Practical Approaches for Image Encryption/Scrambling Using 3D Arnolds Cat Map

Pawan N. Khade and Manish Narnaware

G. H. Raisoni College of Engineering, Nagpur
pawan.khade@gmail.com, manish.narnaware@yahoo.com

Abstract. This paper is exploratory study of the 3D Arnolds Cat Map. The paper discusses the Arnold's cat map and its 3D extension that is 3D Arnolds Cat Map in detail. This paper extends idea of encryption/scrambling to the encrypting/scrambling colour image using encryption/scrambling of R, G, and B components. Experimental implementation of two different 3D Arnolds Cat Maps proposed by different authors are provided along with their results. Paper also discusses inverse ACM transformation to recover the scrambled image.

Keywords: Image encryption, Arnolds Cat Map, 3D Arnolds Cat Map, Chaos.

1 Introduction

Chaotic encryption is relatively new area in the network security & cryptography and gaining widespread acceptance. Chaotic maps are blessed with features of sensitivity to the initial condition, and ergodicity which make them very desirable for encryption [4]. It has been found that chaotic algorithms are faster than classical algorithms like DES, IDEA, MD5 [1]. Chaotic Arnolds Cat Map (ACM) is generally used for image scrambling. ACM can provide only scrambling of image pixels which does not provide desirable level of security and additionally contains less number of constants. To deal with these problem higher dimensional ACM maps has been proposed by many authors [2], [5], [6]. 3D ACM are more secure because they provide the additional substitution apart from scrambling of image and also contain more number of constants. Next section discusses Arnold's cat map, and two different 3D Arnolds Cat Maps in detail along with their implementation strategies and results. Lastly the comparison between ACM and different 3D ACM is made to summarize the output.

2 Arnolds Cat Map (ACM)

Arnold's cat map was invented by Russian mathematician Vladimir I. Arnold in the 1960s. The ACM Transformation is given by $\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod N$,

V.V. Das and J. Stephen (Eds.): CNC 2012, LNICST 108, pp. 398–404, 2012.

where NxN is dimension of image [4]. Above representation is equivalent to $\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$ mod N, where p=1 and q=1. Putting different values of p and q gives the variation of the Arnolds Cat Map transformation.

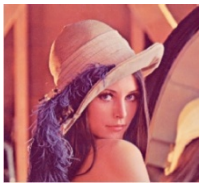

Fig.1. Originalimage

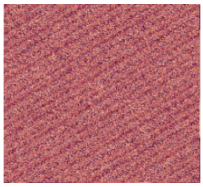

Fig. 2. Scrambled Image

Figure 2 indicate the scrambled 256x256 image after applying 65 rounds of Arnolds Cat Map when the value of p=1 and q=1. The output of the ACM is scrambled and we cannot recognize the original image since all the pixels of an image are repositioned for 65 iterations. Following is the part of Matlab code that perform the scrambling operation, newX=mod((x+y),ImSize(1)); newY=mod((x+2*y), ImSize(2)); im1(newX,newY,:)=im(x,y,:);

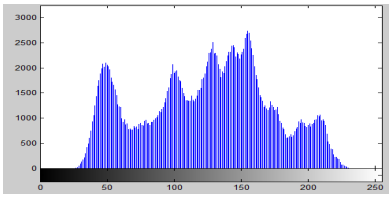

Fig. 4. Histogram of original image

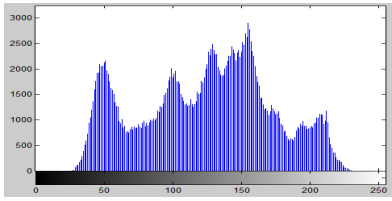

Fig. 5. Histogram of scrambled image

This ACM transformation places pixel at the location $\begin{bmatrix} x \\ y \end{bmatrix}$ to location $\begin{bmatrix} x+y \\ x+2y \end{bmatrix}$. The ACM is used for scrambling the image pixels before sending image for encryption, this provide the additional security. Intensity of the scrambled image remains exactly same as that of original image. In order to recover the image to its original form we need to use the inverse matrix for same numbers of iterations. In this case the inverse ACM transformation is $\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix}$ mod N.

2.1 The Periodicity Property of ACM

The ACM has an interesting property which is explained by Poincaré Recurrence Theorem [7]. The Poincaré Recurrence Theorem states that "**Certain systems will, after a sufficiently long time, return to a state very close to the initial state**" [7].

Fig. 6. Original image

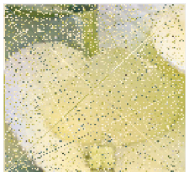

Fig. 7. Recovered image

Figure 7 represents the 124x124 dimension image after 15 iterations of ACM. The recovered image is not exactly same as that of original but is closely similar to the original image.

2.2 Security of ACM

The security of ACM is entirely dependent upon values of p and q, it has been found that for different values of p and q, the scrambling proportion of the same image is different, and also needs different number of iterations for getting recovered image. Number of iterations (Period) required for recovery of the image appears to be random with changing value of p and q. It has been found that periodic behaviour of the ACM makes it weak for the encryption.

2.3 Advances of ACM

In order to increase the security of the Arnold's cat map, many authors have proposed the 3 dimensional Arnold's cat map [2], [5], [6]. In this section we are going to see the two 3D ACM [2], [5] and analyse their behaviour using experimental results.

2.3.1 Implementation1

This section discuss 3D ACM by Hongjuan Liu et al [2] which is improved by introducing two new control parameters c & d. Following is the enhanced ACM

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{pmatrix} 1 & a & 0 \\ b & ab+1 & 0 \\ c & d & 1 \end{pmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \bmod N$$ [3]. This 3D ACM perform the dual

encryption, firstly it performs the shuffling using regular ACM and secondly it will perform the substitution using z component. Using ACM The correlation among the adjacent pixels can be disturbed completely. On the other hand, this 3D ACM can

substitute grey/ colour values according to the positions and original grey/colour values of pixels. 3D ACM for both colour and greyscale image is implemented, following is result for colour image.

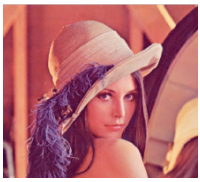

Fig. 8. Original image

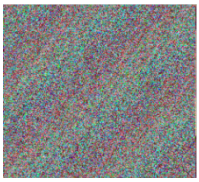

Fig. 9. Scrambled image

ACM was implemented as follows:

$\begin{bmatrix} x' \\ y' \end{bmatrix}$ are the location of pixel after mapping and $\begin{bmatrix} x \\ y \end{bmatrix}$ are location of pixels before mapping. The third parameter inserted is z' which is given by $z' = (c*x+d*y+z) \bmod M$. Here z is the intensity/colour code of pixel before mapping and z' is intensity/colour code of image after mapping [3]. M is the maximum value of intensity of pixel that is M=256.The following part of the Matlab code is used to perform this transformation. im1(newX, newY,:)=mod(c*x+d*y+int32(im(x,y,:)), 256);.The 3D ACM is more secure than that of ACM because of two factors. First, presence of additional constants c and d that can take any random values and secondly ACM can only shuffle the pixel location but 3D ACM can perform the additional substitution and make distribution of colour/gray value uniform [3].

2.3.2 Implementation2

Second 3D ACM was proposed by Zhou Zhe et al.[5], the equation of the 3D ACM is $y = (Ax) mod\ 2^L$[6]. Here $x = [x_1\ x_2\ \dots\ x_n]^t$, $y = [y_1\ y_2\ \dots\ y_n]^t$ and $x_i, y_i \in [0, 2^L - 1]$. A is m*m matrix. Taking value of m=3 we get following matrix [6].

$A=$

$$\begin{pmatrix} 1 & a_{12}+a_{13}b_{23} & a_{12}a_{23}+a_{13}(1+a_{23}b_{23}) \\ b_{12} & 1+a_{12}b_{12}+b_{12}a_{13}b_{23} & (1+a_{12}b_{12})a_{23}+b_{12}a_{13}(1+a_{23}b_{23}) \\ b_{13} & (1+a_{13}b_{12})b_{23} & (1+a_{13}b_{12})(1+a_{23}b_{23}) \end{pmatrix}$$

Though the above 3D ACM is used as S-box by the author, we can also use it for scrambling of the image by multiplying A with $\begin{bmatrix} x \\ y \\ z \end{bmatrix}$ where x and y represent the

pixel location and z represent the intensity/ colour values of the image pixel. Another way is to use this map is by taking R, G, B component of an image pixels as x, y, z and apply 3D ACM. Basically the security of this 3D ACM is huge due to presence of the large number of constant terms. Following are the result obtained by using this 3D ACM for R, G, and B component of an image. Important thing to mention here is, unlike the previous version of 3D ACM this implementation does not provide the scrambling of the image pixels. This implementation only substitutes the new values for pixels R, G, B values. Following is the output of the above implementation.

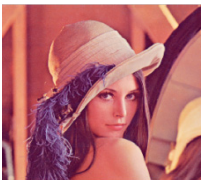

Fig. 12. Original image

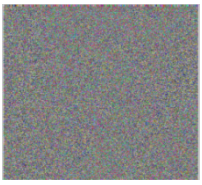

Fig. 13. Scrambled image

The Matlab code is new_color=[red;green;blue]; new1_color =mat*new_color; im1(x,y,:)=mod(new1_color,256);.It is found that intensity of the original image tends to normal distribution which is depicted in figure 16. And all the R, G, B component of the image are uniformly distributed after applying this transformation, this result is depicted in 18, 20, 22. More number of constant terms provides more security than that of previous 3D ACM, but absence of scrambling make it weaker. In this implementation the value of 2^L is 256, because this is the maximum value any pixels R, G, and B component can take. It's been found that intensity of the image undergone this 3D ACM implementation tends to follow normal distribution and R, G, B component of image are distributed uniformly. This uniformity of R, G, B component provide the huge security and make the relationship between original image and scrambled image complex and make the image unrecognizable so that cryptanalyst will not be able to predict it.

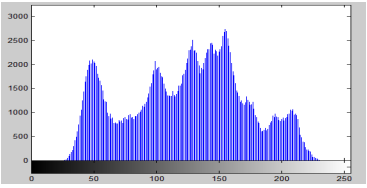

Fig. 15. Intensity histogram before 3D ACM.

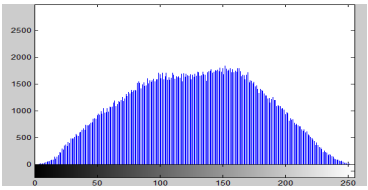

Fig. 16. Intensity histogram after 3D ACM

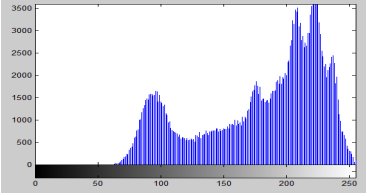

Fig. 17. Intensity histogram of red colour

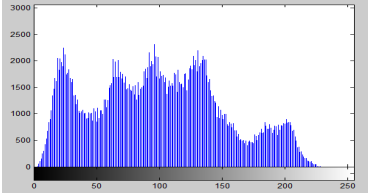

Fig. 18. Intensity histogram of green colour

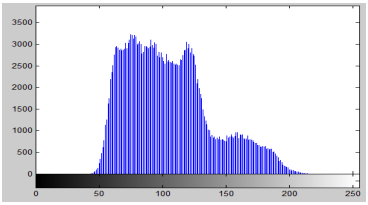

Fig. 19. Intensity histogram of blue colour after 3D ACM

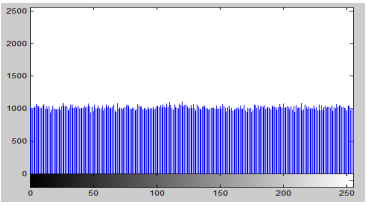

Fig. 20. Histogram of Red, Green, Blue colour

2.4 Summary

Following is the table that compares ACM and 3D ACM according to certain parameters.

Table 1. Comparison between ACM and 3D ACM

Transformation	Substitution	Shuffling	No. Of Constants	Intensity histogram of image
Arnolds Cat Map	No	Yes	2	Unchanged
3D Arnolds Cat Map (Hongjuan Liu et al.)	Yes	Yes	4	Uniform
3D Arnolds Cat Map (Zhou Zhe et al.)	Yes	Implementation specific	6	Tends to normality

3 Inverse Arnolds Cat Map (Inverse ACM)

The image that is scrambled/ encrypted using ACM can be recovered by using inverse ACM transformation and iterating for same number of iterations. The image is scrambled till 15 iteration and then inverse ACM for 15 rounds applied to get following result.

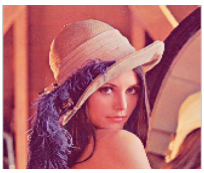

Fig. 23. Original image

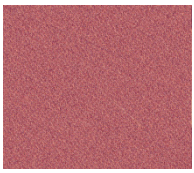

Fig. 24. Scrambled image.

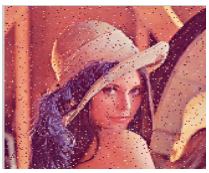

Fig. 25. Unscrambled image

From the result obtained it is found that after applying the inverse transform the original image can be recovered, but there is some amount of loss of image data. It is also found that as number of iterations will increase for ACM transformation more will be loss of image precision after applying the inverse transformation to recover the image.

4 Conclusion

This paper studies the Arnolds Cat Map and 3D Arnolds Cat Map in detail. It's been observed that ACM can do only shuffling of the pixels location. The former implementation of 3D ACM is provided with additional constants c and d was able to shuffle and substitute the pixel values. The later implementation of 3D ACM can be used in two ways firstly we can use it like former implementation or we can use it to substitute the values of R, G, and B component of individual pixel. Hence the conclusion is, 3D ACM is more secure than that of ACM and hence can be successfully implemented in chaotic encryption algorithms. To overcome the periodic property of ACM and 3D ACM we recommend the use of other 3D chaotic maps for encryption after encrypting using 3D ACM. Applying both 3D ACM in cascaded manner can provide the huge level of security.

References

1. Bose, R., Banerjee, A.: Implementing Symmetric Cryptography Using Chaos Function. In: Advanced Computing & Communication Conference (1999)
2. Liu, H., Zhu, Z., Jiang, H., Wang, B.: A Novel Image Encryption Algorithm Based on Improved 3D Chaotic Cat Map. In: The 9th International Conference for Young Computer Scientists (2008)
3. Huang, M.-Y., Huang, Y.-M., Wang, M.-S.: Image encryption algorithm based on chaotic maps. In: Computer Symposium, ICS (2010)
4. Mingming, Z., Xiaojun, T.: A Multiple Chaotic Encryption Scheme for Image. In: 6th International Conference on Wireless Communications Networking and Mobile Computing, WiCOM (2010)
5. Zhe, Z., Haibing, Y., Yu, Z., Wenjie, P., Yunpeng, Z.: A Block Encryption Scheme Based on 3D Chaotic Arnold Maps. In: International Asia Symposium on Intelligent Interaction and Affective Computing (2009)
6. Senthil Arumuga, A., Kiruba Jothi, D.: Image Encryption Algorithm Based On Improved 3d Chaotic Cat Map. In: International Conference on Computational Intelligence and Computing Research, ICCIC (2010)
7. Frazier-Reed, T.: M.I.S.T.: Cat Map, `http://music.calarts.edu/~tcfr33/technology/catmapex.html` (retrieved October 21, 2008)