XOR-Based Security and Affine Transformation for Image Encryption

B. Nagamani[1] Reddy, A. Hariprasad[2], Somala Rama Kishore[3]
[1] *Assistant Professor, Department of AIML, Geetanjali College of Engineering and Technology*
[2] *Professors, Department of CSE, Geetanjali College of Engineering and Technology*
[3]*Associate Professor, Dept.of.ECE, CMR Engineering College*

ABSTRACT

An appropriate technique for safeguarding image data is image encryption. Text and image data each have special characteristics. For text data, the available encryption algorithms work well. Media information may not be ideal for them. The pixels in natural images actually have a strong correlation with one another. Any pixel can be practically predicted from the values of its neighbours because of this strong correlation. In this paper, we suggest an encryption technique based on location transformation. We use four 8-bit keys and the affine transform technique to redistribute the pixel values to different locations. Following transformation, the image is split into blocks of two pixels by two pixels, each of which is encrypted using an XOR operation with four 8-bit keys. Our algorithm uses a 64-bit total key size, which turns out to be sufficiently strong. The experimental findings demonstrated that the correlation between pixel values was significantly higher following the affine transform.

Keywords: Image Correlation, Image encryption, Image histogram, Affine transform, Symmetric key encryption

INTRODUCTION

The trading of electronic information trade is expanding quickly. With the quick development of electronic information trade, the unauthorised information access is likewise expanding. Information security is becoming increasingly important for data transmission and storage in order to prevent this unauthorized access. Images are used in all facets of life and are a very common source of information. The security of picture information from unauthorised access is extremely fundamental. Encryption strategies [1, 2] are extremely valuable instruments to safeguard restricted Intel. They use a key to transform the secret information into an unintelligible form in order to protect it. Using certain keys, the encrypted data must be transformed back into the original data in order to be recovered. The encryption algorithm can be divided into two groups according to the key. The two types of key encryption are (i) symmetric and (ii) asymmetric. While asymmetric key encryption algorithms use separate keys

for encryption and decryption, symmetric key encryption algorithms use the same key for both processes.

Because of its extremely high computational costs, asymmetric key algorithms are frequently unfeasible for multimedia data. Algorithms for symmetric key encryption are relatively less expensive and can be applied to multimedia data. But nevertheless, multimedia data has completely different characteristics than textual information. Unlike all multimedia data, which has a great deal of redundancy, text data has none. The pixel esteem of an area is profoundly corresponded to upsides of its adjoining pixels. In a similar manner, a sound sample is connected with both its prior and subsequent samples. Any common encryption algorithm can be attacked using this correlation. Since, if one can figure out pixel esteem at an area or one sound example, then they can foresee the benefits of adjoining pixels or next sound example with sensible exactness.

Text data is encrypted using the majority of the available algorithms, including DES, AES [1], RSA [1], and IDEA [1]. DES [1], AES [I], RSA [I], and IDEA [I] can all achieve high security, but they might not be appropriate for encrypting images and videos because of their inherent characteristics, which include high redundancy and large data sizes, which necessitate the use of different encryption algorithms [9,10]. Three main categories of image encryption algorithms can be distinguished: (i) algorithms based on position permutation [6, 7], (ii) algorithms based on value transformation [3, 4, 5, 8], and (iii) algorithms based on visual transformation [6]. The LOGXoRP can extract effective texture (edge) features as compared to LBP and LGP [14]. They separated the original image into blocks of four pixels by four pixels, which they then rearranged using a predetermined permutation process to create a permuted image. The RijnDael algorithm was then used to encrypt the permuted image. Their findings demonstrated that applying the combination technique considerably reduced the correlation between image elements. Chaotic maps serve as the foundation for many encryption algorithms [11].He created two-dimensional maps that were invertible and chaotic. New methods for symmetric block encryption. His method works well for encrypting large amounts of data, including digital photos. An image encryption algorithm based on a binary sequence produced from a chaotic system was proposed by Guo and Yen [7]. They used the generated binary sequence to jumble an image. This algorithm has no distortion, high security, and minimal computational complexity.

In this paper, we propose a two-phase encryption and decryption algorithms that is based on shuffling the image pixels using affine transform and they encrypting the resulting image using XOR operation. We have used a key of 64-bit length which is quite good for practical purposes. The affine transform fractures the correlation between adjacent pixels of an image. Affine cipher

is one-to-one mapping, that is, a symbol in the plaintext can be transformed to a unique symbol in the cipher text. In Affine cipher, the relationship between the plaintext P and the cipher text C is given in equations 1 and 2.

$$C = (K_0 + K_1 * P) \bmod N \quad \text{.....(1)}$$

$$P = (C + (-K_0)) * K_1^{-1}) \bmod N \quad \text{.....(2)}$$

Where gcd (K_1, N) =1, K_1^{-1} is multiplicative inverse of K_1 and ($-K_0$) is addictive inverse of K_0. The article is structured as follows: In section 2 we propose the encryption and decryption algorithms. Section 2 also describes the key selection process. The experimental results are presented in section 3. We conclude the paper with a conversation on current work and a few bearings to future works in area 4.

OBJECTIVES

The primary objective of this research is to develop a secure and efficient image encryption technique that effectively protects image data from unauthorized access. Traditional encryption algorithms designed for text data may not be suitable for images due to the strong correlation between neighboring pixels. Therefore, this study aims to implement a location transformation-based encryption method that redistributes pixel values, reducing their correlation and enhancing security.

The proposed method utilizes the affine transformation technique to alter the positions of pixel values within the image. Following this transformation, the image is divided into small blocks and each block undergoes an encryption process using an XOR operation with four 8-bit keys. By employing a 64-bit total key size, the algorithm ensures a robust encryption scheme that is resilient to various cryptographic attacks.

Additionally, the study aims to evaluate the effectiveness of the proposed technique through experimental analysis. A key focus is on assessing the correlation between pixel values before and after encryption, ensuring a significant reduction in predictability. By achieving these objectives, this research contributes to enhancing image security, making it suitable for applications in secure communication, digital watermarking, and confidential image storage.

METHODS

In this section, we propose a symmetric two-phase encryption algorithm. We used a 64-bit symmetric key. The 64-bit key is divided into eight subkeys K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7 and K_8 each with 8 bits. The keys are chosen such that the first subkey is relatively prime, matches the width of the image, and the fourth subkey is relatively prime to the height of the image, i.e., gcd (K_1, M) =1 and gcd(K_3,N)=1. Reasons for choosing gcd(K_1, M) = 1 (wd gcd K_3, f = 1) is the transformed one.

Coordinates by gcd K, ... 1 = 1 i and gcd K;, go from 1 to N. = 1 is unique within the range 1 and M. If subkey is not prime to the image height and width.

Transformation process assigns multiple locations to the same target. For example, if K_1 = 32 and K_2 = 6, then 5 37 will be assigned to the same position as follows:

(5*32 + 6) % 256 = 166

And (37 * 32 + 6) %256 = 166

Initially four sub-keys using the affine cipher algorithm, K_0, K_1, K_2 and K_3 are used to change the location of the image's pixel values. Using a straightforward XOR operation, the next four keys K_4, K_5, K_6, K_7 are used for the second level of encryption. Due to the strong correlation between adjacent pixels in image data, we employ a location transformation of the image's pixel values. Any encryption algorithm's weak point is this strong correlation. Anyone who knows the value of a pixel can use some prediction techniques to reasonably predict the values of the neighbouring pixels. So, first, we use an affine transformation to move the image pixels to new locations, thereby breaking the correlation between the image pixels. The detailed implementation of the affine transformation is described by Equation 3 and 4. Assume that we have a M x N image with pixel locations that range from (1, 1) to (M, N). In the secret image, the pixel location (x, y) where and x ϵ {0, 1, 2... M-1} and y ϵ {0, 1, 2 ....N-1} are changed to a new location (x`, y`) by

$x` = (K_0 + K_1 * x) \bmod M$.....(3)

$y` = (K_2 + K_3 * y) \bmod N$.....(4)

The image has been transformed and decayed into $\frac{M}{2} * \frac{N}{2}$ number of 2*2 blocks. The pixels in each block are encrypted using sub-keys K_4, K_5, K_6 , and K_7. Algorithm 1 provides a detailed description of the encryption procedure.

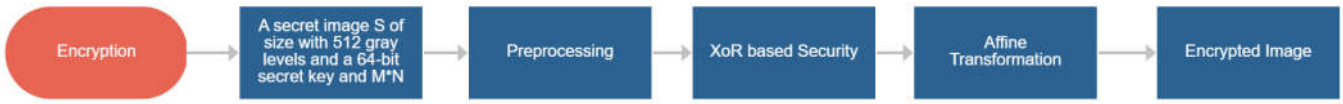

Figure 1: Flowchart

Algorithm 1: Encryption Method

Input: A secret image S of size with 512 gray levels and a 64-bit secret key and M*N.

Output: M*N cipher image with gray scale of 512.

Steps: Encryption Approach

Preprocessing:

- Convert the 512 × 512 image into a numerical matrix (grayscale).
- Normalize pixel values (0–255) if required.

XOR-Based Security:

- Use a secret key (a matrix of the same size or a pseudo-random key sequence).
- Perform XOR between the image pixels and the key

$$C(i,j)=P(i,j) \oplus K(i,j)$$

Where P is the original image, K is the key, and C is the cipher image.

Affine Transformation:

- Apply an affine transformation using a transformation matrix:

$$\begin{bmatrix} \bar{x'} \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

Where (a, b, c, d, e, f) are transformation parameters.

- This scrambles pixel locations, adding confusion.

Final Encrypted Image:

- The XOR operation provides diffusion (bit-level security), while affine transformation enhances confusion (pixel scrambling).
- The final encrypted image should be unrecognizable.

Figure 2: Cameraman's 16-bit grayscale image

Decryption Method

Figure 1: Cameraman's 16-bit grayscale image initially, the cipher image is broken down into $\frac{M}{2} * \frac{N}{2}$ number of 2*2 blocks. Using four least significant bit subkeys K_4, K_5, K_6, and K_7 and an XOR operation, each pixel in each block is decrypted.

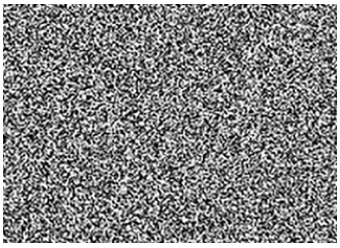

Figure 3: Affine transformed image of Cameraman

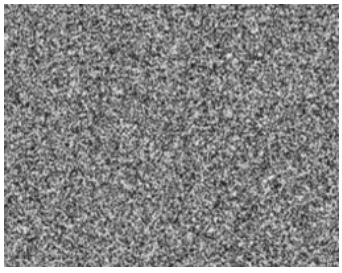

Figure 4: An image of Cameraman that has been affine transformed and XOR encrypted

Algorithm 2: Decryption Method

Input: A 64-bit secret key and M*N Cipher Image C

Output: A secret image S of M*N.

Steps:

Decryption Approach

- Reverse affine transformation using the inverse matrix.
- XOR decryption using the same key:

  $P(i,j) = C(i,j) \oplus K(i,j)$
- End.

RESULTS

Ten 512 by 512 8-bit grayscale photos have been utilized. One such picture of Figure 2 exhibits 512 x 512 8-bit grayscale pictures of Cameraman. Figure 2 exhibits the affinely converted image, while Figure 3 displays the fmal encrypted image. Figures 4, 5, and 6 display the histograms of the original image, the affine modified image, and the XOR encrypted image.

The histogram of figures 5 and 6 shows that the affine cipher transformation moves the pixel values but leaves them unchanged. The average correlation of adjacent pixel values following the affine transformation and the XOR operations is presented in Table 1. It is evident from Table 1 that the correlation between neighboring pixel values is approximately 0.9 after the affine transformation and about 0.149 following the XOR operation.

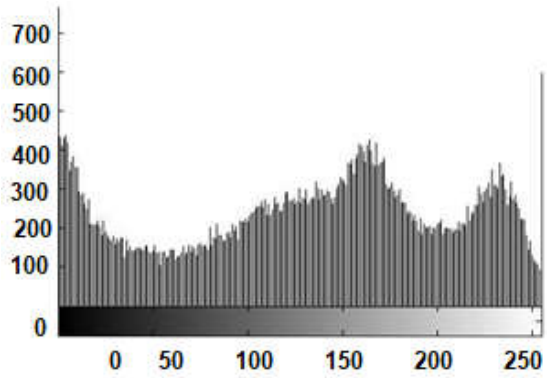

Figure 5: Affinity transformed image histogram of Cameraman

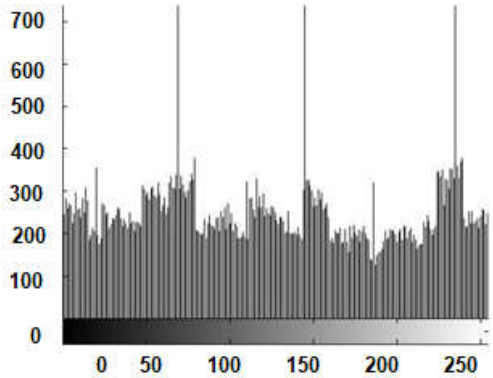

Figure 6: Cameraman's histogram after an affine transformation using XOR encryption

Average correlation between pixel values is shown in **Table 1.**

Image Name	Correlation after affine transformation	Correlation after XOR
Cameraman	0.968	0.512
X Plane	0.923	0.493
Air Plane	0.9811	0.2913

DISCUSSION

In this article, we proposed a symmetric key image encryption method that uses four 8-bit subkeys to first jumble the pixel locations before XORing the chosen 8-bit key to encrypt the pixel values. Affine cipher techniques are used for the scrambling process, which breaks the correlations between nearby pixels and renders the image unidentifiable. The image becomes completely meaningless as a result of the XOR operation changing the pixel values. Although it offers sufficient security, the encryption and decryption procedure is easy enough to perform on any large image or video file. In order to further raise the security level, the authors are now working on randomizing the application of keys.

REFRENCES

[1] W. Stallings, Cryptography and Network Security principles and practices, 3rd ed., Pearson Education, 2003.

[2] H. EI-din H. Ahmed, M. K Hamdy, and O. S. Farag Allah, "Encryption quality analysis of the RC5 block cipher algorithm for digital images," Optical Engineering, Vol. 45, Issue 10107003, 2006. DOI:10.1117/1.2358991

[3] Aloha Sinha, Kehar Singh, "A technique for image encryption using digital signature", Optics Communications, Vol-2 I 8 (2203),229-234. https://doi.org/10.1016/S0030-4018(03)01261-6

[4] S.S.Maniccam, N.G. Bourbakis, "Lossless image compression and encryption using SCAN", Pattern Recognition 34 (2001), 1229-1245. https://doi.org/10.1016/S0031-3203(00)00062-5

[5) Chin-Chen Chang, Min-Shian Hwang, Tung-Shou Chen, "A new encryption algorithm for image cryptosystems", The Journal of Systems and Software 58 (200 I), 83-9 I. DOI:10.1016/S0164-1212(01)00029-2

[6] Jiun-In Guo, Jui-Cheng Yen, "A new mirror-like image encryption algorithm and its VLSI architecture", Pattern Recognition and Image Analysis, vol.IO, no.2, pp.236-247, 2000. DOI:10.1109/SIPS.1999.822348

[7] Haojiang Gao, Yisheng Zhang, Shuyun Liang, Dequn Li "A New Chaotic Image Encryption Algorithm," Chaos, Solitons & Fractals, Volume 29, Issue 2, July 2006, Pages 393-399. https://doi.org/10.1016/j.chaos.2005.08.110

[8] Shuqun Zhang and Mohammed A Karim, " Color image encryption using double random phase encoding", Microwave And Optical Technology Letters Vol. 21, No. 5, June 5 1999,318-322. https://doi.org/10.1002/(SICI)1098-2760(19990605)21:5<318::AID-MOP4>3.0.CO;2-A

[9] Abhinav Gupta1,and Aayush Gupta2, " A New Technique of Image Encryption using Modified AES Algorithm," International Journal of Multidisciplinary Innovative Research. ISSN: 2583-0228 Volume 1, Number 1 (Jul' 2021) pp. 34-43. DOI:10.14569/IJACSA.2017.080212

[10] S. Changgui, B. K Bharat, "An efficient MPEG video encryption algorithm," Proceedings of the symposium on reliable distributed systems", 1998, pp. 38 I -386. DOI: 10.1109/RELDIS.1998.740527

[11] J. Cheng; J.1. Guo, "A new chaotic key-based design for image encryption and decryption," The 2000 IEEE International Symposium on Circuits and Systems, volA, no. 4, pp. 49 - 52, May. 2000. DOI: 10.1109/ISCAS.2000.858685

[12] S.Behnia,AAkhshani , S.Ahadpour, H.Mahmodi,A Akha-van, A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps, Physics Letters A 6(2007):39 I -396. DOI:10.1016/j.physleta.2007.01.081

[13] Jiri Fridrich, "Image Encryption Based on Chaotic Maps", Proceeding of IEEE Conference On Systems, Man, and Cybernetics, pp. 1 105-I I 10, 1997. DOI: 10.1109/ICSMC.1997.638097

[14] AH Reddy, N.S.Chandra, "Local Orientation Gradient Xor Patterns: A New Feature Descriptor For Image Indexing And Retrieval". i-manager's Journal on Pattern Recognition, Vol. 2 l No. 4 l December 2015 - February 2016. DOI:10.26634/jpr.2.4.5943