Secure Searchable Image Encryption in Cloud Using Hyper Chaos

Shaheen Ayyub and Praveen Kaushik
Department of Computer Science and Engineering, Maulana Azad National Institute of Technology, India

Abstract: *In cloud computing, security is the main issue to many cloud providers and researchers. As we know that cloud acts as a big black box. Nothing inside the cloud is visible to the cloud user. This means that when we store our data or images in the cloud, we lost our control upon it. The data in the provider's hands could make security and privacy issues in cloud storage as users lose their control over their data. So it is necessary for protecting user's private data that they should be stored in the encrypted form and server should not learn anything about the stored data. These data may be personal images. In this paper we have worked on the user's personal images which should be kept secret. The proposed scheme is to do the encryption of the images stored in the cloud. In this paper Hyper Chaos based encryption is done, which is applied on the masked images. Comparing with conventional algorithms chaos based ones have suggested more secure and fast encryption methods. The flicker images are used to create the mask for the original image and then hyper chaos is applied for encrypting the image. Prior methods in this regard are restricted to either some attacks possibility or key transfer mechanism. One of the advantages of proposed algorithm is that, the key is also encrypted. Some values of generated encrypted key with the index is sent to the server & other value is sent to the user. After decrypting the key, an encrypted image can be decrypted. The key encryption is used to enhance the security and privacy of the algorithm. Index is also created for the images before storing them on the cloud.*

Keywords: *Cloud computing, encryption, cloud security, privacy and integrity, hyper chaos, decryption, logistic map.*

Received June 27, 2015; accepted June 1, 2016

1. Introduction

The world we are living in is a connected and highly intensive world. The great advances in networking and information technologies have enabled users to collect and generate large amount of multimedia data. To store and process such kind of multimedia data requires additional storage and high computational power that may not be available to all users especially in case of light weight device users (e.g., Mobile and iphone devices). For this type of scenario cloud computing is best suited. As cloud computing has been emerged as a new technology that offer to its users attractive financial and technological advantages [14]. However using cloud for storage and processing of images, especially when images have sensitive information, is risky because private data can leak to outside world. The cloud act as a big black box nothing inside the cloud is visible to its users. Cloud users have no idea or control over what happens inside the cloud. Even if cloud provider is honest, it can have malicious system administrators who can violate the confidentiality and integrity of images. To restrict unauthorized access of the images, users usually encrypt their sensitive data or personal images before uploading them onto the cloud servers.

However, traditional encryption algorithms pose a significant barrier towards searching the encrypted data [1] because data needs to be decrypted before processing. Thus, storing data inside the cloud would be useless if it cannot be processed. So we need an encrypted algorithm which is able to search or process encrypted data. These types of algorithms are called searchable encryption algorithms. Various searchable algorithms in the past years have been proposed in this regard. This paper introduces a new algorithm for enhancing the privacy and security of the images before uploading them onto the cloud. In this algorithm hyper chaos is applied to encrypt the image and to enhance the security key encryption is also performed.

The rest of this paper is organized as follows. Related research work is explained in section 2. In section 3, the proposed System Model has been described. In section 4 proposed algorithm is explained. Section 5 contains a threat analysis and results of the proposed algorithm for some experiments. Section 6 describes the conclusion and future work.

2. Related Work

Over the years, several Searchable Encryption (SE) approaches have been proposed [4, 17] to provide the ability for selectively retrieving the encrypted documents. As it is known that the images are different from texts in many aspects, such as high redundancy and correlation. The main obstacle in designing effective image encryption algorithms is the difficulty of shuffling and diffusing such images by traditional cryptographic means. In most of the natural images,

the value of any given pixel can be reasonably predicted from the values of its neighbors. Several researchers have used traditional crypto graphic primitives to protect images before storing them in untrusted storage [9] which do not admit computation in clouds. Due to the processing overhead resulting from the large data size of digital images and the high correlation among pixels, traditional encryption algorithms, such as Data Encryption Standard (DES), Advanced Encryption Standard (AES) [22] and Rivest, Shamir, and Adelman (RSA), are found to be inefficient for image encryption [7]. Using mask images to enhance the privacy of an image has been explored in image stenography [5].

Comparing with conventional algorithms, chaos based ones have suggested more secure and fast encryption methods. Chaotic maps are sensitive to their initial conditions. A simple change in one pixel of input image affects large number of pixels in the cipher image which makes the computation on cipher image impossible. Chaotic maps have been explored by many researchers for image encryption [6, 10, 11, 12, 15, 21]. Fu and Zhu [7] proposed a technique based on logistic maps with permutation and circular bit-shift methods for confusion and diffusion. Nourian *et al.* [12] have proposed an algorithm for color image encryption in cloud using cat map. They have used image masking to encrypt the original images. To some extent their algorithm is not secure enough to some known attacks.

3. Proposed System Framework

In the proposed work, a user having a low computational power (e.g., mobile devices) connects to the cloud. The user desires to use the storage capacity and cloud computational power to store his/her personal data (images). However, the user wants that his/her personal data must be secure enough before outsourcing it to the cloud. Figure1 shows the system framework of proposed work which consists of eight functional blocks for image storage and accessing images securely from data centers in public cloud servers.

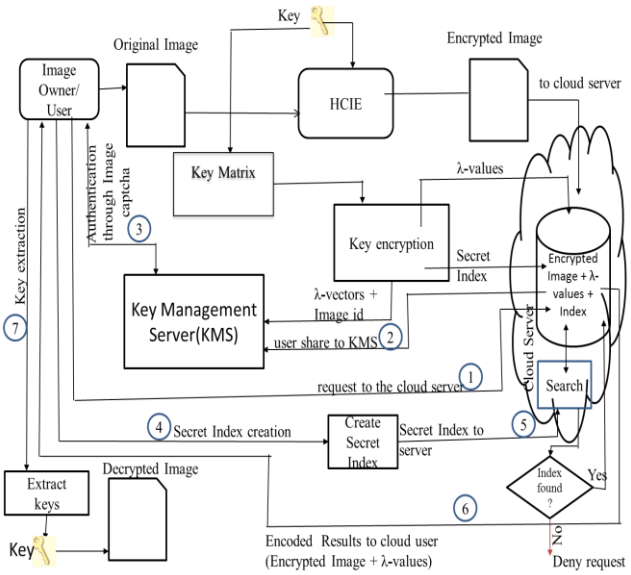


Figure 1. System framework.

For encrypting the images using Hyper Chaotic Image Encryption (HCIE), the user needs to get registered itself through Key Management Server (KMS). The encryption process creates λ-vectors, λ-values and a Secret Index. During this process λ-values and Secret Index are sent to the cloud server and λ-values with some other information are stored in KMS. When user wants to retrieve the image from cloud server, it sends the request to the cloud server, this request is processed after verification. After receiving the request from user, the cloud server searches for a particular index in the database. If a match is found then the encrypted image and λ-values are sent to the user otherwise the request is denied. When user got the encrypted image and λ-values, it then extract the keys for decrypting the image. The process of Image encryption and key encryption is discussed in the next section.

4. Proposed Algorithm

Our proposed algorithm consists of the following main tasks:

4.1. Image Encryption

The original color image is first mixed with the image obtained from the social media (flicker) by using the flicker ID (flk_ID). Mixing of an image with other image is called image masking. Two hash functions [2] h_1 (z) and h_2(x, y, z) have been used to create the flk_ID. These hash functions depend upon the features of the original image. By doing this first complexity is being applied to the encryption algorithm that makes it more robust against widespread attacks. After getting the masked image, logistic map is used to shuffle the image to get the encrypted image. To shuffle the masked image using logistic map, image subdivision & permutation and pixel shuffling is applied. On this

shuffled image hyper chaos are applied. Image permutation makes the original image prediction little bit confusing. Figure 2 shows the basic structure of encryption process and Figure 3 elaborates the hyper chaotic encryption block of Figure 2.

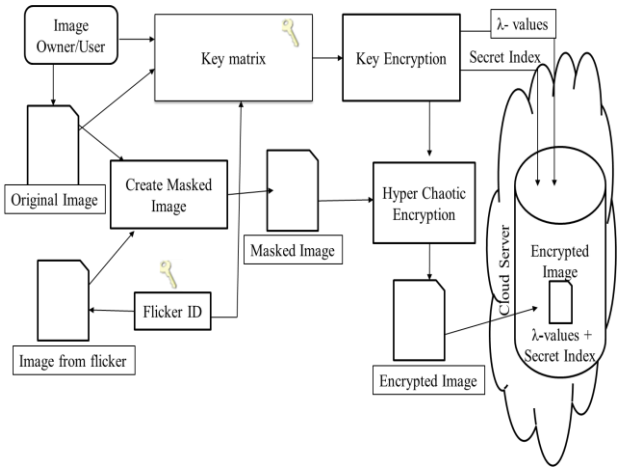


Figure 2. Basic structure of encryption process.

4.1.1. Image Subdivision and Permutation of blocks

For image subdivision, the whole masked image is divided into 8 equal blocks and then a pseudo-random array is generated from the logistic map shown in Equation (1).

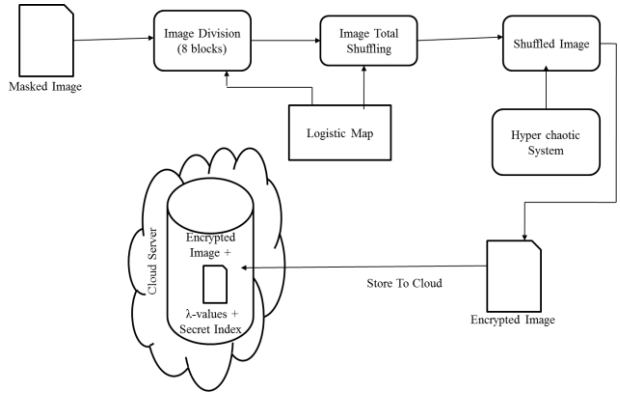


Figure 3. Hyper chaotic encryption.

Which contains 8 pseudo-random numbers that is used to disorder the actual arrangement of image blocks.

$$x = rx_n(1 - x_n) \tag{1}$$

For an initial value of x_0, perform some iteration and obtain a new x_0, and then let:

$$Randompermutation = [\mathrm{mod}(x_0 \times 10^{14}, 7)] + 1 \tag{2}$$

Continuously iterate the logistic map and perform Equation (2) until 8 different values which are all between 1 and 8 are not obtained. This kind of procedure adds another complexity to the encryption process as it adds an extra step to it, and furthermore the length of the key will become longer.

This process can be more understood from Figure 4 As shown in Figure 4, the disordered image is obtained as follows. First, the block number "1" is changed with block number "6." Next, the block number "2" is changed with block number "8." Then block number 3, 4, 5, 6, 7 and 8 are changed with block number 7, 2, 5, 4, 1 and 3 respectively.

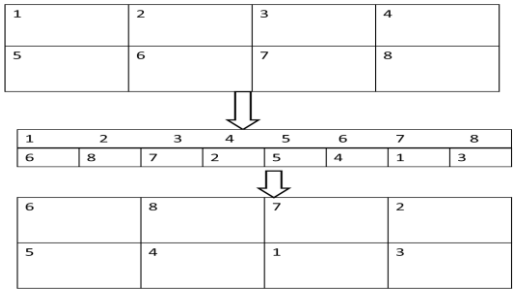


Figure 4. Block shuffling.

4.1.2. Shuffling of Pixels

Pixel shuffling matrix is used in order to shuffle the position of the pixels in the blocks and to disturb the high correlation among adjacent pixels and weaken this strong correlation among them [8].

The generation procedure of shuffling matrix is described in the following steps.

1. Row Shuffling

$$\begin{gathered} \textit{Dimension of a block : } B_m \times B_n \\ \textit{Pixels Position matrix : } P_{i,j}\,, \\ \textit{where } i = 0,1,2,\ldots\ldots B_m\text{-}1 \;\; \& \;\; j = 0,1,2,\ldots\ldots B_n\text{-}1 \\ RS = [\mathrm{mod}(r_0 \times 10^{14}, B_m)]\,, \quad RS \epsilon [0, B_m\text{-}1] \end{gathered} \tag{3}$$

2. Continue to do the iterations of logistic map until B_m different values are not formed as:

$$[h_i \;,\; i = 1,2\ldots\ldots,B_m] \quad \textit{where} \quad h_i \neq h_j$$
$$\textit{new Pixel Position Matrix} = P^h{}_{ij}$$

3. Column shuffling

$$CS = [\mathrm{mod}(c_0 \times 10^{14}, B_n)]\,, \; CS \epsilon [0, B_n\text{-}1] \tag{4}$$

4. Continue to do the iterations of logistic map until B_n different values are not formed as:

$$[l_i \;,\; i = 1,2\ldots\ldots,B_n] \quad \textit{where} \quad l_i \neq l_j$$
$$\textit{new Pixel Position Matrix} = P^h{}_{ij}$$

All the above steps are repeated continuously for each and every block of the mask image.

4.1.3. Liu and Chen's Chaotic System

In the proposed algorithm, the Liu chaotic system used in cryptography is described as follows [20]:

$$\begin{aligned} X &= r(y - X) \\ Y &= sX - XZ \\ z &= tZ + xX^2 \end{aligned} \tag{5}$$

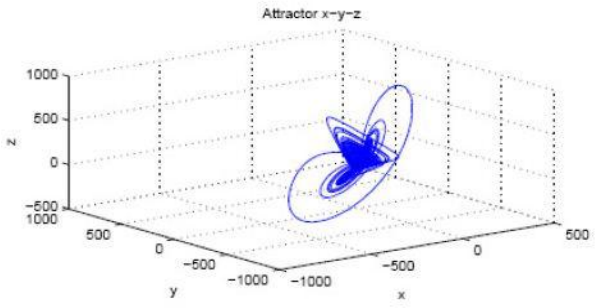
Where *r*, *s*, *t* and *u* are the parameters when *r*=10; s=40; t=8/3, u = 4 the system is hyper chaotic.

Another Chaotic system we have used is Chen's chaotic system described as follows[20]:

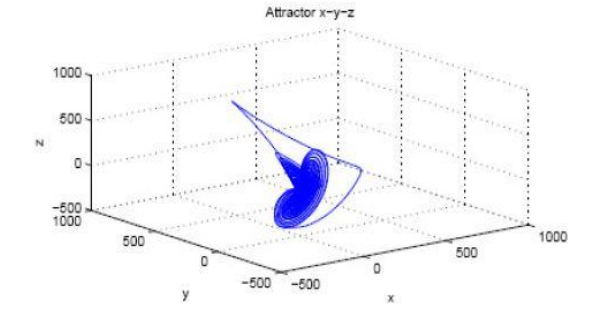
$$\begin{aligned} P &= a(Q-P) \\ Q &= (c-a)P - PR + cQ \\ R &= PQ - bR \end{aligned} \quad (6)$$

Where *a*, *b*, *c* are the parameters when a=35; b=3 c=28 the system is hyper chaotic.

A hyper chaotic attractor is typically defined as chaotic behavior with at least two positive Lyapunov exponents. Combined with one null exponent along the flow and one negative exponent to ensure the boundness of the solution, the minimal dimension for a (continuous) hyper chaotic system is 4. We have two positive lyapunov exponents of Liu and Chen's hyper chaotic system and so we know that the linearization system of the Liu and Chen hyper chaotic system is unstable [13]. The chaotic behavior of the two chaotic systems is shown in Figure5. We have used 4th order RK method to solve the two chaotic systems.



a) Attractors of liu system in x-y-z plain [18].



b) Attractors of chen's system in x-y-z plain [18].

Figure 5. The chaotic behavior of the two chaotic systems.

4.1.4. State Combination

The state combination behavior of two chaotic systems is used in this algorithm. Three of the variables are combined differently, which is given in Table 1.After applying Hyper chaos final encryption is done by using three bytes of original image, three bytes of shuffled image and three bytes of the matrix created by using hyper chaos.

4.2. Key Encryption

In the above proposed algorithm the keys are Initial value of Logistic map for block, row and column permutation respectively: b_0, r_0, col_0. The number of iterations for logistic map: b_I, r_I, c_I. Initial values of the Liu and Chen's systems respectively: x(1), y(1), z(1) & p(1), q(1), r(1).

Table 1. State combination.

Serial Number	Combination of states	Serial Number	Combination of states
0	P Q R	4	X Q R
1	Z P Q	5	X Z Q
2	Y P R	6	X Y R
3	Y Z P	7	X Y Z

Number of iterations: N_0, M_0 and *flk_ID* to retreive the random image from flicker. For encrypting the key, first of all create a matrix from all the keys while leaving one value as zero as follows.

$$A = \begin{bmatrix} x(1) & p(1) & b_0 & bI \\ y(1) & q(1) & r_0 & rI \\ z(1) & r(1) & col_0 & cI \\ M_0 & N_0 & flk\_ID & 0 \end{bmatrix} \quad (7)$$

By leaving one value as zero we just reduce the number of unknown variables. If the number of unknowns is less, then computation will be comparatively simple and lightweight. After creating the matrix *A*, characteristic equation for A is created by using the following formula:

$$A - \lambda I = 0 \quad (8)$$

And then calculate 4 λ-values (λ_1 λ_2 λ_3 λ_4). From the λ-values created above, four λ-vectors are obtained as follows:

$$V1.\begin{bmatrix} S1 \\ S2 \\ S3 \\ S4 \end{bmatrix} \quad V2.\begin{bmatrix} S5 \\ S6 \\ S7 \\ S8 \end{bmatrix} \quad V3.\begin{bmatrix} S9 \\ S10 \\ S11 \\ S12 \end{bmatrix} \quad V4.\begin{bmatrix} S13 \\ S14 \\ S15 \\ S16 \end{bmatrix}$$

From the above vectors index is created for the encrypted image as in Equation (9). λ-values and Index will be sent to the server with the encrypted image and λ-vectors will be sent to the authenticated user.

$$Index = [\max(s_1, s_2, s_3, s_4) + \max(s_5, s_6, s_7, s_8) + \max(s_9, s_{10}, s_{11}, s_{12})] \bmod [\min(s_{13}, s_{14}, s_{15}, s_{16})] \quad (9)$$

4.3. Decryption

The authenticated user, who has verified with the cloud server generates the Secret Index from the *λ-vectors* and sends request to the cloud server by sending the Secret Index to retrieve the encrypted image. Upon receiving the user's request, cloud server sends the *λ-values* and the encrypted image to the user. After

getting the *λ-values* and encrypted image from the cloud server, user performs the process of decryption. It consists of two steps: Key Decryption and Image Decryption.

4.3.1 Key Decryption

When user receives the λ-values it will perform some calculation to generate the key. Key decryption will be done using the λ-vectors and λ-values from the server by using the following formula, and create the matrix *A*.

$$\begin{array}{c}\text{for i = 1 to 4}\\ A\,v_i = \lambda_i\,v_i\\ \text{end}\end{array} \qquad (10)$$

4.3.2. Image Decryption

Decryption will be done by using the keys as in matrix A. Decryption process is to perform all the operations in reverse. Figure 6 and 7 shows the flow diagram of Encryption and Decryption process.

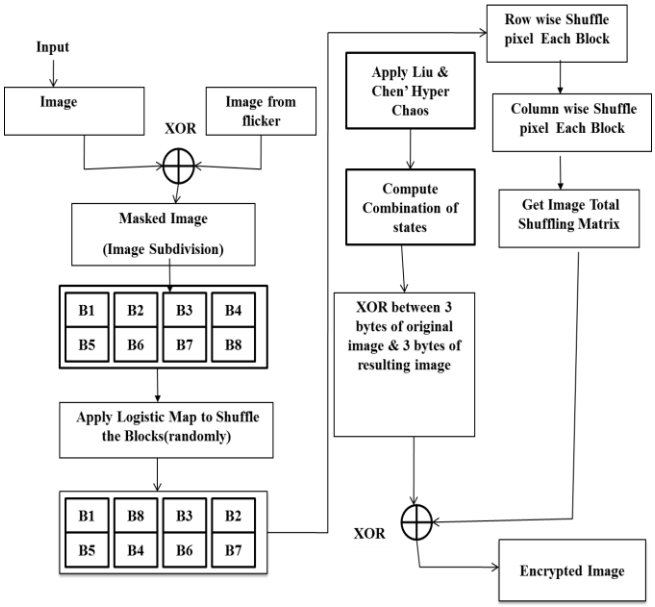


Figure 6. Flow diagram of encryption algorithm.

5. Performance Analysis

Implementation of the proposed work is done in MATLAB2013(a) and run on an Intel(R) Core™2Duo CPU 2.10 GHZ PC with 2GB RAM. MATLAB provides cloud environment using MathWorks Cloud. MATLAB distributed computing server can be used in a variety of cloud environments.

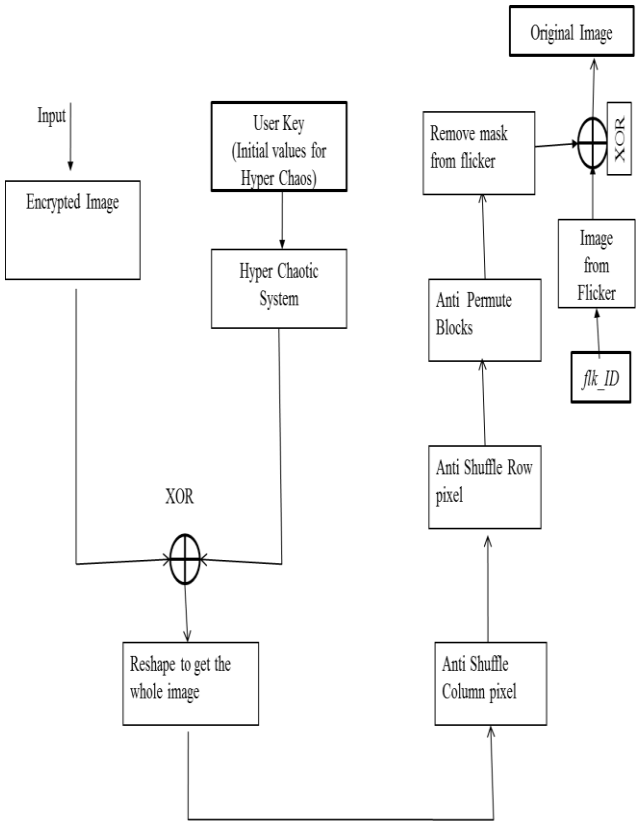
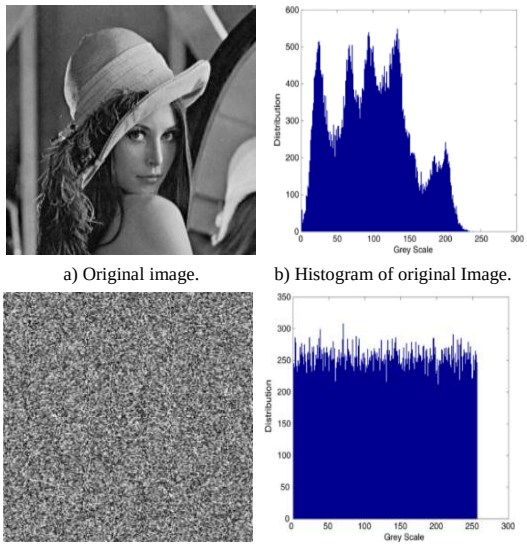


Figure 7. Flow diagram of decryption algorithm.

5.1. Histogram Analysis

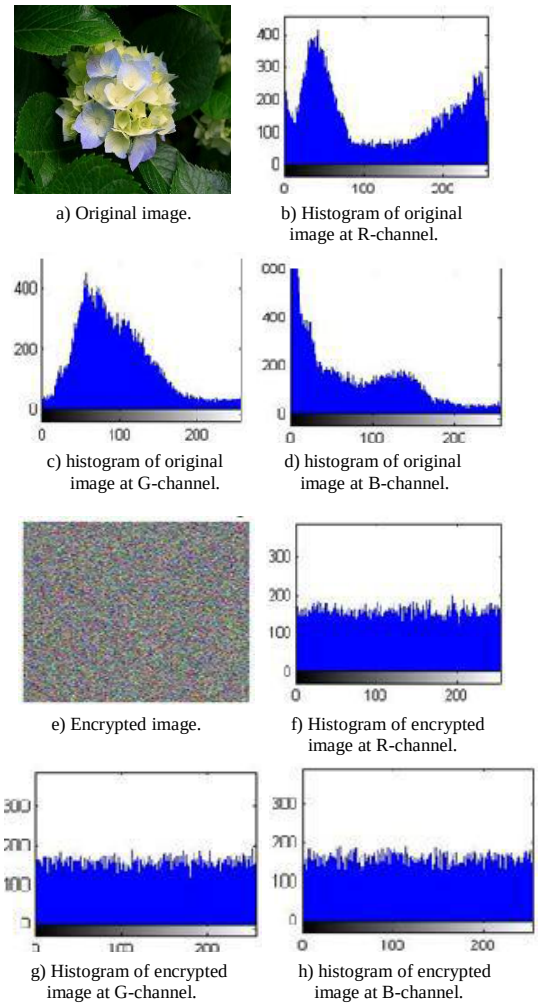
Histogram analysis is done as an attack technique to assign the image to a specific class of images by looking for common histogram shapes or data. To address such attacks, the histogram of the encrypted image should be different than that of the original image. An encryption algorithm to be secure enough against attack techniques related to histogram should produce cipher-images such that they have salient differences with their corresponding original images from statistical points of view. We have done experimental analysis of the proposed image encryption algorithm on color image as well as gray image. The image Lena (gray image) of size 512×512 and the histogram of the gray image is shown in Figure 8-a and 8-b respectively. Corresponding encrypted image and the histogram of the encrypted image is shown in Figure 8–c and 8-d respectively. From the figure, we can observe that the histogram of the ciphered image is fairly uniform and is significantly different from that of the original image.



a) Original image. b) Histogram of original Image.

c) Encrypted image. d) Histogram of encrypted image.

Figure 8. Image lena.



a) Original image. b) Histogram of original image at R-channel.

c) histogram of original image at G-channel. d) histogram of original image at B-channel.

e) Encrypted image. f) Histogram of encrypted image at R-channel.

g) Histogram of encrypted image at G-channel. h) histogram of encrypted image at B-channel.

Figure 9. Image flower.

5.2. Correlation Coefficient of Adjacent Pixels

The correlation between two adjacent pixels in the encrypted image should be as low as possible. The low correlation between two adjacent pixels in the encrypted image means that the two adjacent pixels in the encrypted image are less correlated, whereas the correlation between two adjacent pixels in the original image is high. The high correlation between two adjacent pixels in the original image shows that the two adjacent pixels are highly correlated. In order to test the correlation of adjacent pixels correlation coefficient of adjacent pixels is calculated. For this first of all randomly select 3000 pairs of two adjacent pixels from the image, and then calculate the correlation coefficient of each pair by using the following formulas:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} xi \tag{11}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x_i))^2 \tag{12}$$

$$\text{cov}(x, y) = \frac{1}{N}\sum(x_i - E(x_i))(y_i - E(y_i)) \tag{13}$$

$$rxy = \frac{\text{cov}(x, y)}{\sqrt{D(x)D(y)}} \tag{14}$$

Where x and y are gray values of two adjacent pixels in the image.

Tables 2 and 3 shows the simulation result of the proposed algorithm for the image Flower (color image) and image Lena (gray level). The results show that the correlation coefficient is very close to zero in the encrypted image, and thus the proposed algorithm is less predictable and more secure.

Figures 10 and 11 shows the graph of correlation coefficients for different categories of images. The graph is plotted for horizontal and vertical direction for encrypted image. It can be seen from both of the two graphs that the correlation coefficient of original image is very high as compared to encrypted image and for some image categories it touches the highest value.

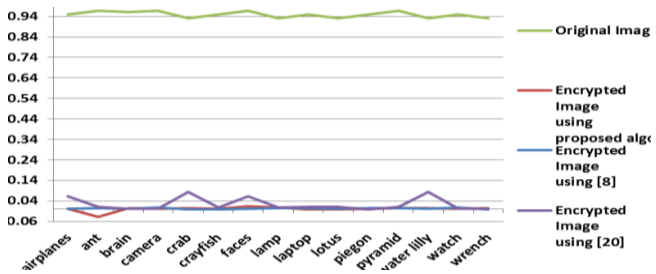


Figure 10. Correlation coefficient in horizontal direction.

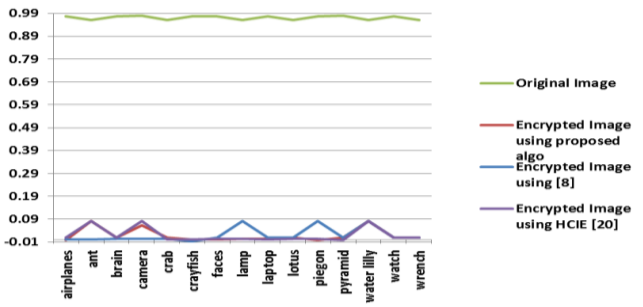


Figure 11. Correlation coefficient in vertical direction.

Table 2.Coefficient of correlation of original and ciphered image.

Scan Direction	Image Lena		Image pepper					
	Original Image	Encrypted Image	Original Image			Encrypted Image		
			R	G	B	R	G	B
Horizontal	0.9491	-0.0006	0.9604	0.9778	0.9561	0.0069	-0.0019	-0.0015
Vertical	0.9768	-0.0030	0.9674	0.9796	0.9573	0.0024	0.0033	0.0096
Diagonal	0.9304	0.0061	0.9327	0.9608	0.9195	0.0001	0.0067	-0.0024

Table 3. Comparison of coefficient of correlation of proposed method with recent methods.

Scan Direction	Horizontal	Vertical	Diagonal
Original Image(Lena)	0.9506031	0.9761479	0.9254270
Proposed Algorithm	-0.0006	-0.0030	0.0061
Algorithm Based on Hyper-chaos[4]	-0.04005	0.08158	-0.00471
parallel sub-image encryption with hyper chaos[17]	0.00534	0.00846	0.00355
Original Image(Flower)	0.9681485	0.9616387	0.9396524
Proposed Algorithm	0.00401	0.00863	0.00253
Color Image Encryption algorithm [5]	0.00070	0.00216	0.01488
Privacy Enhanced Limited Image Processing in the Clouds[18]	0.01183	0.00016	0.01480

5.3. Information Entropy

Entropy is one of the most outstanding features that make the images to have a random-like behavior. This parameter was first introduced by Shannon [16]. And can be obtained using the following formula:

$$Entropy = \sum_{i=0}^{2N-1} Prob(g_i) Log\left(\frac{1}{Prob(g_i)}\right) \qquad (15)$$

Where N is the number of gray scale levels in an image (ex: N = 256 for 8 bit image pixels) and Prob (g_i) is the occurrence probability of gray scale in the image. The entropy value will be 8 for images that are produced totally randomly. The closer the entropy of an encryption algorithm is to 8 the less predictable, and thus more secure is that Algorithm. The entropy value for the proposed encryption algorithm for an image is shown in Table 4.

Table 4. Comparison of entropy of the encrypted image and the original image.

Original Image	Encrypted Image				
	Ref[4]	Ref[20]	Ref[3]	Ref [18]	Proposed
256*256(7.5060)	7.9993	7.8980	7.9912	NA	7.9992
1024*768(7.121)	NA	7.9121	7.9871	7.9992	7.8929
4265*4265(7.234)	7.9991	NA	NA	7.8911	7.8192

5.4. Attacks Analysis

The proposed algorithm the Initial value of X_0 which is used for image division is 0.546 (32 bit) and the Initial Parameters of the Liu and Chen's systems are: r=10; s=40; t=8/3; u = 4 and a=35; b=3; c=28; ≈(96 bits) with the Initial values : x(1)=2.9; y(1)=-1.3; z(1)=25.9; p(1)=0.3; q(1)=0.4; r(1)=1.2; (104 bits) and the Initial iteration numbers N_0= 3000, M_0 = 2000 (64 bits), and the *flk_ID*. Thus total number of bits needed to store all the encryption parameteres is 296. Therefore combinations can be 2^{296} which is a very large key space. So we observe that the proposed algorithm is sensitive to the key because we have used the chaos and it is not possible to find the combination of this large key space. Therefore the algorithm can resist the Brute force attack. In known plain-text, an attacker has gained access to a set of plain and encrypted images willing to analyze them to extract the key [21]. In our algorithm each input image gets its own mask and its pixels' positions are changed by logistic map and hyper chaos with specific parameters based on the content of input image. Therefore, attackers are not able to gain any information by analyzing the pairs of plain and encrypted images. In chosen plain-text attack, the attacker chooses a set of plain images and observes the corresponding encrypted images to recover the key. In our algorithm while getting the flicker ID to get mask of the image an attacker cannot get the original image because we are providing double masking using hyper chaos, and by using the exact keys only we get the original image. Little bit difference changes the output and results in wrong decryption.

5.5. Encryption Speed

We have used Matlab 2013(a) to run encryption programs in a computer with a Intel(R) Core™2Duo CPU 2.10 GHZ PC with 2GB RAM and the operating system is Microsoft Windows7. Table 5 shows the comparison of experiment results for the proposed algorithm for various dimensions. Compared to this encryption system, we can see that the operation speed of our algorithm is faster than other algorithms for some dimensions.

Table 5. Comparison of encryption time of proposed algorithm for various dimensions.

Image Dimension	Image Encryption time(sec)		
	Ref [15]	Ref [8]	Proposed Algorithm
500 * 500	0.192 sec	0.248 sec	0.240 sec
1000*1000	0.597 sec	0.561 sec	0.585 sec
2000*2000	2.36 sec	3.00 sec	2.50 sec
4000*4000	7.685 sec	7.124 sec	7.875 sec

6. Conclusions and Future Work

An image encryption algorithm that enhances the security and privacy of images outsourced to the cloud for storage is proposed in this thesis. One of the distinguishing features of the proposed algorithm is that the computation takes place without cloud server having the opportunity to gather any intelligence about the images. This algorithm is effective against cloud services that are "Honest-but-curious", where they are trying to learn as much about the customers. In this algorithm we used color images as well as gray images. We have used chaotic system for encrypting the image, as chaotic system is comparatively cryptographically secure. The experimental results show that the algorithm possesses high security and a large key space. Although for some images its performances goes low but it is effective in most of the cases. The proposed encryption algorithm is also fast; there are only some XOR operations for each pixel. Our future work is to enhance the performance of the algorithm for all variety of images and we are also going to implement the authentication procedure for the cloud user.

References

[1] Abdulsada A., Ali M., Abduljabbar Z., and Hashim H., "Secure Image Retrieval Over Untrusted Cloud Servers," *International Journal of Engineering and Advanced Technology*, vol. 3, no. 1, pp. 140-147, 2013.

[2] Canetti R. and Dakdouk R., "Extractable Perfectly one Way Functions," *International Colloquium on Automata, Languages, and Programming*, Reykjavik, pp. 449-460, 2008.

[3] Chen G., Wang Y., Wong K., and Liao X., "A new Chaos based Fast Image Encryption Algorithm," *Applied Soft Computing*, vol. 11, no. 1, pp. 514-522, 2011.

[4] Curtmola R., Garay N., Kamara S., and Ostrovsky R., "Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions," *in Proceedings of 13th ACM Conference on Computer and Communications Security*, Virginia, pp. 79-88, 2006.

[5] Eggers J., Bauml R., and Girod B., "A Communications Approach to Image Steganography," *in Proceedings of SPIE: Electronic Imaging, Security and Watermarking of Multimedia Contents*, San Jose, pp. 26-37, 2007.

[6] Faroun K., "Chaos-Based Key Stream Generator Based on Multiple Maps Combination and its Application to Images Encryption," *The International Arab Journal of Information Technology*, vol. 7, no. 3, pp. 231-240, 2010.

[7] Fu C. and Zhu Z., "A Chaotic Image Encryption Scheme based on Circular Bit Shift Method," *in Proceedings of The 9th International Conference for Young Computer Scientists*, Hunan, pp. 3057-3061, 2008.

[8] Gao T. and Chen Z., "A New Image Encryption Algorithm Based on Hyper-Chaos," *Physics Letters*, vol. 372, no. 4, pp. 394-400, 2008.

[9] Goh E., Shacham H., Modadugu N., and Boneh D., "Securing Remote Untrusted Storage," *in Proceedings of Network and Distributed Systems Security Symposium*, San Diego, pp. 131-145, 2003.

[10] Hong L., Ming B., and Hui H., "New Image Encryption Algorithm Based on Logistic Map and Hyper-Chaos," *in Proceedings of International Conference on Computational and Information Sciences*, Shiyang, pp. 713-716, 2013.

[11] Mirzaei O., Yaghoobi M., and Irani H., "A New Image Encryption Method: Parallel Sub-image Encryption with Hyper Chaos," *Nonlinear Dynamics*, vol. 67, no. 1, pp. 557-566, 2011.

[12] Nourian A., "Towards Privacy Enhanced Limited Image Processing in the Clouds," *in Proceedings of 9th Middleware Doctoral Symposium of the 13th ACM/IFIP/USENIX International Middleware Conference*, Canada, pp. 1-6, 2012.

[13] Niu H., Ma S., Fan T., Chen C., and He P., "Linear State Feedback Stabilization of Unified Hyperchaotic Systems," *World Journal of Modelling and Simulation*, vol. 10, no. 1, pp. 34-48, 2014.

[14] Reese G., *Cloud Application Architecture: Building Applications and Infrastructure in the Cloud*, O'reilly Media, 2009.

[15] Roohbaksh D. and Yaghoobi M., "Color Image Encryption using Hyper Chaos Chen," *International Journal of Computer Applications*, vol. 110, no. 4, pp. 9-12, 2015.

[16] Shannon C., "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, pp. 379-423, 1948.

[17] Song D., Wagner D., and Perrig A, "Practical Techniques for Searches on Encrypted Data," *in Proceedings of IEEE Symposium on Security and Privacy*, Berkeley, pp. 44-55, 2000.

[18] Vaidyanathan S., "Global Chaos Anti-Synchronization of Liu and Chen Systems by Nonlinear Control," *International Journal of Mathematical Sciences and Applications*, vol. 1, no. 2, pp. 691-702, 2011.

[19] Wang Q., Li J., Wang C., Cao N., Ren K., and Lou W., "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," *in Proceedings IEEE INFOCOM*, San Diego, pp. 441-445, 2010.

[20] Wang X., Chen F., and Wang T., "A New Compound Mode of Confusion and Diffusion for Block Encryption of Image based on Chaos," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 9, pp. 2479-2485, 2010.

[21] Wei W., Fen L., Xinl G., and Yebin Y., "Color Image Encryption Algorithm Based on Hyper Chaos," *in Proceedings of 2nd IEEE International Conference on Information Management and Engineering*, Chengdu, pp. 271-274, 2010.

[22] Zeghid M., Machhout M., Khriji L., Baganne A., and Tourki R., "Modified AES Based Algorithm for Image Encryption," *International Journal of Computer Science and Engineering*, vol. 1, no. 1, pp. 745-750, 2007.

Shaheen Ayyub received BE degree from the Department of Computer Science and Engineering, Barkatullah University Institute of Technology, India, in 2005 and M.Tech. degree from the Department of Computer Science and Engineering RGPV, Bhopal, India, in 2009. Currently, she is pursuing her PhD degree under Maulana Azad National Institute of Technology, Bhopal, India. Her research area includes cloud computing, information security and mobile computing.

Praveen Kaushik obtained his B.E. and M.Tech.degree in CSE from India. He did his Ph.D. in the field of wireless sensor networks. He is presently working as an Assist-atant Professor in the department of Computer Science and Engineering, Maulana Azad National Institute of Technology, Bhopal (M.P) India. He is having a total of 14 years of teaching experience. His research area includes network security, wireless sensor network, cloud and mobile computing. He has been author/co-author of more than 15 research papers in international conferences and journals of high repute, he has been the reviewer of IEEE international conferences and IET Journal of Networks. He is an active member of Computer Society of India and International Association of Engineering (IAENG).