Chaotic Map Based Image Encryption with Scan Pattern Technique

Mr. Sandeep B[1], Rakshith D S[2], Nikhil B R[3], Pramod Nayaku Totakar[4], Mukesh Chetty D[5]

Department of Computer Science and Engineering

Jawaharlal Nehru New College of Engineering, Shivamogga, Karnataka, India. sandeepb@jnnce.ac.in,

rakshithds27@gmail.com , nikhilbalappar@gmail.com, pramodntotakar.kcs2020@gmail.com,

mukeshchettydk@gmail.com

***Abstract*— This work proposes an image encryption method combining chaotic maps and a dynamic scan pattern for enhanced security. Chaotic maps, with their pseudorandom behavior, generate cryptographic keys that add unpredictability to the encryption process. These keys dynamically influence the encryption algorithm, increasing security. Additionally, a unique scan pattern is used to shuffle pixel values, improving the diffusion of information. The method starts with the chaotic map to generate keys and then applies the scan pattern to redistribute pixels, further strengthening the encryption.**

I. INTRODUCTION

With the rapid growth of data communication, securing sensitive information is crucial. Cryptographic techniques are needed to protect data before transmission, especially for images, which are larger and more complex than text. Traditional encryption methods like DES, AES, and RSA are inefficient for image encryption due to size and redundancy.

This work introduces a chaotic map-based encryption approach for images, leveraging the nonlinearity and unpredictability of chaos theory. Chaotic maps generate pseudorandom streams or control parameters, making them suitable for both stream and block ciphers. Chaos-based encryption offers fast, secure implementation with high sensitivity and efficiency.

II. LITERATURE SURVEY

The three image encryption techniques enhance security through cryptography and steganography. The first method, *"Image Encryption using Scan Patterns and Hill Cipher,"* uses SCAN patterns and the Hill cipher for multi-layered encryption, improving security but adding complexity and overhead. The second, *"Triple-Layer Image Security Using a Zigzag Embedding Pattern,"* combines AES-128, chaotic encryption, and zigzag-based steganography, offering strong security and high image quality but being computationally expensive. The third method, *"Image Encryption and Decryption Using Scan Pattern and XTEA Encryption Algorithm,"* uses SCAN patterns with XTEA encryption for efficient image security, though it can cause distortion and XTEA has known vulnerabilities. These methods provide robust image encryption with varying trade-offs in complexity and performance.

The proposed image encryption techniques use chaotic systems to enhance security. *"An Image Encryption Technique using Logistic Map and Z-order Curve"* combines the chaotic Logistic map and Z-order curve to generate a key stream based on the image, offering strong encryption but potential vulnerability to sophisticated attacks. *"An Image Cryptography using Henon Map and Arnold Cat Map"* uses Henon's chaotic system and the Arnold Cat Map for pixel shuffling, improving security with a computational cost. *"Chaos-Based Confusion and Diffusion of Image Pixels Using Dynamic Substitution"* applies Henon and Ikeda maps with dynamic substitution for enhanced pixel confusion and diffusion, ensuring high security but with increased computational complexity. These methods all leverage chaos theory for secure image encryption but face challenges with computational overhead and potential vulnerabilities.

The studies on image encryption using chaotic maps explore methods to secure image data, particularly in sectors like telemedicine and digital communication. One approach uses Henon chaotic maps, offering strong security but facing implementation and computational complexity challenges. Another combines Logistic, Tent, and Sine maps for better encryption, though efficiency is reduced. A system for tele-dermatology employs Arnold's cat map and Henon map for confusion and diffusion, providing strong security but with high computational demands. A hybrid encryption method using these maps improves security but slows processing speed. Additionally, a survey compares various chaotic maps, highlighting their strengths and weaknesses, though some methods struggle with speed. These techniques enhance data security but present efficiency and real-time application challenges.

III. METHADOLOGY

The methodology involves selecting an image, applying chaos-based encryption using maps like Henon and Arnold's Cat map, and generating a secret key. The image is encrypted by scrambling pixel positions and values, with decryption reversing the process. Performance is evaluated based on statistical measures and efficiency, followed by conclusions and future research directions.

A. System Design

A hybrid chaotic encryption algorithm is proposed for enhanced security in image transmission. The encryption process uses chaotic maps, such as Arnold's Cat map and Logistic map, to generate key sequences for encrypting and decrypting the input image. Chaotic maps provide two key properties: confusion, which obscures the relationship between the ciphertext and the key, making it hard to deduce the key from the ciphertext, and diffusion, where a single bit change in the plaintext or ciphertext alters about half of the bits, ensuring an avalanche effect. These properties improve the security and privacy of the encryption system.

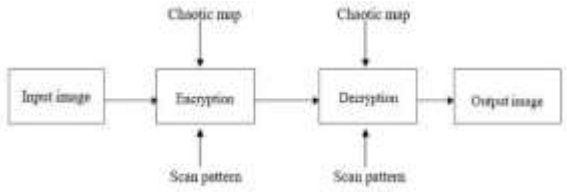

Fig no.1 : System design

B. Encryption

This image encryption scheme uses chaotic maps (Logistic and Arnold's Cat maps) and scan patterns. The process includes three stages: a Zeta Z scan of the image, confusion with Arnold's cat map, and diffusion using the Spiral-In scan pattern. Random key sequences from the Logistic map further enhance security. An XOR operation between the shuffled image and key sequence creates the encrypted image, ensuring confidentiality.

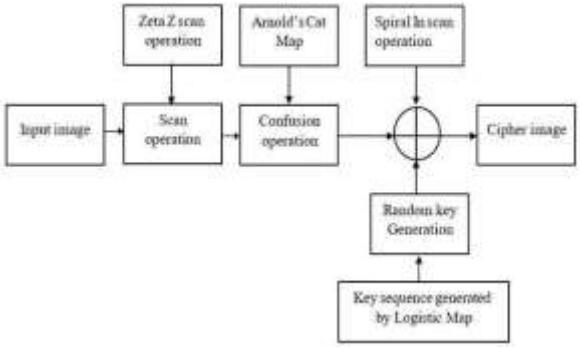

Fig no. 2 : Encryption process

C. Decryption

In the decryption process, the steps of encryption are reversed to recover the original image. The cipher image undergoes a Diffusion operation using the Spiral-In scan pattern and a key sequence generated by the Logistic map. An XOR operation is applied between the pixel values and the key sequence to begin reconstructing the image. The Confusion operation follows, using Arnold's cat map, and the Zeta Z scan pattern finalizes the process, successfully retrieving the original image from the cipher.

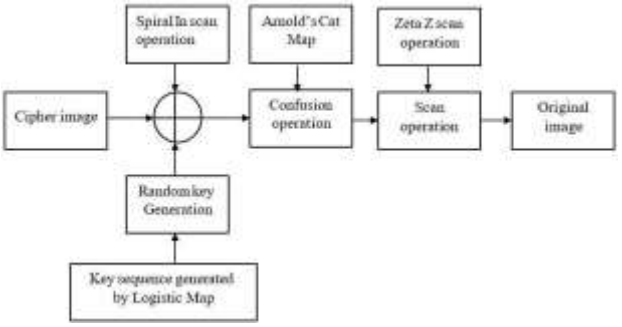

Fig no. 3 : Decryption process

IV. IMPLEMENTATION DETAILS

1. Encryption Algorithm

The encryption process begins with selecting the original image in .png format. The first step is to perform the scan operation using the Zeta Z scan pattern, based on the image's height and width. After this, the confusion operation is applied to the image using Arnold's Cat map, which introduces chaotic behavior to the pixel positions. Next, random key sequences are generated using the Logistics map, which add an additional layer of unpredictability. The diffusion operation follows, where the image pixels are shuffled using the Spiral-In scan pattern, which also exhibits chaotic behavior. Then, the XOR operation is performed between the pixel values derived from the Spiral-In pattern and the key sequence generated by the Logistics map. Finally, the encrypted image, or Cipher image, is obtained, completing the encryption process.

2. Scan Operation

In the encryption process, the image pixels are rearranged using the Zeta Z scan pattern. For a 256×256 image, it is divided into 8×8 blocks, and the Zeta Z scan pattern shuffles the pixels in a zig-zag manner, as illustrated.

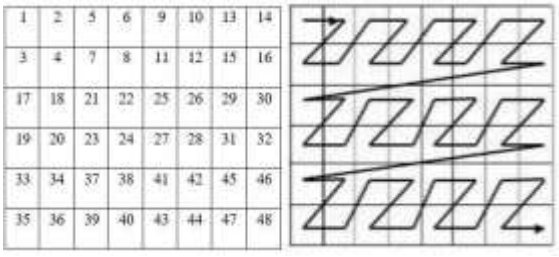

Fig no. 4 : scan operation in encryption

In this stage, the Lena image (256x256) is converted to grayscale, and its pixels are shuffled using Arnold's Cat map. The map's secret key, defined by its initial conditions, is used to confuse the pixel positions, making the image unidentifiable. Arnold's Cat map rearranges pixel positions but eventually returns to the original state after several iterations. The map is described by:

$\Gamma:(x,y)=(2x+y,x+y) \bmod 1$

Where (Xn,Yn) (X_n, Y_n) represents the original pixel location and (Xn+1,Yn+1)(X_{n+1}, Y_{n+1}) is the new shuffled location.

To improve the security further, the third stage of the encryption process aims to generate encryption key sequence. Where the encryption key sequence is generated by logistic map.

$$Xn+1 = \mu * Xn(1 - Xn) \quad (2)$$

Where the parameter μ is taken in the range [3.57 to 4]. Here, the initial value X0 and the parameter μ are regarded as the secret keys

3. Decryption Algorithm

The decryption process starts by selecting the cipher image, followed by the Diffusion operation, where pixels are rearranged using the Spiral-In scan pattern. A random key sequence is generated using the Logistics map, and the XOR operation is applied between the pixel values and key sequence to reconstruct the image. The Confusion operation is then performed using Arnold's Cat map, and the Zeta Z scan pattern is applied to retrieve the original image, completing the decryption process.

The decryption process begins with the cipher image, which is shuffled using the Spiral-In scan operation and then XORed with the key sequence generated by the Logistic map. The decryption key sequence is created using the Logistic map. The reverse diffused image is then subjected to the confusion process, where the pixel positions are shuffled using Arnold's Cat map, with the initial conditions acting as the second secret key. Finally, the Zeta Z scan operation is applied to the image, rearranging the pixels and restoring the original image through interpolation and inverse transformation techniques, ensuring accurate image retrieval.

V. RESULT

The encryption algorithm, based on chaotic maps like the Arnold's cat map and Logistic map, effectively transforms the Lena image (512x512) into a random-like cipher image. The Arnold's cat map scrambles the image pixels, while the Logistic map generates a key sequence for confusion and diffusion, enhancing security. The encrypted image appears as random noise, making it unrecognizable without the correct key. Upon decryption, using the same key, the original image is successfully restored. The system's performance was evaluated using various metrics: the correlation coefficient between the original and encrypted images was low, indicating effective encryption; the MSE was high, and the PSNR was low, showing minimal quality loss; NPCR and UACI values were high, signifying the system's sensitivity to changes in the input image; SSIM was low, demonstrating significant structural difference between the original and encrypted images; and the entropy was high, reflecting increased randomness. These results confirm the encryption system's strength in securing images, with the Python-based implementation providing an intuitive GUI via Tkinter.

Tkinter is a widely used Python library for creating graphical user interfaces (GUIs), offering a simple and intuitive interface for developers. It is known for its ease of use, making it a preferred choice for both beginners and experienced developers alike. Tkinter provides a range of features for building functional and interactive applications, and its cross-platform compatibility ensures that applications can run on multiple operating systems without modification. The layout of the Graphical User Interface (GUI), as shown in the figure, demonstrates the various components integrated for the encryption and decryption processes. These components typically include buttons, text fields, and labels, all designed to facilitate user interaction with the cryptographic functions.

Fig no. 5 : Layout of the GUI

The Browse Button is used to select the input image as shown in Fig 6

Fig no. 6 : Input image

The input image is processed for confusion by shuffling its pixels using the Zeta Z scan pattern. The resulting image, known as the scan image.

The input image is processed for confusion by applying the Zeta Z scan pattern, which rearranges the pixels in a specific, chaotic manner. This shuffling of pixels obscures the original structure of the image, making it unrecognizable. The resulting image, referred to as the "scan image," is a transformed version of the input, where pixel positions have been altered without changing their values. This step enhances the security of the encryption process by introducing complexity and reducing any patterns that could be exploited by potential attackers. The scan image serves as an intermediate step before further cryptographic operations are applied, such as confusion and diffusion.

Fig no. 7 : Scan Image

The use of the chaotic Arnold's Cat map in the confusion phase significantly strengthens the encryption system by creating complex, non-linear transformations of the image's pixel positions. This randomness ensures that the encrypted image bears no resemblance to the original, making it resistant to pattern recognition techniques or cryptanalysis. Additionally, the inherent sensitivity of the map to initial conditions (key) means that slight changes in the key or sequence will lead to vastly different encrypted images. This contributes to the system's robustness against brute-force attacks. As the map iterates, it scrambles pixel positions in a manner that enhances both the security and unpredictability of the encrypted image, ensuring that it cannot be deciphered without the correct decryption key.

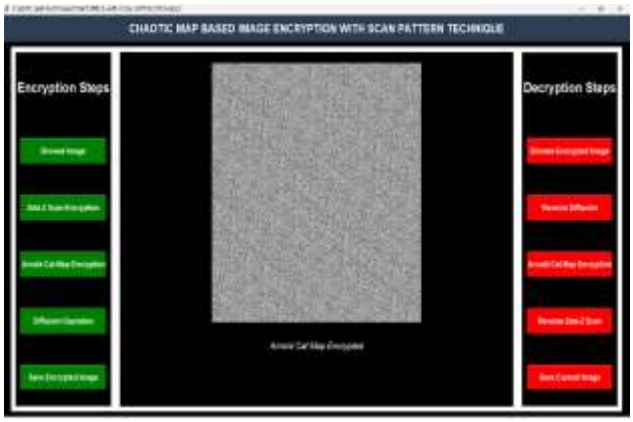

Fig no. 8 : Confused image

The random key sequences produced by the Logistic map are used to shuffle the image pixels during the Diffusion operation with the Spiral-In scan pattern, enhancing the encryption's chaotic nature. The XOR operation is then applied between the pixel values from the Spiral-In scan pattern and the key sequence, creating a complex transformation of the original image. This procedure effectively hides the original content, ensuring the encrypted image remains unintelligible without the proper decryption key. The resulting Cipher image, shown in Fig , illustrates the effectiveness of this process in safeguarding the image's confidentiality.

Fig no. 9 : Cipher image

An image decryption algorithm must successfully revert the cipher image back to the original image, provided the correct key is used. The decryption process begins by taking the cipher image, shown in Fig, as the input. The key sequence, generated during the encryption phase, is applied in reverse to unravel the changes introduced during encryption. Using the inverse of the operations performed in encryption—such as reversing the diffusion and confusion steps—the original image is reconstructed. This ensures that only the correct key can correctly retrieve the original image, preserving both security and integrity throughout the process.

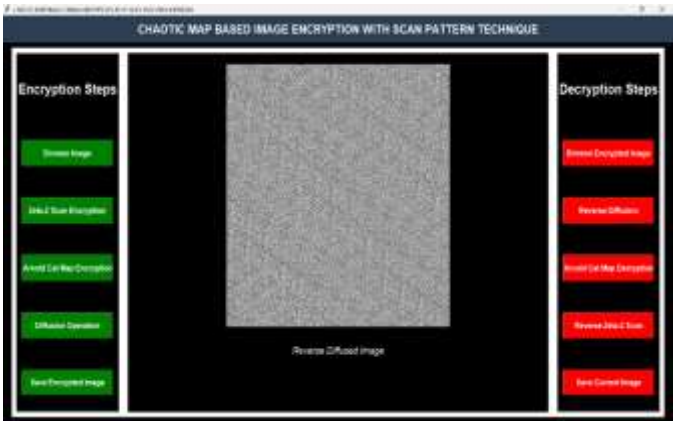

Fig no. 10 : Reversed diffusion operation button

The confused image obtained from the diffusion process is further processed using the Arnold's Cat map operation, which rearranges the pixel positions, effectively reducing the level of confusion and returning the image closer to its original form. This operation creates a scan pattern image, as shown in Fig 6. Next, the Zeta Z scan operation is applied to the image, which rearranges the pixels in a specific, predefined pattern, ultimately restoring the image to its original state. The result, displayed in Fig 11, marks the final step in the decryption process, successfully recovering the original image from the cipher.

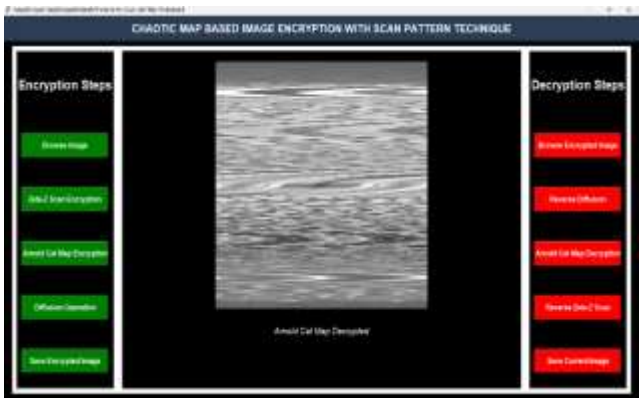

Fig no. 11 : Decryption result

After the Zeta Z scan pattern is applied to shuffle the image pixels, the reverse of Arnold's Cat map is then performed to retrieve the scan image. This decryption step is executed using the Arnold's Cat Map Decryption button, as shown in Fig 12. By reversing the chaotic transformation applied during encryption, the positions of the pixels are restored to their original form, although the image remains in its shuffled state. This process is crucial for unraveling the confusion introduced earlier in the encryption cycle, allowing the system to progressively restore the original image in subsequent stages of decryption.

Fig no. 11 : Output image

An encryption technique is only valuable if it can withstand attacks from intruders, interceptors, or other potential threats. The strength of an encryption algorithm is determined by its ability to resist such attacks. Therefore, before implementing any encryption process in real-world applications, it is essential to thoroughly test its robustness. Failure to do so would result in wasted time and resources. To evaluate the strength of the proposed algorithm, five types of tests are typically conducted. These tests assess various aspects of security, including resistance to unauthorized access, encryption reliability, and data integrity. Each test provides insights into the algorithm's performance under different conditions, ensuring that it can be trusted for secure data protection.

VI. CONCLUSION

To further elaborate on the encryption process, the Zeta Z scan pattern used in the first step is crucial because it reorders the pixels in a seemingly random yet controlled manner. This initial step of shuffling the pixels based on the dimensions of the image prevents the original structure from being easily recognized, laying the foundation for the following stages of transformation. The scan pattern ensures that the positions of the pixels are altered systematically, yet in a complex manner, increasing the difficulty of decrypting the image without the correct decryption key.

Arnold's Cat map, applied during the confusion stage, is one of the most effective chaotic systems used in cryptography due to its ability to generate seemingly random outputs from deterministic input. By iterating over the image multiple times, the map progressively distorts the image's pixels, further contributing to the unrecognizability of the image. This chaotic behavior ensures that even a slight change in the initial conditions or parameters of Arnold's Cat map leads to entirely different outputs, reinforcing the security of the encryption process. The key generated by Arnold's Cat map can only be reversed if the correct parameters and initial conditions are known.

The next stage, key generation, plays an essential role in strengthening the encryption. The Logistic map, a well-known chaotic system, generates random sequences that are sensitive to initial conditions. This randomness is important because it ensures that the key sequence is not easily predictable, making it difficult for attackers to perform a successful brute-force attack. Since the key sequence is derived from the chaotic behavior of the Logistic map, it provides an additional layer of unpredictability and complexity to the encryption process.

During the Diffusion stage, the Spiral-In scan pattern is used to scatter the pixel values across the image. This pattern further disrupts the spatial correlation of pixel values, ensuring that each pixel is affected by its neighbors in a chaotic manner. By diffusing the image, this operation increases the entropy of the encrypted image, making it even harder to correlate the ciphertext to the original image. The final XOR operation between the randomized key sequence and the pixel values ensures that even if an attacker has some partial knowledge of the image, they cannot deduce the original data without the key.

The final encrypted image, or Cipher image, is essentially a cryptographically transformed version of the original image, where the pixel values are obscured beyond recognition. Only through the correct decryption process, which reverses each of these complex transformations in the proper sequence, can the original image be recovered. The combination of chaotic systems, including the Zeta Z scan pattern, Arnold's Cat map, the Logistic map, and the Spiral-In scan pattern, makes this encryption process robust and resilient against a wide range of cryptographic attacks, ensuring that sensitive image data remains protected.

VII. REFERENCE

1. V Praveen et al. (2018) - Image encryption using scan patterns and Hill Cipher (RTEICT Conference).

2. Sara Farrag et al. (2019) - Triple-layer image security using a zigzag embedding pattern.

3. Krishna Raj A et al. (2018) - Image encryption with scan patterns and XTEA algorithm (IOSR-JECE).

4. Shana K U et al. (2018) - Image encryption using the Logistic map and Z-order curve (IEEE Access).

5. Pranjali Sankhe et al. (2018) - Cryptography using Henon map and Arnold's Cat map.

6. Abdhulla Qayyum et al. (2020) - Chaos-based image confusion and diffusion with dynamic substitution (IEEE Access).

7. Diya Achu Pradeep et al. (2021) - Image encryption using chaotic maps (IEEE Access).

8. C Manikandan et al. (2023) - Exploring efficacy of image encryption with chaotic maps (ICOSEC 2023).

9. Anak Agung Putri Ratna et al. (2021) - Chaos-based image encryption using Arnold's Cat map and Henon map (ASTES).

10. Veena G & Dr. Ramakrishna M (2021) - Survey on image encryption using chaos-based techniques (IJACSA).