with $sc = shift\_columns$ we get

$$Kr(sc(S,3)) = S_{i,j+3 \bmod H}$$

if $i = k$ and $j = \ell$ and

$$Kr(sc(S,3)) = S_{i,j+3 \bmod H} \oplus S_{k,\ell+3 \bmod H},$$

otherwise. Therefore, if we define $k^{(3)}$ as follows

$$k^{(3)}_{i\cdot W+k, j\cdot H+\ell} = k^{(1)}_{i,j+3 \bmod H} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell},$$

if $i = k$ and $j = \ell$ and

$$k^{(3)}_{i\cdot W+k, j\cdot H+\ell} = k^{(1)}_{i,j+3 \bmod H} \oplus k^{(1)}_{k,\ell+3 \bmod H} \oplus k^{(2)}_{i\cdot W+k, j\cdot H+\ell}$$

otherwise, we get the equivalent description of lines 4-7, Algorithm 2 provided in lines 4-6, Algorithm 4.

Algorithm 4: Equivalent encryption algorithm.

Input: A plaintext P, a secret key $k^{(3)}$, and a secret matrix h
Output: A ciphertext C

1 $Q \leftarrow shift\_rows(P, shift)$
2 $R \leftarrow xor\_between\_pairwise\_columns(Q)$
3 $S \leftarrow Hill(R, h)$
4 $T \leftarrow shift\_columns(S, 3)$
5 $U \leftarrow Kronecker\_xor\_transformation(T)$
6 **for** $i \in [0,W^2)$ *and* $j \in [0,H^2)$ **do**
  $C_{i,j} \leftarrow U_{i,j} \oplus k^{(3)}_{i,j}$
7 **return** C

Algorithm 5: Equivalent decryption algorithm.

Input: A ciphertext C, a secret key $k^{(3)}$, and a secret matrix h
Output: A plaintext P

for $i \in [0,W^2)$ *and* $j \in [0,H^2)$ **do**
  $U_{i,j} \leftarrow C_{i,j} \oplus k^{(3)}_{i,j}$
$T \leftarrow compress\_Kronecker\_xor\_transformation(U)$
$S \leftarrow shift\_columns(T, -3)$
$R \leftarrow Hill(S, h^{-1})$
$Q \leftarrow xor\_between\_pairwise\_columns(R)$
$P \leftarrow shift\_rows(Q, inv\_shift)$
return P

4 CHOSEN PLAINTEXT ATTACK

A chosen plaintext attack (CPA) is a scenario in which the attacker A briefly gains access to the encryption machine O_{enc} and is permitted to query it with various inputs. In this way, A generates specific plaintexts that can facilitate his attack and uses O_{enc} to obtain the corresponding ciphertexts. We demonstrate in this paper that Mfungo *et al.*'s image encryption scheme is vulnerable to such attacks.

In the first step of our attack we aim to retrieve $k^{(3)}$. This can be easily done if we encrypt an image I_0 with all its pixels set to 0. By setting all the pixels to 0, after passing the image through lines 1-5, Algorithm 4 we end up with the same image I_0. Therefore, we retrieve the key from $k^{(3)}_{i,j} = C_{i,j}$.

Let

$$P_{[0,4),[0,4)} = \begin{bmatrix} P_{0,0} & P_{1,0} & P_{2,0} & P_{3,0} \\ P_{0,1} & P_{1,1} & P_{2,1} & P_{3,1} \\ P_{0,2} & P_{1,2} & P_{2,2} & P_{3,2} \\ P_{0,3} & P_{1,3} & P_{2,3} & P_{3,3} \end{bmatrix}.$$

Now we aim to find the secret matrix h. Hence, we create an image I_h such that

$$P_{[0,4),[0,4)} \leftarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

and the remaining pixels are set to 0. Since we are only interested in the first 4×4 block, we will only study its evolution. Thus, after the $shift\_row$ and *xor_between_pairwise_columns* operations we obtain

$$Q_{[0,4),[0,4)} \leftarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

and

$$R_{[0,4),[0,4)} \leftarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Therefore, we obtain that

$$S_{[0,4),[0,4)} \leftarrow \begin{bmatrix} h_{0,0} & h_{1,0} & h_{2,0} & h_{3,0} \\ h_{0,1} & h_{1,1} & h_{2,1} & h_{3,1} \\ h_{0,2} & h_{1,2} & h_{2,2} & h_{3,2} \\ h_{0,3} & h_{1,3} & h_{2,3} & h_{3,3} \end{bmatrix}.$$

is the transpose of h. Since we already know $k^{(3)}$ and the remaining operations are easily reversible, it results that we can retrieve h from the ciphertext corresponding to I_h. The formal description of our CPA attack is provided in Algorithm 6.

The complexity of Algorithm 6 is $O(H^2W^2 + 2HW)$ and we need 2 oracle queries. Note that Mfugo *et al.*'s encryption scheme has a complexity of $O(2H^2W^2 + 8HW)$ and according to the authors the maximum image size that they experimented on is $H = W = 256$. Thus, in this case, our attack has a complexity of $O(2^{32})$, while Mfugo *et al.*'s scheme has one of $O(2^{33})$. Remark that if we already recovered h in a previous iteration, we only need to run lines 2-5, Algorithm 6. Thus, the complexity becomes $O(1)$ and we need 1 oracle query.

Algorithm 6: CPA attack.

1 %recover $k^{(3)}$
2 **for** $i \in [0,H)$ *and* $j \in [0,W)$ **do** $P_{i,j} \leftarrow 0$
3 Send the plaintext P to the encryption oracle O_{enc}.
4 Receive the ciphertext C from the encryption oracle O_{enc}.
5 $k^{(3)} \leftarrow C$
6 %recover h
7 $P_{0,0}, P_{1,0}, P_{2,0}, P_{3,0} \leftarrow 1, 0, 0, 0$
8 $P_{0,1}, P_{1,1}, P_{2,1}, P_{3,1} \leftarrow 0, 0, 0, 0$
9 $P_{0,2}, P_{1,2}, P_{2,2}, P_{3,2} \leftarrow 1, 0, 0, 1$
10 $P_{0,3}, P_{1,3}, P_{2,3}, P_{3,3} \leftarrow 1, 0, 1, 0$
11 Send the plaintext P to the encryption oracle O_{enc}.
12 Receive the ciphertext C from the encryption oracle O_{enc}.
13 **for** $i \in [0,W^2)$ *and* $[0,H^2)$ **do** $U_{i,j} \leftarrow C_{i,j} \oplus k^{(3)}_{i,j}$
14 $T \leftarrow compress\_Kronecker\_xor\_transformation(U)$
15 $S \leftarrow shift\_columns(T, -3)$
16 $h^T \leftarrow S_{[0,4),[0,4)}$
17 **return** $k^{(3)}, h$

5 CONCLUSIONS

In (Mfungo et al., 2023), the authors presented a scheme for encrypting images using a combination of the Kronecker xor product, Hill cipher, and a chaotic map. They claimed that their proposal provided a security strength of 149 bits. However, our analysis of the scheme's security has revealed that its actual strength is only $O(2^{32})$ in the worst-case scenario. Note that the attack only requires two oracle queries. Consequently, the proposed cryptosystem fails to meet the necessary security strength needed to protect confidential information.

REFERENCES

Alanazi, A. S., Munir, N., Khan, M., Asif, M., and Hussain, I. (2021). Cryptanalysis of Novel Image Encryption Scheme Based on Multiple Chaotic Substitution Boxes. *IEEE Access*, 9:93795–93802.

Arroyo, D., Diaz, J., and Rodriguez, F. (2013). Cryptanalysis of a One Round Chaos-Based Substitution Permutation Network. *Signal Processing*, 93(5):1358–1364.

Chen, J., Chen, L., and Zhou, Y. (2020). Cryptanalysis of a DNA-Based Image Encryption Scheme. *Information Sciences*, 520:130–141.

Chen, J.-x., Zhu, Z.-l., Fu, C., Zhang, L.-b., and Zhang, Y. (2015). An Efficient Image Encryption Scheme Using Lookup Table-Based Confusion and Diffusion. *Nonlinear Dynamics*, 81(3):1151–1166.

Essaid, M., Akharraz, I., Saaidi, A., and Mouhib, A. (2019). A New Approach of Image Encryption Based on Dynamic Substitution and Diffusion Operations. In *SysCoBIoTS 2019*, pages 1–6. IEEE.

Fan, H., Zhang, C., Lu, H., Li, M., and Liu, Y. (2021). Cryptanalysis of a New Chaotic Image Encryption Technique Based on Multiple Discrete Dynamical Maps. *Entropy*, 23(12):1581.

Hosny, K. M. (2020). *Multimedia Security Using Chaotic Maps: Principles and Methodologies*, volume 884. Springer.

Hu, G., Xiao, D., Wang, Y., and Li, X. (2017). Cryptanalysis of a Chaotic Image Cipher using Latin Square-Based Confusion and Diffusion. *Nonlinear Dynamics*, 88(2):1305–1316.

Hua, Z. and Zhou, Y. (2017). Design of Image Cipher Using Block-Based Scrambling and Image Filtering. *Information sciences*, 396:97–113.

Huang, X., Sun, T., Li, Y., and Liang, J. (2014). A Color Image Encryption Algorithm Based on a Fractional-Order Hyperchaotic System. *Entropy*, 17(1):28–38.

Khan, M. (2015). A Novel Image Encryption Scheme Based on Multiple Chaotic S-Boxes. *Nonlinear Dynamics*, 82(1):527–533.

Khan, M. and Masood, F. (2019). A Novel Chaotic Image Encryption Technique Based on Multiple Discrete Dynamical Maps. *Multimedia Tools and Applications*, 78(18):26203–26222.

Li, M., Lu, D., Wen, W., Ren, H., and Zhang, Y. (2018). Cryptanalyzing a Color Image Encryption Scheme Based on Hybrid Hyper-Chaotic System and Cellular Automata. *IEEE access*, 6:47102–47111.

Li, M., Lu, D., Xiang, Y., Zhang, Y., and Ren, H. (2019a). Cryptanalysis and Improvement in a Chaotic Image Cipher Using Two-Round Permutation and Diffusion. *Nonlinear Dynamics*, 96(1):31–47.

Li, M., Wang, P., Liu, Y., and Fan, H. (2019b). Cryptanalysis of a Novel Bit-Level Color Image Encryption Using Improved 1D Chaotic Map. *IEEE Access*, 7:145798–145806.

Li, M., Wang, P., Yue, Y., and Liu, Y. (2021). Cryptanalysis of a Secure Image Encryption Scheme Based on a Novel 2D Sine–Cosine Cross-Chaotic Map. *Journal of Real-Time Image Processing*, 18(6):2135–2149.

Li, S. and Zheng, X. (2002). Cryptanalysis of a Chaotic Image Encryption Method. In *ISCAS 2002*, volume 2, pages 708–711. IEEE.

Liu, L., Hao, S., Lin, J., Wang, Z., Hu, X., and Miao, S. (2018). Image Block Encryption Algorithm Based on Chaotic Maps. *IET Signal Processing*, 12(1):22–30.

Liu, Y., Qin, Z., Liao, X., and Wu, J. (2020). Cryptanalysis and Enhancement of an Image Encryption Scheme Based on a 1-D Coupled Sine Map. *Nonlinear Dynamics*, 100(3):2917–2931.

Ma, Y., Li, C., and Ou, B. (2020). Cryptanalysis of an Image Block Encryption Algorithm Based on Chaotic Maps. *Journal of Information Security and Applications*, 54:102566.

Matoba, O. and Javidi, B. (2004). Secure Holographic Memory by Double-Random Polarization Encryption. *Applied Optics*, 43(14):2915–2919.

Mfungo, D. E., Fu, X., Wang, X., and Xian, Y. (2023). Enhancing Image Encryption with the Kronecker Xor Product, the Hill Cipher, and the Sigmoid Logistic Map. *Applied Sciences*, 13(6).

Mondal, B., Behera, P. K., and Gangopadhyay, S. (2021). A Secure Image Encryption Scheme Based on a Novel 2D Sine–Cosine Cross-Chaotic (SC3) Map. *Journal of Real-Time Image Processing*, 18(1):1–18.

Muthu, J. S. and Murali, P. (2021). Review of Chaos Detection Techniques Performed on Chaotic Maps and Systems in Image Encryption. *SN Computer Science*, 2(5):1–24.

Niyat, A. Y., Moattar, M. H., and Torshiz, M. N. (2017). Color Image Encryption Based on Hybrid Hyper-Chaotic System and Cellular Automata. *Optics and Lasers in Engineering*, 90:225–237.

Özkaynak, F. (2018). Brief Review on Application of Nonlinear Dynamics in Image Encryption. *Nonlinear Dynamics*, 92(2):305–313.

Pak, C., An, K., Jang, P., Kim, J., and Kim, S. (2019). A Novel Bit-Level Color Image Encryption Using Improved 1D Chaotic Map. *Multimedia Tools and Applications*, 78(9):12027–12042.

Pak, C. and Huang, L. (2017). A New Color Image Encryption Using Combination of the 1D Chaotic Map. *Signal Processing*, 138:129–137.

Shafique, A. and Shahid, J. (2018). Novel Image Encryption Cryptosystem Based on Binary Bit Planes Extraction and Multiple Chaotic Maps. *The European Physical Journal Plus*, 133(8):1–16.

Sheela, S., Suresh, K., and Tandur, D. (2018). Image Encryption Based on Modified Henon Map Using Hybrid Chaotic Shift Transform. *Multimedia Tools and Applications*, 77(19):25223–25251.

Song, C. and Qiao, Y. (2015). A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy*, 17(10):6954–6968.

Teşeleanu, G. (2023). Security Analysis of a Color Image Encryption Scheme Based on Dynamic Substitution and Diffusion Operations. In *ICISSP 2023*. SCITEPRESS.

Wang, H., Xiao, D., Chen, X., and Huang, H. (2018). Cryptanalysis and Enhancements of Image Encryption Using Combination of the 1D Chaotic Map. *Signal processing*, 144:444–452.

Wang, L., Wu, Q., and Situ, G. (2019). Chosen-Plaintext Attack on the Double Random Polarization Encryption. *Optics Express*, 27(22):32158–32167.

Wang, X., Teng, L., and Qin, X. (2012). A Novel Colour Image Encryption Algorithm Based on Chaos. *Signal Processing*, 92(4):1101–1108.

Wen, H. and Yu, S. (2019). Cryptanalysis of an Image Encryption Cryptosystem Based on Binary Bit Planes Extraction and Multiple Chaotic Maps. *The European Physical Journal Plus*, 134(7):1–16.

Wen, H., Yu, S., and Lü, J. (2019). Breaking an Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy*, 21(3):246.

Wen, H., Zhang, C., Huang, L., Ke, J., and Xiong, D. (2021). Security Analysis of a Color Image Encryption Algorithm Using a Fractional-Order Chaos. *Entropy*, 23(2):258.

Wu, J., Liao, X., and Yang, B. (2018). Image Encryption Using 2D Hénon-Sine Map and DNA Approach. *Signal processing*, 153:11–23.

Yen, J.-C. and Guo, J.-I. (2000). A New Chaotic Key-Based Design for Image Encryption and Decryption. In *ISCAS 2000*, volume 4, pages 49–52. IEEE.

Yosefnezhad Irani, B., Ayubi, P., Amani Jabalkandi, F., Yousefi Valandar, M., and Jafari Barani, M. (2019). Digital Image Scrambling Based on a New One-Dimensional Coupled Sine Map. *Nonlinear Dynamics*, 97(4):2693–2721.

Yu, F., Gong, X., Li, H., and Wang, S. (2021). Differential Cryptanalysis of Image Cipher Using Block-Based Scrambling and Image Filtering. *Information Sciences*, 554:145–156.

Zhou, K., Xu, M., Luo, J., Fan, H., and Li, M. (2019). Cryptanalyzing an Image Encryption Based on a Modified Henon Map Using Hybrid Chaotic Shift Transform. *Digital Signal Processing*, 93:115–127.

Zolfaghari, B. and Koshiba, T. (2022). Chaotic Image Encryption: State-of-the-Art, Ecosystem, and Future Roadmap. *Applied System Innovation*, 5(3):57.