DYNAMIC DATA STRUCTURE-BASED IMAGE SECURITY METHOD

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

DYNAMIC DATA STRUCTURE-BASED IMAGE SECURITY METHOD

Submitted in the partial fulfillment of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

under

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

BY

Swapan Kumar Shee

University Enrollment No.12021002001009

Registration No. 204202100200014

Shivam Kumar Mishra

University Enrollment. No.12021002001042

Registration No. 204202100200046

UNDER THE GUIDANCE OF

Prof. Jyoti Khandelwal

COMPUTER SCIENCE & ENGINEERING

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Approval Certificate

This is to certify that the project report entitled **“DYNAMIC DATA STRUCTURE-BASED IMAGE SECURITY METHOD”** submitted by **Swapan Kumar Shee (Roll:12021002001009) & Shivam Kumar Mishra (Roll:12021002001042)** in partial fulfillment of the requirements of the degree of **Bachelor of Technology** in **Computer Science & Engineering** from University **of Engineering and Management, Jaipur** was carried out in a systematic and procedural manner to the best of our knowledge. It is a bona fide work of the candidate and was carriedout under our supervision and guidance during the academic session of 2021- 2025.

Prof. Jyoti Khandelwal
Project Guide, Assistant Professor (CSE)
UEM, JAIPUR

Prof. Mrinal Kanti Sarkar
HOD (CSE)
UEM, JAIPUR

Prof. A Mukherjee
Dean
UEM, JAIPUR

ACKNOWLEDGEMENT

The endless thanks go to Lord Almighty for all the blessings he has showered on us, which has enabled us to write this last note in our research work. During the period of our research, as in the rest of our life, we have been blessed by Almighty with some extraordinary people who have spun a web of support around us. Words can never be enough to express how grateful we are to those incredible people in our life who made this thesis possible. We would like to thank them for making our time during my research in the Institute a period We will treasure. We deeply indebted to our research supervisor, Professor Jyoti Khandelwal for such an interesting thesis topic. Each meeting with her added invaluable aspects to the implementation and broadened our perspective. She has guided us with his invaluable suggestions, lightened up the way in our darkest times, and encouraged us a lot in academic life.

Swapan Kumar Shee

Shivam Kumar Mishra.

ABSTRACT

The burgeoning field of digital communication necessitates robust security measures to safeguard sensitive image data during transmission or storage. This project delves into the realm of image cryptography, a critical technique for scrambling image information, rendering it unintelligible to unauthorized parties.

This project unveils a captivating technique for image manipulation, offering a robust solution for scrambling and descrambling visual data. The proposed method leverages the elegance of binary trees and the efficiency of hash tables to achieve secure image transformation. By meticulously rearranging pixel intensities within a meticulously crafted binary tree structure, the scrambling process conceals the image's original content. To unveil the hidden image, a complementary descrambling technique meticulously reconstructs the original pixel order using the meticulously designed hash table and the inherent order within the binary tree. This captivating approach ensures the integrity of the image data while offering a robust layer of obfuscation.

Keywords: image manipulation, scrambling, descrambling, binary trees, hash tables, image transformation,

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1.

INTRODUCTION

In today's digital age, the proliferation of image data across various platforms has accentuated the need for robust methods of preserving data integrity and ensuring confidentiality. From personal photographs to sensitive documents, the importance of safeguarding visual content against unauthorized access cannot be overstated. In response to this imperative, our project endeavors to explore innovative techniques for image scrambling and descrambling, leveraging the synergy between advanced data structures and image processing algorithms.

The objective of our project is two-fold: first, to develop a method for scrambling images in a manner that renders them visually incomprehensible while preserving their structural integrity, and second, to devise an efficient descrambling mechanism capable of faithfully reconstructing the original image from its scrambled counterpart. Central to our approach is the integration of binary tree structures and hash tables, which form the backbone of our scrambling and descrambling algorithms, respectively.

In the subsequent sections of this report, we shall delve into the theoretical underpinnings of our methodology, elucidating the principles of binary tree construction, hash table implementation, and their application to image manipulation. We shall also present a detailed account of our implementation strategy, discussing the intricacies of image preprocessing, data structure instantiation, and algorithmic optimization.

Through this project, we aim to contribute to the burgeoning field of image security and privacy, offering a pragmatic solution that combines mathematical rigor with practical utility. By elucidating the inner workings of our algorithms and showcasing their efficacy through empirical evaluations, we seek to establish a foundation for further research and development in this domain.

In the chapters that follow, we shall embark on a journey through the intricacies of image scrambling and descrambling, unraveling the mysteries of data transformation and reconstruction, and ultimately, illuminating a path towards enhanced image security in the digital age.

CHAPTER 2.

LITERATURE REVIEW

There were a lot of researchers present who had already completed research on the same topic. From their research, lot of information has been gathered. Some of these are discussed below:

In [1], proposes an improved reversible data hiding scheme in encrypted images using parametric binary tree labeling (IPBTL-RDHEI), which takes advantage of the spatial correlation in the entire original image but not in small image blocks to reserve room for hiding data. The IPBTL-RDHEI method involves encrypting the original image with a key and categorizing encrypted pixels using a parametric binary tree. It then embeds secret information by replacing bits in one of the two categories of encrypted pixels. Experimental results show that IPBTL-RDHEI achieves a higher embedding rate and outperforms competing methods. Notably, IPBTL-RDHEI allows for reversible extraction of both the original plaintext image and the secret information without loss.

In [2], binary search tree (BST) is implemented to produce a new algorithm for image encryption. The BST is utilized to generate an encryption key that consists of two parts; local and global with flexible length capabilities that provide better security. Sharing all image contents to encrypt any byte of the source image helped to achieve Shannon's concept of diffusion and confusion. The experimental application of this algorithm has manifested a satisfactory security performance as compared with the widely used cryptographic systems such as Advanced Encryption Standard (AES) and Data Encryption Standard (DES). These Comparisons included measurement of encryption time complexity, Peak Signal to Noise Ratio (PSNR), Entropy, encryption key space. Besides, the new method offers encryption key length flexibility and involvement of all image contents in its generation.

In [3], different keys are used for each round to secure the image from a chosen plain text and a chosen cipher text attack. In internal key generation process, external keys are given as input instead of random initial value and keys are internally generated. The permutation order is generated to permute the image and diffusion bits are used to diffuse the image. Different keys are selected to encrypt the image for every round.

In [4], proposes an image encryption algorithm based on hash table structure scrambling and DNA substitution. The algorithm uses the classical 'scrambling-diffusion' process, and the pseudo-random sequence used in each process is generated by the hyper-chaotic Chen system. Firstly, in the process of scrambling, two sequences with no repeated values are constructed by using the closed hash method in the hash table structure for chaotic sequences, and the plain image is scrambled twice according to the two sequences. Then, in the process of diffusion, DNA coding and decoding rules and DNA substitution rules are made according to the chaotic sequences. The image is dynamically encoded, substituted and decoded with DNA. In this way, while modifying the gray value of image pixels, it also destroys the strong correlation between adjacent pixels. Through a series of verifications, it is concluded that the algorithm not only has good encryption effect, high security and large key space, but also has the advantages of high key sensitivity and strong anti-attack capability.

In [5], presents a novel idea for symmetric image encryption utilizing binary search tree mechanism for generating the encryption/decryption key. This key consists of local and global parts with free user controlled key length, hence, it provides good flexibility for high security, which is manifested in the encrypted images. For every single byte of the encrypted image, the substitution and transposition operations involve all the image contents, fulfilling Shannon's diffusion and confusion concept. Experimental computations of the proposed encryption method have shown satisfactory visual and image energy distribution histogram. Besides, the Peak Signal to Noise Ratio (PSNR) and Normalized Mean Absolute Error (NMAE) measurement results were comparable with the widely used cryptosystems such as DES and AES. On the other hand, the proposed method has flexible key length and shorter execution time by 10% or more depending on the image contents.

In [6], generates another new concept called Visual Cryptography, which is extracted from the technique called secret image sharing. This technique facilitates the distribution of secret among the participants who are all involved in the secret sharing scheme via the communication channel. The generated secret image shares do not reveal any information about the secret if they are not aligned in a prescribed way. The proposed algorithm is evaluated based on its running time of the shares and threshold sharing patterns. As a result of this evaluation, the (2, 2) sharing scheme and diagonal sharing pattern propagates less time with higher security. The most commonly used file formats are taken and it is analyzed by proposed algorithm. That analysis concludes, the proposed algorithm is well suit for PNG file format and the security level of image is also very high as compared to other file formats. The various stages of the proposed algorithm are analyzed by using MATLAB and their corresponding metric values are also obtained. The obtained metric values are compared with the benchmarked image security algorithm and its shares. The proposed algorithm outperforms the existing algorithms.

CHAPTER 3.

OBJECTIVE & SCOPE OF THE PROJECT

3.1 Objective

- To design a robust method for image manipulation that could effectively conceal the content of digital images while enabling accurate reconstruction.
- To exploit the inherent advantages of Binary Tree and Hash Table data structures to achieve efficient image scrambling and descrambling.
- To Implement the developed algorithms and conduct comprehensive evaluations to assess their performance.
- To Design an algorithm capable of accurately reconstructing the original image from its scrambled counterpart, thus ensuring fidelity in the descrambling process.

3.2 Scope of the Project

- Explore the integration of machine learning for automatic detection and classification of sensitive image content, enabling targeted scrambling strategies based on content analysis.
- Extend techniques to video and audio data to address broader security concerns in multimedia communication.
- Develop strategies that dynamically adjust scrambling parameters based on image content to maximize effectiveness and minimize visual distortion.
- Project currently handles grayscale images, explore ways to extend it to support multi-channel images with color information (e.g., RGB or RGBA).
- Integrate encryption techniques with the scrambling process to provide an additional layer of protection for highly sensitive images.
- Implement robust error handling mechanisms to gracefully handle potential issues like invalid image formats, corrupted data, or unexpected user inputs.

CHAPTER 4.

EXPERIMENTAL SETUP & PROPOSED METHOD

4.1 Requirements

Table 1: Requirements of the Projects.

Software Requirements	Hardware Requirements
Operating system: Windows (10 or above) or Linux or MAC	Processor: Intel core i5
Programming Language: MATLAB	Hard disk:10 GB minimum
IDE: R2023b Update 6(23.2.0.2485118)	RAM:256 MB or more

Table 1, outlines the software and hardware needed to run a particular computer program. The software side specifies a compatible operating system (Windows, Linux, or Mac), the programming language (MATLAB), and a specific version of an integrated development environment (IDE) for creating the program. On the hardware side, it details the minimum processor (Intel Core i5), hard drive space (10 GB), and random-access memory (256 MB).

The Image Processing Toolbox in MATLAB which used for project offers a comprehensive set of functions for manipulating and analyzing digital images. It includes tools for tasks such as image enhancement, filtering, segmentation, and feature extraction. This toolbox is essential for various applications, including medical imaging, remote sensing, and computer vision. Users can perform complex operations on images efficiently using built-in algorithms and functions. Additionally, the toolbox provides a user-friendly environment for developing custom image processing algorithms and applications.

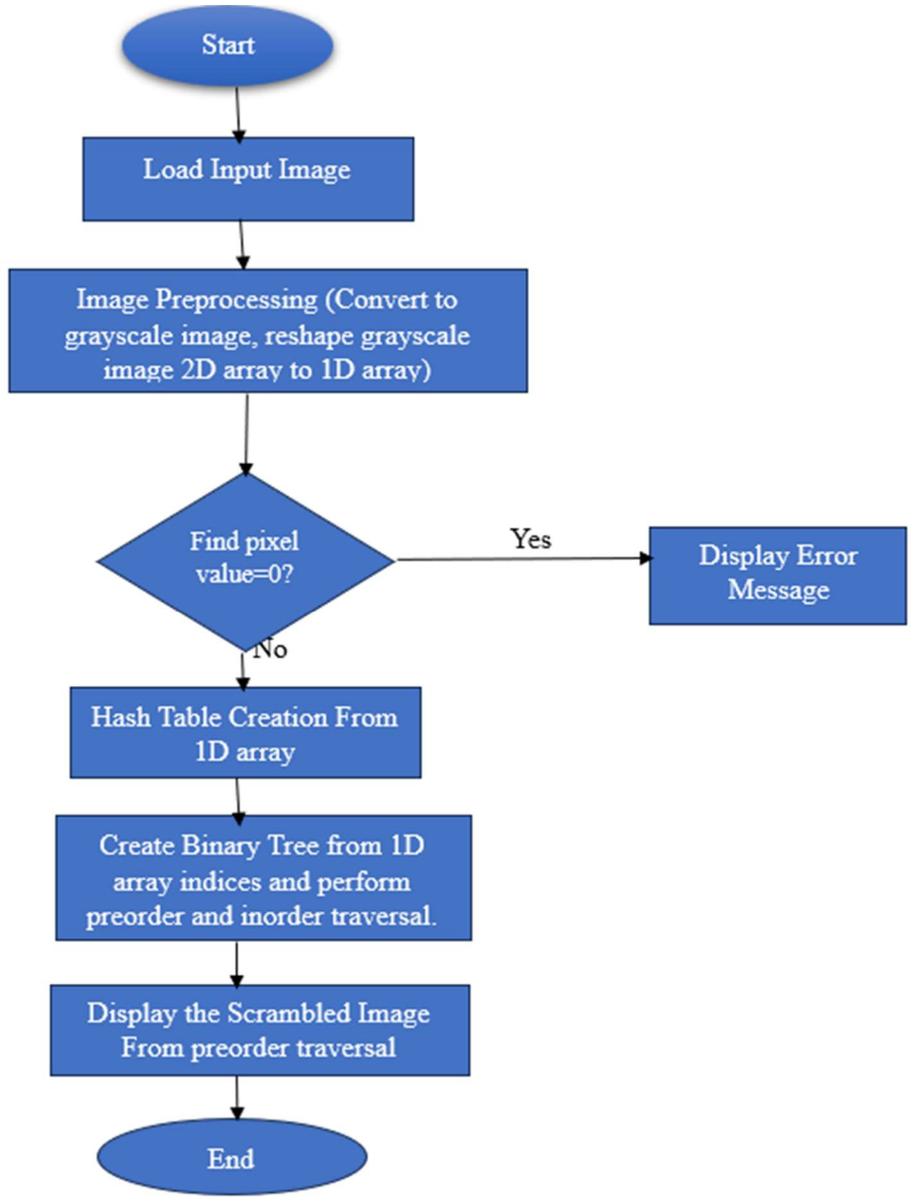


Fig.1 Proposed Model for Scrambling.

In Fig.1 Proposed Model divides the Scrambling work process into different stages which includes Image Loading, Image Preprocessing, Hash Table and Binary Tree generation etc.

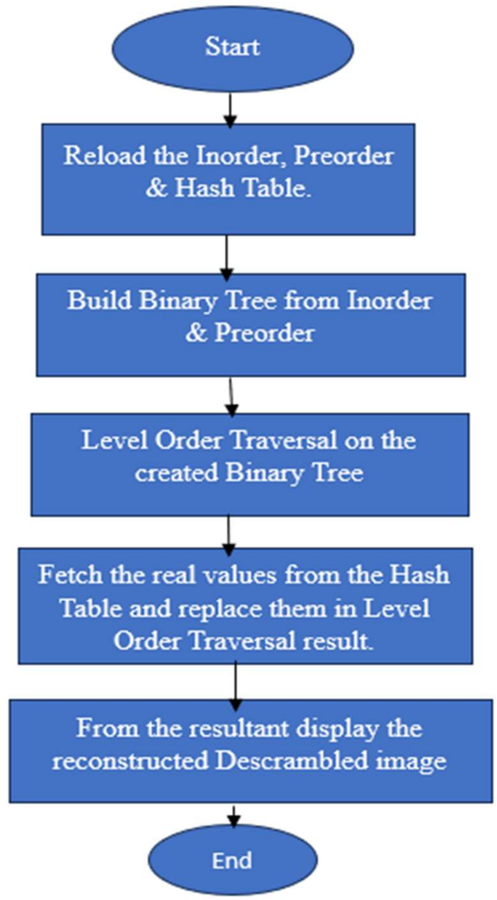


Fig.2 Proposed Model for Descrambling.

In Fig.2 Proposed Model divides the Descrambling work process into different stages which includes reloading inorder, preorder and hash Table, Binary Tree construction and level order traversal etc.

<u>Proposed Image Encryption Algorithm:</u>

Input: Input Image($\boldsymbol{S}$).

Output: Scrambled Image ($\boldsymbol{SS1}$).

Begin

1. **Read the image(S)** and Convert the **RGB image** to grayscale using "**rgb2gray**". And then **Reshape** the **2D grayscale image** into a **1D array** using reshape.
2. Check if the 1D array contains any zeros. If zeros are present, display an error message indicating the image cannot be scrambled and exit the program; otherwise, proceed to step 3.
3. Initialize an empty hash table with 255 cells.
    - Iterates through **each pixel value (1-255):**

I. Find occurrences and either create a new entry with occurrences if cell empty or chain with linked list if collision.

4. Create a binary tree using recursive helper function **construct_tree_helper**.
    - Helper takes 1D array, start/end indices.
    - **If start > end**, return empty node.
    - **Else:** Create node with value at start. call helper for left child and right child.
5. Perform an **in-order traversal** of the binary tree with a recursive function **inorder**.
6. Perform a **pre-order traversal** of the binary tree with a recursive function **preorder**.
7. **Reshape the in-order and pre-order traversal** lists back to the original 2D image dimensions using reshape and save the reshaped arrays as separate **“. MAT”** as a key.

Return scrambled image SS1 from reshaped 2D pre-order traversal.

End

<u>Proposed Image Decryption Algorithm:</u>

Input:

a) $\boldsymbol{inorder}$**:** 2D array representing a structure in inorder traversal.
b) $\boldsymbol{preorder}$**:** 2D array representing the same structure in preorder traversal.
c) $\boldsymbol{hash\_Table}$**:** Hash table for efficient value lookups.

Output: $\boldsymbol{regenrate\_2D}$**:** Regenerated 2D array to its original grayscale Image.

Begin

1. Load $\boldsymbol{inorder}$, $\boldsymbol{preorder}$, and $\boldsymbol{hash\_Table}$ from their respective **“MAT”-**files.
2. Create an empty binary tree and **Call buildBinaryTree(inorder, preorder)** to recursively construct the tree:
    - **If** preorder is empty **then** return an **empty node**.
    - **Else:** extract root data, find root's index in inorder to split into left/right subtrees, recursively build binary tree on them, and set current node's children accordingly.
3. **Level-Order Traversal Binary Tree:**
    - Create an **empty queue** and **Enqueue** the **root** node.
    - While the queue is not empty, dequeue a node, add its data to the result list, and enqueue its left and right children if they exist.
4. **Hash Table Lookups:**
    - Iterate through each element in the result, call search **(hash_Table, element)** to find the corresponding value in the hash table, and replace the element in the result with the retrieved value.
5. **Reshape** result back into a **2D array (**$\boldsymbol{regenrate\_2D}$**)**.

Return $regenerate\_2D$

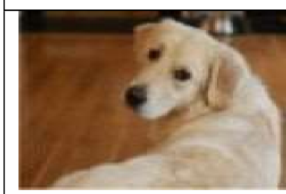
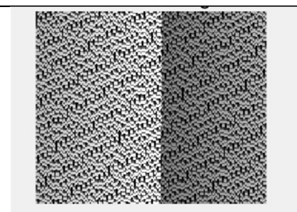
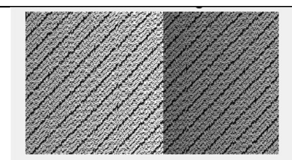
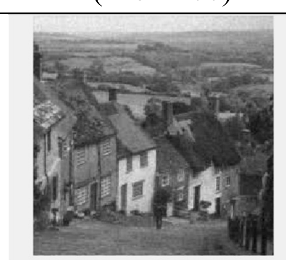
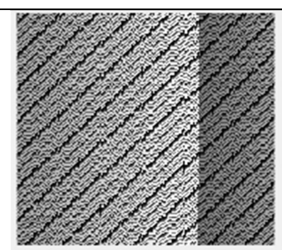
End

CHAPTER 5.

EXPERIMENTAL RESULTS

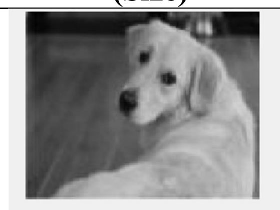
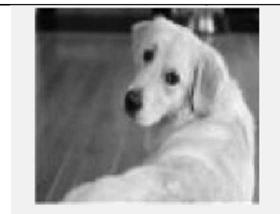
In Table 2, we tested 3 images using proposed scrambling method. According to the expected result the value of PSNR should always be lesser as possible as and SSIM should be lesser than 1. The best result provided by “Lenna.jpg” image and the value of PSNR is 3.87 dB, and SSIM 0.32003 which is shows that the scrambled image in not identical with the input Grayscale image.

Table 2: Result Analysis between Grayscale & Scrambled Images

Original Image (Name)	**Input Grayscale Image (Size)**	**Scrambled Image**	**SSIM**	**PSNR (dB)**
 Dog.jpg	 (150 X 150)		0.31387	4.51
 Lenna.jpg	 (225X400)		0.32003	3.87
 Goldhill.jpeg	 (225X225)		0.224903	4.54

In Table 3, we tested 3 images using proposed Descrambling method. According to the expected result the value of PSNR should always be greater as possible as and SSIM should be equal to 1. All the images have PSNR Inf and SSIM 1 which shows that the Descrambled image in identical with the input Grayscale image.

Table 3: Result Analysis Between Grayscale & Descrambled Image

Original Image (Name)	**Input Grayscale Image (Size)**	**Descrambled Image**	**SSIM**	**PSNR (dB)**
 Dog.jpg	 (150 X 150)		1	Inf
 Lenna.jpg	 (225X400)		1	Inf
 Goldhill.jpeg	 (225X225)		1	Inf

CHAPTER 6.

CONCLUSION & FUTURE SCOPE

6.1 Conclusion

- This project successfully developed a novel image scrambling and descrambling technique utilizing binary trees and hash tables.
- The method effectively rearranges pixel intensities within a binary tree structure, achieving a level of image concealment.
- The use of hash tables facilitates efficient retrieval of original pixel indices during descrambling, leading to accurate image reconstruction.
- By integrating advanced data structures and image processing techniques, we have contributed to the field of image security and privacy, providing a practical solution for safeguarding sensitive visual content.

6.2 Future Scope

- Continuously optimize the algorithm for speed and efficiency, aiming to minimize computational time and memory usage without compromising scrambling quality.
- Explore the integration of deep learning techniques, such as convolutional neural networks (CNNs), for enhancing the image scrambling algorithm. Deep learning models can learn complex patterns and representations from data, potentially leading to more effective scrambling strategies and improved security measures.
- Design and develop a user-friendly graphical user interface (GUI) for the image scrambling algorithm. The GUI would allow users to easily input images, adjust scrambling parameters, visualize scrambled outputs, and interact with various functionalities of the algorithm, enhancing usability and accessibility for a wider audience.
- Implement robust error handling mechanisms to gracefully handle potential issues like invalid image formats, corrupted data, or unexpected user inputs.

REFERENCES

[1]. Wu, Y., Xiang, Y., Guo, Y., Tang, J., & Yin, Z. (2020). An improved reversible data hiding in encrypted images using parametric binary tree labeling. *IEEE Transactions on Multimedia*, *22*(8), 1929–1938.

[**2**]**.** Abbas Fadhil Al-Husainy, M., A. A. Al-Sewadi, H., & Al-Shargabi, B. (2022). Image encryption using a binary search tree structure-based key. *International Journal of Computing and Digital Systems*, *12*(1), 823–836.

[3]. Gopalakrishnan t. (2018). *Image encryption using multiple chaotic maps in hybrid domain* [doctor of philosophy, anna university].

[4]. Wang, X., & Liu, L. (2020). Image encryption based on hash table scrambling and DNA substitution. *IEEE Access*, *8*, 68533–68547.

[5]. Al-Husainy, M. A., & Al-Sewadi, H. A. (2019). Implementing binary search tree concept for image cryptography. *International Journal of Advanced Science and Technology*, *130*, 21–32.

[6]. K. Chitra. (2019). *Enhancement of image security using improved visual cryptography visual cryptography and digital watermarking techniques* [doctor of philosophy, Pondicherry university].