Chaotic Based Lightweight Image Encryption Algorithm for Real-time Application Systems

Aguru Aswani Devi, Attada Venkata Ramana

Abstract: Several real-time applications in military, banking and biometric systems involves images which should be stored and transferred with security. Due to the sensitiveness on initial conditions, randomness and less complexity, chaotic maps are useful and effective for image encryption. The current work involves chaotic based lightweight image encryption algorithm along with lightweight properties is presented, which can be used in real time applications. The proposed cryptosystem can be implemented on Raspberry Pi or on any micro-controller to ensure its real time usability. Several lightweight algorithms with 1D chaotic map were implemented on microcontrollers. The proposed cryptosystem uses 2D chaotic map along with lightweight operations to achieve high randomness in the key stream. The original image, Shuffled image, Encrypted image, Decrypted image and Reverse shuffled image are generated in sequential order to depict the result of crypto system. This work also includes the experimental result analysis to ensure the security levels of the proposed encryption algorithm.

Index terms: Chaos, Image Encryption, Lightweight, Microcontroller, Raspberry Pi

I. INTRODUCTION

Due to the expansion in multimedia applications, it became a necessity to secure multimedia data, especially images. This security can be achieved by image cryptography [11] and image steganography [2]. Cryptography hides the readability of the image through a process called encryption. Steganography hides the image behind a cover, so the image will be invisible. Lightweight cryptography holds the encryption techniques which utilizes less resource so that they can be used for real-time applications [3]. This lightweight property is ensured by light weight operations with low demand for memory and less execution time [8,22]. Gray scale images uses 8-bits to represent its pixel value, so 2^8 different shades are possible in a gray scale image[23]. Because of having a single channel, it is optimal to use gray scale images than color images in the real-time imaging applications. There are various schemes that have been designed for securing digital images.

These schemes based on different aspects of images. In some schemes, the pixel values of image are shuffled by changing their position[24]. In the other, position of pixels will remain same, but data values in original image are replaced by other values which are taken from lookup table called S-box. In Some schemes, both position and data values are altered. Finally, some schemes use chaotic maps or functions to change positions and to substitute with different data values. Among them, Chaos based techniques are more efficient with their security and speed[25]. So, these are preferred for lightweight image encryption. Chaos based techniques are simple with less computational overhead, so that these are suitable for lightweight devices like mobiles and various hand held devices. Various works were carried out using 1D logistic chaotic map and lightweight properties for image encryption on embedded devices. 2D chaotic maps are complex and use more memory than 1D chaotic map, but they generate highly random key stream for encryption [9].

The contributions of the Proposed System are outlined below:

1. Pixels of the input image are shuffled in a zig-zag pattern.
2. A 2 Dimensional Chaotic map called Henon map along with lightweight properties encrypts each pixel of shuffled image. System uses 192-bit symmetric key.
3. Decryption is performed with same chaotic map and symmetric key.
4. Reverse shuffling is performed by implementing the inverse zig-zag pattern of the decrypted image.

II. LITERATURE SURVEY

A lightweight chaotic encryption algorithms for images was presented by Siva Janaki raman, K Thenmozi, et al. [1].It also includes the implementation and analysis on 32-bit microcontroller.

This image encryption algorithm uses 1D logistic chaotic map for generating random keys[16].

It is a 128-bit stream cipher. Before encryption the scrambling of the pixels is performed using an inward spiral pattern. The algorithm encrypts gray scale image with lightweight operations was proposed by Bahrami and Naderi [5].

Another chaotic image encryption scheme was proposed by Ali Soleymani, Md Jan Nordin, et al. with good security analysis [6]. It used Henon map and Arnold Cat Map for encryption. Ciphered image was strengthened by improvement in the randomness of transformation and efficient bit permutation.

Encryption and decryption of the color image with Henon chaotic system along with pixel shuffling was proposed by Asia Mahdi and Naser Alzubaidi [7]. Lightweight properties are not included as this encryption algorithm is not targeted for embedded device. This paper enlightens an attempt that utilizes a zigzag scanning pattern

Revised Manuscript Received on December 22, 2018.

Aguru Aswani Devi, M.Tech Student, CSE, GMR Institute of Technology, India, aswaniaguru@gmail.com
Attada Venkata Ramana, Professor, CSE, GMR Institute of Technology, India, mithun.avr@gmail.com

to upset the correlation among the neighboring pixels of an image without any change in pixel values[26,27]. After this shuffling process, an encryption algorithm with lightweight operations is designed. Two dimensional Henon chaotic map generates the stream of secret keys to encrypt each pixel of gray scale image [10,28].

III. PROPOSED MOTHODOLOGY

The Henon map, the simplest two dimensional chaotic maps, was proposed by M.Henon in 1976. The simplest form of Henon system is the discrete time system that maps a point (x_n, y_n) in this fashion:

$$x_{n+1} = 1 - a{x_n}^2 + y_n \qquad (1)$$

$$y_{n+1} = 0.3 * x_n \qquad (2)$$

For a = [1.07, 1.4] and b=0.3, the Henon function generates random sequence and exhibits chaotic behaviour. Proposed cryptosystem uses 192-bit symmetric key and to encrypt each pixel of the image, a unique key is generated by 2D Henon chaotic map using Eq.1 and Eq.2. This key is generated by the concatenation of six 32-bit inputs in IEEE754 format.

Table 1. IEEE Format of Floating Point Numbers

Significance of the field	Sign	Exponent	Mantissa
Size	1 bit	8 bits	23 bits
Bit position	D31	D30 to D23	D22 to D0

A. Flow of Proposed Algorithm

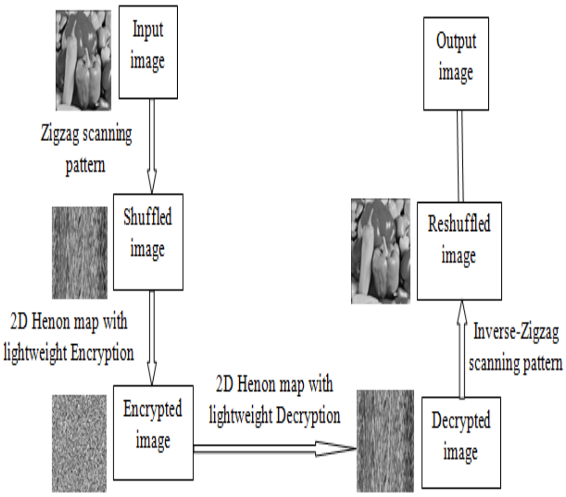
Input image
Zigzag scanning pattern
Shuffled image
2D Henon map with lightweight Encryption
Encrypted image
2D Henon map with lightweight Decryption
Decrypted image
Inverse-Zigzag scanning pattern
Reshuffled image
Output image

Figure 1. Flow of Proposed Cryptosystem

The flow of proposed system includes shuffling, encryption, decryption and reverse shuffling of plain image. Confusion scheme uses zigzag scanning pattern and inverse zigzag pattern for encryption and decryption respectively. Diffusion scheme uses 2D Henon map for key stream generation to encrypt each pixel of image. The proposed method has 4 phases and are narrated in following subsections.

B. Confusion Scheme by Pixel Shuffling

Shuffling the pixels of the image is performed to disturb the correlation among adjacent pixels, without altering the values of the input image pixels to achieve randomness. The strength of the image encryption algorithm is increased if this scrambling is performed before the encryption of image.

In this paper, zigzag scanning pattern of the pixels of input image is performed for scattering the position of pixels to get shuffled image. The plain gray scale image ‘P’ of ‘U’ rows and ‘V’ columns is converted to the shuffled image ‘S’ of same size. After decryption, reverse shuffling is performed by inverse zigzag scanning scheme to get final output image. The Top-Left pixel is the start point of scanning pattern and it ends with Bottom-Right pixel position.

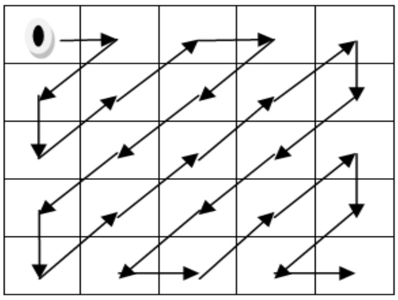

Figure 2. Scanning Pattern of Image Shuffling

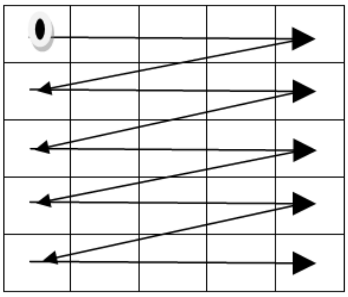

Figure 3. Resulted Pixel Position in Shuffled Image

Shuffling the image pixels before the encryption process results the better ciphered image than encrypting the plain image. So, the shuffled image with zigzag shuffled pattern will be fed into the cryptosystem.

C. Diffusion Scheme by Henon Chaotic Map and Light Weight Operations

Diffusion in encryption process

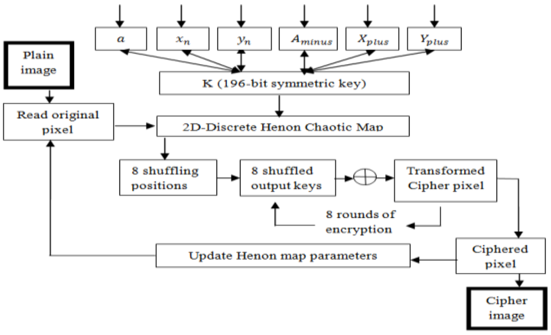

Figure 4. Block Diagram of Diffusion Scheme in Encryption Process

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

D. Proposed encryption algorithm

Step 1: Choose six floating point

inputs$x_n, y_n, A_{minus}, X_{plus}$, and Y_{plus} *such that*

$1.07 < a < 1.4$ *and* $0 < P < 1$

where$P \varepsilon \{ x_n, y_n, A_{minus}, X_{plus}, Y_{plus} \}$.

Step 2: Concatenate the 32-bit IEEE754 representation of these inputs to get 192-bit symmetric key of the cryptosystem.

Step 3: The floating point random numbers x_{n+1}, y_{n+1} of the two-dimensional Henon Chaotic system are calculated by the Eq. (1) and (2).

Step 4: The 8-bit exponent fields of x_{n+1}, y_{n+1} is used as dynamic keys and the 8-bit exponent fields ofA_{minus}, X_{plus}, Y_{plus} are used as static keys.

Step 5: The 8 shuffling positions are obtained by dividing the mantissa fields of x_{n+1}, y_{n+1}.

Step 6: The 8 shuffling output keys are obtained from the dynamic key with respect to the above 8 shuffling positions.

Step 7: Final Transformed pixel is obtained using xor operation between the previous transformed pixel and present shuffling output key for 8 rounds.

Step 8: Encrypted pixel is generated from the final transformed pixel based on the threshold value of shuffling positions.

Step 9: Static keys are used to transform the original pixel and the dynamic keys are used to encrypt the transformed pixel for 8 rounds, which results in the ciphered pixel.

Step 10: Chaotic parameters are updated after the encryption of each pixel byte to incorporate randomness in the key stream.

Step 11: Repeat the steps from 3 to 9 until the input image are encrypted.

Diffusion in decryption process

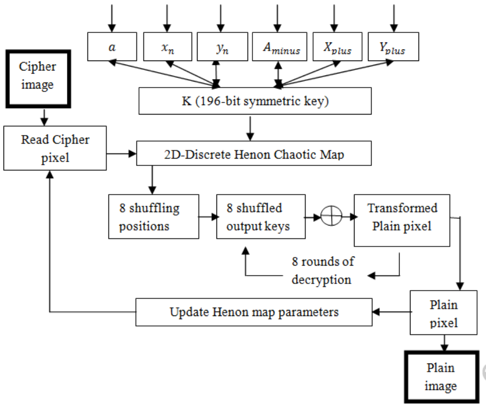

Figure 4. Block Diagram of Diffusion Scheme in Decryption Process

E.Proposed Decryption Algorithm

Step 1: Give the same six floating point inputs to get 192-bit symmetric key to cryptosystem as same as the encryption process.

Step 2: Obtain static keys, dynamic keys, shuffling positions and shuffling output keys as mentioned in encryption.

Step 3: Threshold value of the shuffling positions decides the initial ciphered value to be decrypted.

Step 4: Temporary decrypted pixel is obtained using the xor operation between the previous ciphered pixel and the shuffled output keys in reverse order for 8 rounds.

Step 5: Final decrypted value is obtained by the reverse transformation using the static keys.

Step 6: Chaotic parameters are updated after the decryption of each pixel byte to incorporate randomness in the key stream.

Step 7: Repeat the Steps from 2 to 6 until each pixel of the encrypted image is decrypted.

The pixels of decrypted image should be scanned in the inverse zigzag fashion to get the reverse shuffled image which will be same as plain image. Reverse shuffling of the pixel positions should be done in the opposite way as done in shuffling process.

IV. RESULTS & DISSCUSSIONS

Security analysis will be carried out using the following set of procedures to ensure the security of the encrypted images using the proposed encryption algorithm.

Key Size Analysis: At present, a crypto system should have more than 2^{100} strong private keys to withstand the brute force attack [1]. The presented system uses 196-bit secret key, so 2^{196} keys are generated.

Key Sensitivity Analysis: The sensitivity of the keys is analyzed by giving the correct key once and wrong key once at the time of decryption stage. The decrypted image with wrong key should not reveal any part of the plain image as well as the correct key should decrypt the encrypted image [20].

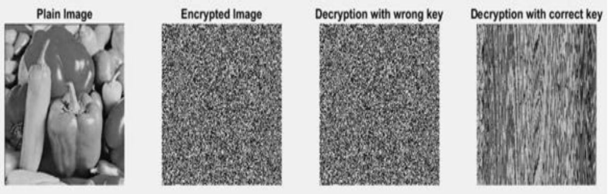

Figure 6. Key Sensitivity Analysis in Matlab

a) **Visual Quality Analysis:** This mainly deals with 2 properties-
   1) Encrypted image should not divulge any part of plain image information
   2) Lossless decryption ability of encryption algorithm means decrypted image should be same as plain image.

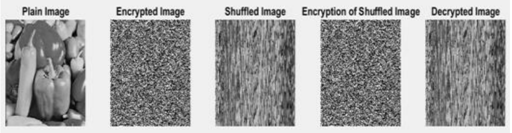

Figure 7. Visual Quality Analysis in Matlab

b) **Analysis of Encryption Quality:** The strength of the encrypted image can be analyzed using the Image Quality Assessment (IQA) parameters, such as Mean Absolute Error (abbreviated as MAE), Mean Square Error (abbreviated as MSE) and Peak Signal to Noise Ratio (abbreviated as PSNR). If two images are more similar, then the values of MAE and MSE between two images are very small and vice versa. PSNR is another measure calculated from MSE, gives an idea about security of encrypted images by means of its visual quality level. PSNR value is represented in decibels (dB). Table 2 presents the encryption ability of the proposed encryption scheme.

$$MAE = \frac{1}{U\times V}\sum_{x=0}^{u-1}\sum_{y=0}^{v-1}|C(x,y) - P(x,y)| \quad (3)$$

$$MSE = \frac{1}{U\times V}\sum_{x=0}^{u-1}\sum_{y=0}^{v-1}|C(x,y) - P(x,y)|^2 \quad (4)$$

$$PSNR = 10\log_{10}\left(\frac{I_{MAX}^2}{MSE}\right) \quad (5)$$

Where,

I_{MAX}^2 = Maximum Pixel Intensity (255 for grayscale images)

C, P = Pixels of Plain and Cipher images

c) Correlation analysis:

Adjacent pixels of the original image will have greater correlation among them, so the correlation coefficient will be 1. Encryption algorithm should result an image with coefficient value should be approximately 0. It means encrypted image should result in the scattered distribution of neighboring pixels in horizontal, vertical, and diagonal direction.

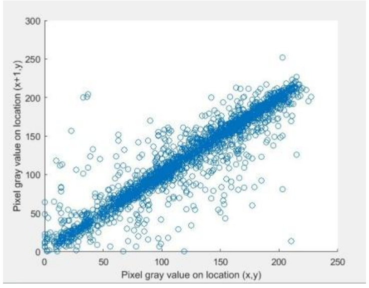

Figure 8. Correlation plot of Plain image

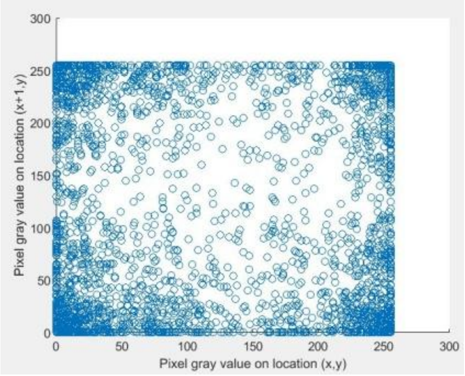

Figure 9. Correlation plot of Ciphered image

d) Binary Histogram Analysis:

To preserve the information about an image, uniformity in the pixel distribution is required. In general, the plain images and shuffled images have non uniformity in the frequency of occurrence various pixel values. It is clearly visible from Fig 10 and Fig.11 that, the histogram of plain image contains too high and too low intensities of pixel values. But in the histogram of encrypted image the uniform distribution of pixel intensity values is achieved.

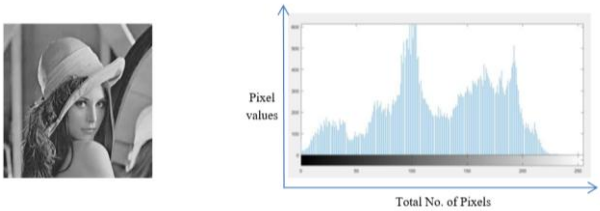

Figure 10. Plain Image and Histogram with Uneven Distribution of Pixels

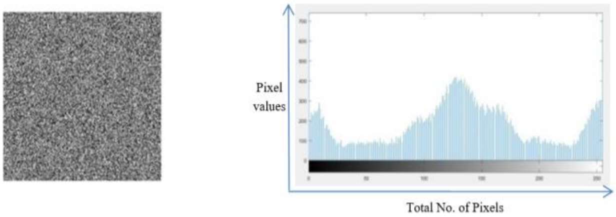

Figure 10. Encrypted Image and Histogram with Uniform Distribution of Pixels

e) Entropy Analysis:

Entropy Analysis: It measures uncertainty and randomness of image pixels. The entropy of the final encrypted image is close to 8, which indicates the good level of encryption. Entropy of Plain and shuffled image is less than pure encrypted image. Shuffled and encrypted image has highest entropy than remaining images as it has highest randomness of pixels.

Table 3. Entropy Measures of Proposed Algorithm

Image	Entropy
Plain Image	7.53
Shuffled Image	7.53
Encrypted Image	7.92
Shuffled and Encrypted Image	7.96

f) Floating Frequency Analysis:

Floating Frequency Analysis: It is the visual investigation analysis of randomness via CrypTool. It will be performed by a graph in which, the number of different grey values possible in each 64-pixel block of image are plotted on vertical axis and the number of pixels of the same image are plotted on horizontal axis. The floating frequency value of encrypted image is close to maximum vertical value than plain image, which shows the strength of encryption algorithm.

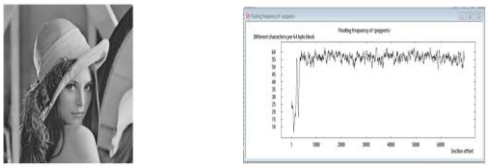
Floating frequency of <peppers>
Floating frequency of <peppers>
Different characters per 64 byte block
60
55
50
45
40
35
30
25
20
15
10
1
1000
2000
3000
4000
5000
6000
Section offset

Figure 12. Plain Image and Floating Frequency Graph

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

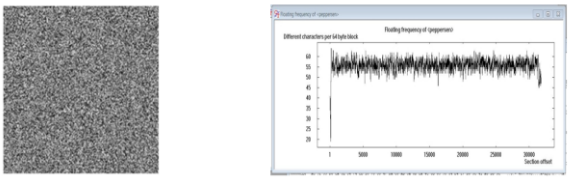

Figure 13. Encrypted Image and Floating Frequency Graph

g) **Differential Analysis*:*** It is an important measure to compare the degree of similarity between the two different images. The Number of Pixels Change Rate (abbreviated as NPCR) and Unified Average Changing Intensity (abbreviated as UACI) are the two metrics analyze the strength of encryption algorithm. NPCR measures the number of changed pixel rate by changing the one pixel of plain image. UACI is calculated between two encrypted images with change in one pixel in corresponding plain images. Table 4 presents the differential measures of the Plain image. Values of NPCR is more than 90 and UACI is more than 30 of the proposed system proves the ability over differential analysis [19]

$$NPCR = \frac{1}{U \times V} \sum_{x=0}^{u-1} \sum_{y=0}^{v-1} D(x,y) \times 100\% \qquad (6)$$

$$UACI = \frac{1}{U \times V} \sum_{x=0}^{u-1} \sum_{y=0}^{v-1} \left[\frac{C(x,y) - P(x,y)}{255} \right] \times 100\% \qquad (7)$$

Where,

$$D(x,y) = \begin{cases} 0, if\ C(x,y) = P(x,y) \\ 1, if\ C(x,y) \neq P(x,y) \end{cases}$$

Table 4. Differential Measures of Proposed Algorithm

Image	NPCR	UACI
Peppers	96.4	30.23

h) **Randomness Test:** This test measures the randomness in the key stream generated by Henon Chaotic Map. A software called CrypTool is popular to test the randomness and it is analysed via Frequency test, Poker test, Long runs test, Serial test. The encrypted image should pass all these tests are treated as robust.

Table 5. Results of Randomness Measures of Proposed Algorithm

Input Image	Frequency Test	Pokers Test	Long Runs Test	Serial Test
Plain	Negative	Negative	Negative	Negative
Shuffled	Negative	Negative	Negative	Negative
Shuffled and Encrypted	Positive	Positive	Positive	Positive

V. COMPARISION OF PROPOSED ALGORITHM WITH EXISTING WORKS

The work by Siva Janakiraman, et al. uses 1-D Logistic Chaotic Map to generate key stream to encrypt gray scale image with lightweight properties. The proposed system uses 2-D Henon Chaotic Map along with lightweight properties which is generating more random key stream.

The algorithm in the work done by Ali Soleymani, et al. uses 2-D Henon Map for encryption and Arnold Cat Map for shuffling the pixels of grayscale image.

Proposed algorithm also uses the same chaotic map but uses zigzag scanning pattern which shuffles the pixels within the less time.

Work by Asia Mahdi and Naser Alzubaidi is implemented by pixel shuffling and Henon Map but the proposed cryptosystem is targeted for embedded devices so light weight properties were also included.

VI. REAL-TIME APPLICATIONS OF PROPOSED ALGORITHM

The proposed algorithm performs light weight encryption of grayscale plain image which provides robust security for the images to be transmitted. Proposed cryptosystem is targeted to be deployed on any microcontroller or on Raspberry Pi. This can be used in Medical field like X-Ray images, Route maps of Military people, E-Commerce, Biometric applications and Confidential E-Mails those contain valuable image data.

VII. CONCLUSION

Chaotic based lightweight image encryption algorithm for grayscale images was presented in this paper along with a set of security analysis. Statistical measures and some visual evidences of security are also presented to strengthen the proposed system. The drawback of existing methods limit the ability to perform with various image sizes. The proposed method can simultaneously work on various image sizes. This system can serve any embedded applications. Even though the proposed algorithm is targeted on embedded platforms, it is not deployed on any hardware. The future work involves the implementation of proposed algorithm on Raspberry Pi and analyses its performance via memory footprint, Timing and system throughput.

FUTURE WORK

In future, we aim to extend the proposed work to achieve practical implementation of the proposed algorithm on Raspberry Pi. Further, we intend to implement in real-time systems. We have chosen symmetric key cryptography because it is fast. But we want to propose the same work with asymmetric key algorithm because it is scalable and provide more authentication and non-repudiation easily.

REFERENCES

1. Lightweight chaotic image encryption algorithm for real-time embedded system: Implementation and analysis on 32-bit microcontroller, SivaJanakiraman, K.Thenmozhi, John Bosco, BalaguruRayappan, RengarajanAmirtharajan, https://doi.org/10.1016/j.micpro.2017.10.013
2. Digital image steganography: Survey and analysis of current methods , Abbas, Cheddad, Joan Condell, Kevin, CurranPaul, Mc Kevitt , https://doi.org/10.1016/j.sigpro.2009.08.010
3. Janakiraman, S.; Amirtharajan, R.; Thenmozhi, K., and Rayappan, J. B. B. (2012). Firmware for data security: A review. *Res. J. Inform. Technol*, 4, 61-72. DOI: 10.3923/rjit.2012.61.72
4. Asia Mahdi Naser Alzubaidi. Color Image Encryption and Decryption using Pixel Shuffling with Henon Chaotic System.ISSN:

2278 – 0181, (IJARCET), Volume 3, Issue 3, March 2014

5. Ali Soleymani, Md Jan Nordin and Elankovan Sundararajan. A Chaotic Cryptosystem for Images Based on Henon and Arnold Cat Map, http://dx.doi.org/10.1155/2014/53693
6. Asia Mahdi and Naser Alzubaidi. Color Image Encryption and Decryption using Pixel Shuffling with Henon Chaotic System, (IJERT), ISSN: 2278-0181
7. Payal Maggo, Rajender Singh Chhillar. Lightweight Image Encryption Scheme for Multimedia Security *International Journal of Computer Applications (0975 – 8887) Volume 71– No.13, May 2013*
8. Bingbing Song and Qun Ding. Comparisons of Typical Discrete Logistic Map and Henon Map
9. Ismail, I. A.; Amin, M. and Diab, H. (2007). An efficient image encryption scheme based chaotic logistic maps. *International Journal of Soft Compution*, 2, 285-29.DOI: ijscomp.2007.285.291
10. Schneier, B. (2007). Applied Cryptography: Protocols, Algorithm and Source Code in C. 2nd Edition, *Wiley*, India. DOI:10.1.1.394.3163
11. S. Koppu and V. M. Viswanatham(2017). A Fast Enhanced Secure Image Chaotic Cryptosystem Based on Hybrid Chaotic Magic Transform, DOI: 10.1155/2017/7470204
12. M.A. Murillo-Escobar , C. Cruz-Hernández, F. Abundiz-Pérez, R.M. López-Gutiérrez(2016). Implementation of an improved chaotic encryption algorithm for real-time emb e dde d systems by using a 32-bit microcontroller, DOI:10.1016/j.micpro.2016.06.004
13. Sunil Jaiswal, Supriya M.Gharat, Sankusu Sharma(2016). Securing image using Chaotic System. ISSN: 2278 – 1323, (IJARCET), Volume 5, Issue 4, April 2016
14. G. Alvarez , S. Li , Some basic cryptographic requirements for chaos-based cryptosystems, Int. J. Bifurcation Chaos 16 (2006) 2129–2151
15. B. S Kumar, S. Karthi, K. Karthika, and Rajan Cristin, “A Systematic Study of Image Forgery Detection”, International Journal of Computational and Theoretical Nanosciences, Vol. 15, No.8, pp. 2560–2564, August 2018
16. Kahan W. (1997). IEEE Standard 754 for Binary Floating-Point Arithmetic [Available Online]. http://www.eecs.berkeley.edu/~wkahan/ieee754status/IEEE754.PDF
17. Fu, C.; Lin, B. B.; Miao, Y. S.; Liu, X. and Chen, J. J. (2011). A novel chaos-based bit-level permutation scheme for digital image encryption. *Optics communications*, 284*, 5415-5423.*
18. Kalra, R., Singhal, A., Kaler, R., & Singhal, P. (2011). Performance Analysis of Proposed MAES Cryptographic Techniques. In *High Performance Architecture and Grid Computing, Springer Berlin Heidelberg*, 316-321. DOI: 10.1007/978-3-642-22577-2_43
19. Praveenkumar, P.; Amirtharajan, R.; Thenmozhi, K.and Rayappan, J. B. B. (2015). Pixel scattering matrix formalism for image encryption—A key scheduled substitution and diffusion approach. *AEU-International Journal of Electronics and Communications*, 69, 562-572
20. Kalpana, J.and Murali, P. (2015). An improved color image encryption based on multiple DNA sequence operations with DNA synthetic image and chaos. *Optik-International Journal for Light and Electron Optics*, DOI, 10.1016/j.ijleo.2015.09.091
21. BalaAnand, M., Karthikeyan, N. & Karthik, S.” Designing a Framework for Communal Software: Based on the Assessment Using Relation Modelling”, Int J Parallel Prog (2018). https://doi.org/10.1007/s10766-018-0598-2
22. M.BalaAnand, S.Sankari, R.Sowmipriya, S.Sivaranjani "Identifying Fake User’s in Social Networks Using Non Verbal Behavior", International Journal of Technology and Engineering System (IJTES), Vol.7(2), pg:157-161.
23. Maram, B., Gnanasekar, J.M., Manogaran, G. et al. SOCA (2018). https://doi.org/10.1007/s11761-018-0249-x
24. M. BalaAnand, N. Karthikeyan, S. Karthick and C. B. Sivaparthipan, "Demonetization: a Visual Exploration and Pattern Identification of People Opinion on Tweets," 2018 International Conference on Soft-computing and Network Security (ICSNS), Coimbatore, India, 2018, pp. 1-7. doi: 10.1109/ICSNS.2018.8573616
25. K. Anupriya, R. Gayathri, M. Balaanand and C. B. Sivaparthipan, "Eshopping Scam Identification using Machine Learning," 2018 International Conference on Soft-computing and Network Security (ICSNS), Coimbatore, India, 2018, pp. 1-7. doi: 10.1109/ICSNS.2018.8573687.
26. CB Sivaparthipan, N Karthikeyan, S Karthik “Designing statistical assessment healthcare information system for diabetics analysis using big data” Multimedia Tools and Applications, 2018
27. Zemedkun Solomon, C.B. Sivaparthipan, P. Punitha, M. BalaAnand, N. Karthikeyan “Certain Investigation on Power Preservation in Sensor Networks” ," 2018 International Conference on Soft-computing and Network Security (ICSNS), Coimbatore, India, 2018, doi: 10.1109/ICSNS.2018.8573688