Image Representation through Pixel-Based Encoding

M.Sheerin Banu
Department of Information Technology, R.M.K. Engineering College
Pariti Venkata Raghavendra, Shaik Basheerulla, Kothapalli Krishna Chaithanya
Department of Information Technology, R.M.K. Engineering College

Abstract:Now a days, in every communication channel there is a necessity of transmission of messages securely from sender to the authentic receiver. In recent years, for different information transfer systems, a number of data encryption techniques has been evolved. Several encryption approaches based on permutation have been proposed by various researchers. An Image Encryption algorithm is useful to protect digital images from cyber-attacks. A pixels-based keys encryption algorithm uses a XOR operator to the rows and columns of the image using two secret keys. This Algorithm results in lossless image encryption, and it is highly secured as two keys are associated with the encryption of each and every pixel of the image. The decryption of the image is not possible without using those keys. The proposed encryption method is simple and ensures the security of the encrypted images.

Keywords— Image Encryption, Image Processing, Scrambling

INTRODUCTION

In recent years, along with the rapid promotion and popularization of network technology and digital communication technology in the world, digital images, and digital video-based digital images have become an important medium for information storage and transmission in the computer network in the civil and military fields. Image Encryption is the process of encoding messages in such a way that eavesdroppers or hackers cannot read it. The security of digital images has attracted more attention recently, and many different image encryption methods have been proposed to enhance the security of these images**.** Image encryption techniques try to convert an image to another one that is hard to understand. On the other hand, image decryption retrieves the original image from the encrypted one. Image encryption techniques try to convert an image to another one that is hard to understand. On the other hand, image decryption retrieves the original image from the encrypted one. There are several methods of scrambling technique, as an example block based scrambling, pixel scrambling etc. Utilizing scrambling strategies numerous researchers had stated many algorithms to build cipher images. A pixels based image encryption and decryption algorithm are designed and implemented to provide confidentiality and security in transmission of the image-based data as well as in storage. Since the pixel of image is highly correlated to their neighboring pixels. Due to this strong correlation any pixel can be practically predicted from a value of its neighbors. So, there is a need of a technique that can shuffle the pixels to reduce the correlation between the neighbor pixels. Hence, we used Scrambling technique that Shuffles the pixels of image. This Scrambled image is called transformed image. Then, for further secure encryption we use XOR operator. The total size of key in our algorithm is 32 bit long which proves to be strong enough.

EXISTING SYSTEM

Although, extensive research is carrying on image Encryption and Decryption, for successful and loss less image quality, gray scale images are being encrypted for better outcomes. Even though this system exists for further better results, encryption of different image types, color images have taken place.

PROPOSED SYSTEM

Here, in our proposed system a pixels-based keys encryption algorithm is used. This encryption algorithm uses a XOR operator to the rows and columns of the image using two secret keys For better understanding,

a) As we know, each pixel in a color image would contain R, G, B values. We split them into 3 matrices and store their values.

b) Then we move onto generate two key arrays. The first key array which is a one-dimensional(1-D) array named as Kr. The second key array which is also a one-dimensional(1-D) array named as Kc. These arrays consist of values ranging from 0 to 255 as pixels too range from 0 to 255. The size of these arrays depends on the M x N Image.

c) Next, we shuffle each row and column in the R, G, B arrays to get the scrambled image. In order to do this, we go through each row in R, G, B matrices, find out its sum, if the sum of the row is even, we rotate the row clockwise by corresponding value in Kr array and if the sum of the row is odd, we rotate the row anticlockwise by corresponding value in Kc array. Similarly, we go through each column, find out its sum, if it is even, we upshift by Kc times and if the sum is odd, we downshift by Kc times.

Now, we have the scrambled image. We perform XOR operation on the scrambled image with Kr and Kc. Firstly, we iterate each row in the R, G, B arrays, if the corresponding row is odd, we replace each element present in the row with, XOR of the current element in the row and Kc value of the current iteration. If the row is even, we replace each element present in the row with, we rotate the corresponding Kc value (flip the bitwise representation of kc value)

a) and perform XOR of the current element in the row and rotated Kc value of the current iteration.

b) Similarly, we iterate each column in the R, G, B arrays, if the corresponding column is even, we replace each

c) element present in the column with, XOR of the current element in the column and Kr value of the current iteration. If the column is odd, we replace each element present in the column with, we rotate the corresponding Kr value (flip the bitwise representation of kc value) and perform XOR of the current element in the row and rotated Kr value of the current iteration.

d) Finally, we have the Encrypted the Image.

e) To decrypt the image, first we perform XOR operation on the Encrypted Image's R, G, B arrays with the same Kr and Kc values in the same way, as the Encryption process. Then we back rotate using the same logic as above mentioned.

Processes for encrypting and decrypting the image:

- **User Interface**
- **File Existence Check**
- **Image Encryption**
- **Encrypted Image**
- **Generation of Keys (Kr and Kc)**
- **Decrypted Image**

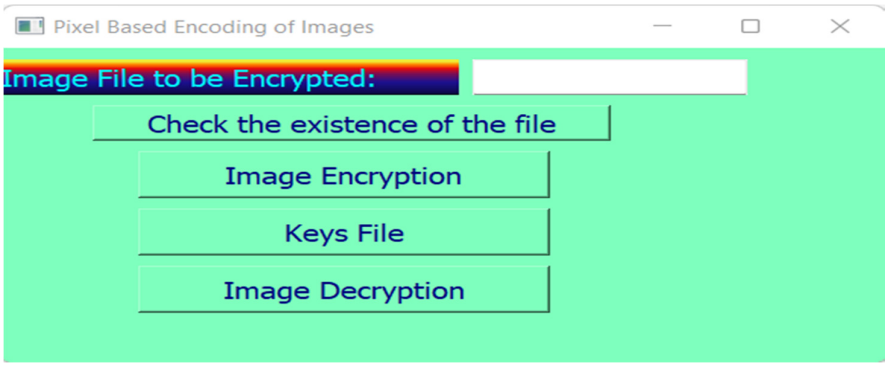

Fig. 1 User Interface

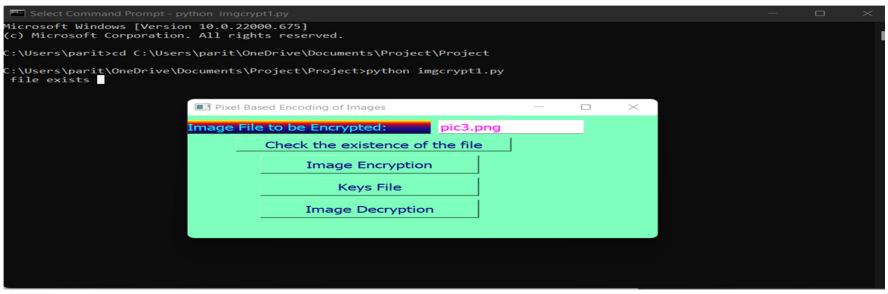

Fig. 2 File Existence Check

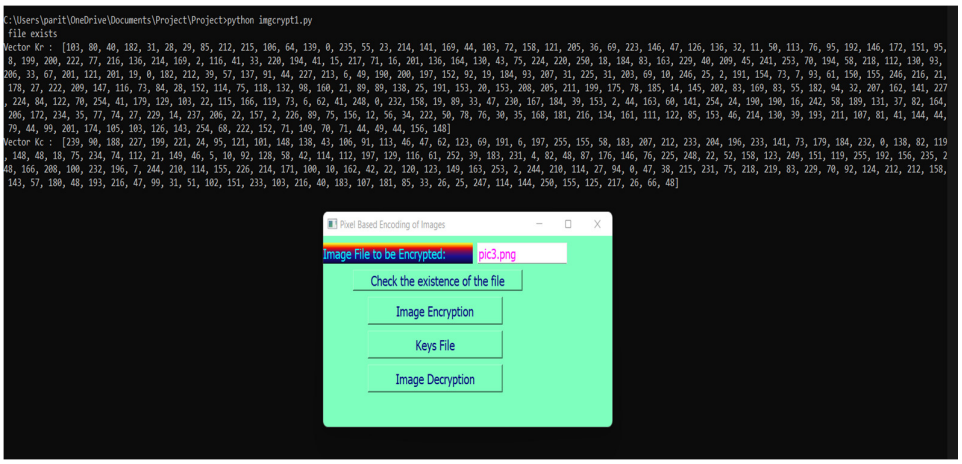

Fig. 3 Image Encryption

Fig.4 Encrypted Image

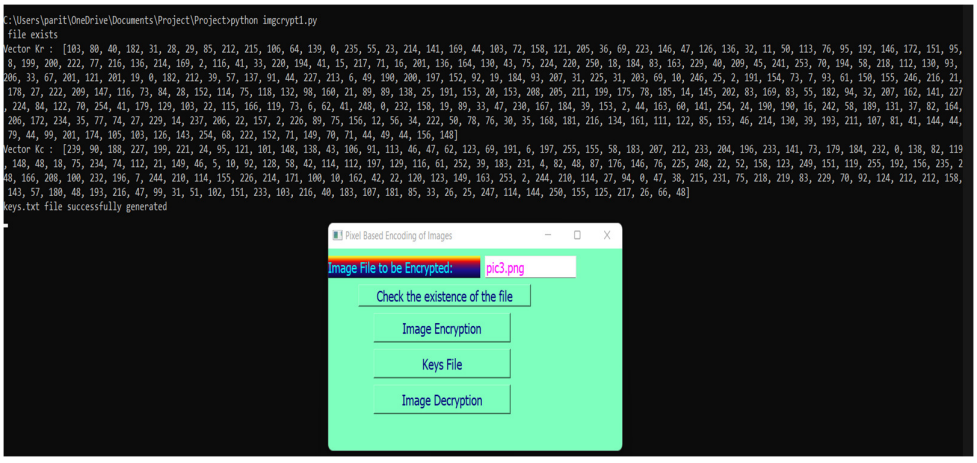

Fig. 5 Generation of Keys (Kr and Kc)

Fig. 6 Decrypted Image

SYSTEM DESIGN

A. Use Case Diagram

A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

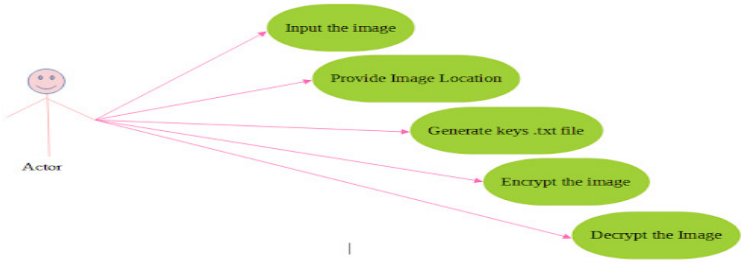

Fig. 7 Use Case Diagram

B. Activity Diagram

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow from one activity to another activity.

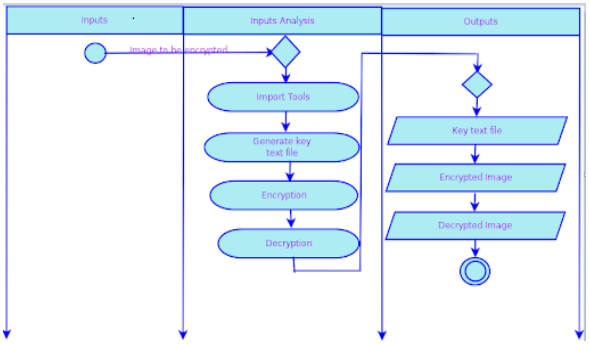

Fig. 8 Activity Diagram

C. Sequence Diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence.

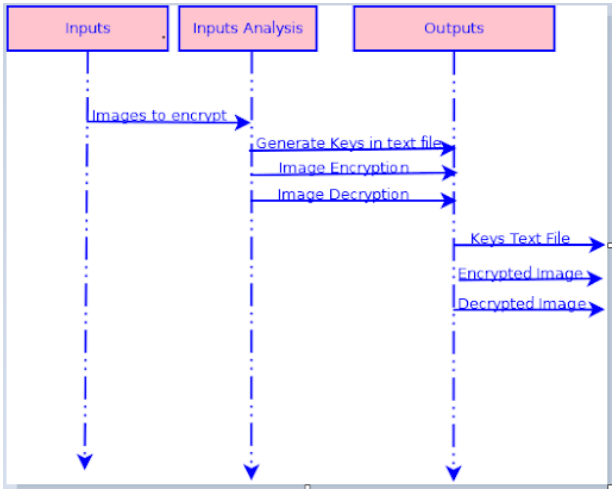

Fig. 9 Sequence Diagram

D. Data Flow Diagram

A data-flow diagram is **a way of representing a flow of data through a process or a system** (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops.

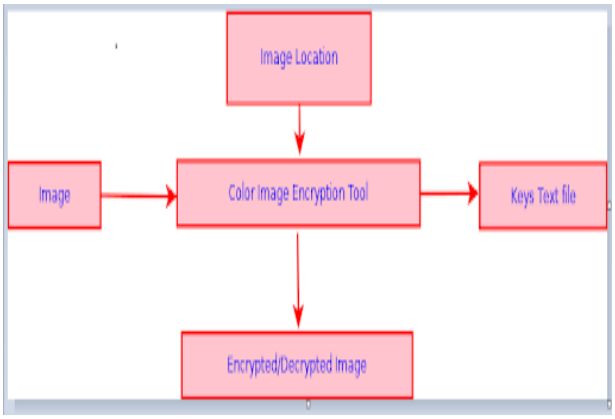

Fig. 10 Data Flow Diagram

CONCLUSION

In this paper, we found that XOR cipher is an important tool to encrypt an image. When, we use XOR cipher to encrypt an image then the randomness of pixels of an original image increase. If randomness is more, we can say that the image is more secure**.**

The proposed encryption algorithm can ensure multiple criteria such as lossless, maximum distortion, maximum performance and maximum speed. The proposed encryption method in this study has been tested on different color images and showed good results.

The security keys which are used to encrypt and decrypt the original images are same. In this process to encrypt an image XOR operation is performed between original image and security key. Then to decrypt the image again XOR operation is performed between encrypted image and Security key. So, decryption using security key is totally dependent on the result after encryption. So, we can conclude that this process increases security of an images.

REFERENCES

[1] Aesha N. Elghandour, Ahmed M. Salah, Yasser A. Elmasry, And Abdelrahman A. Karawia (Member, IEEE): An Image Encryption Algorithm Based on Bisection Method and One-Dimensional Piecewise Chaotic Map, IEEE Access

[2] M. Kaur, D. Singh, K. Sun, U. Rawat, Color image encryption using non-dominated sorting genetic algorithm with local chaotic search based 5D chaotic map', Future Generation Computer Systems.

[3] https://iopscience.iop.org/book/978-0-7503-2220-1/chapter/bk978-0-7503-2220-1ch1

[4] https://github.com/baoboa/pyqt5/blob/master/pyuic/uic/pyuic.py

[5] https://riverbankcomputing.com/software/pyqt/intro

[6] https://www.tutorialspoint.com/pyqt5/index.htm

[7] https://www.w3schools.com/python/numpy/default.asp