Spatial Domain Video Encryption using Chaotic Maps

Karan Nair
Eduvance,
Mumbai, India

Janhavi Kulkarni
Eduvance
Mumbai, India

Karan Asher
K.J.Somaiya College of Engg.
Mumbai, India

Vicky Chheda
K.J.Somaiya College of Engg.
Mumbai, India

Jonathan Joshi
Eduvance,
Mumbai, India

ABSTRACT

Digital video is one of the most popular multimedia data exchanged over the internet. Previous cryptography studies have focused on text data. The encryption algorithms developed for text data may not be suitable to multimedia applications because of large sizes of video. Analgorithm is proposed in which a video file is encrypted by considering each frame a colour image. Each video is broken down into its constituent frames. Chaotic mapping algorithms are applied on all the frames and in the temporal domain of the video as well. The algorithm was run on different videos, and results were obtained show improved performance time and good security. Initial comparison against existing methods also shows that encryption time required is less, while recovered plaintext also has fewer distortions.

General Terms

Encryption, multimedia security, video processing

Keywords

Chaotic maps, encryption, multimedia, rectangular map, spatial domain, video

1. INTRODUCTION

Encryption is the process of encoding messages or information so that the original data, or plaintext, cannot be intercepted by an attacker. With the development of both personal computer and internet technology, multimedia data is being used widely in applications like video-on-demand, video conferencing, live event-streaming, forensics, surveillance, military drone feeds. Textual data has its own unique characteristics. Multimedia data has additional redundancies, which gives an attacker extra advantage for deciphering the ciphertext. Traditional block and stream ciphers have been designed to eliminate the statistical redundancies in data. However, video data has its own unique types of redundancies which do not factor into these methods. Video shows spatial redundancy i.e. adjacent pixels in a frame are correlated, as well as temporal redundancy i.e. same pixel in successive frames are correlated. When video is treated as generic data, this double redundancy is not exploited, leading to higher processing time and lower achievable security. It is imperative to exploit these redundancies to make the ciphertext look completely random. In real time applications, it is necessary to have fast encryption algorithms which doesn't cause any lag or delay. The processuses chaotic maps for encryption of the video. Chaotic maps have been widely investigated over the last decade to meet the increasing demand of fast encryption. Due to the properties of high initial valuesensitivity, appearing completely random and having superior performance in terms of speed and complexity [1], chaotic maps are suitable for data encryption. Each frame is considered as an imageand divided into 8x8 macroblocks. Chaotic maps are applied on the differentframes in the video to shuffle the macroblocks. A different chaotic map is applied in the temporal domainto achievegreater diffusion of original data. The algorithm has been implemented in MATLAB to evaluate performance.

2. RELATED WORK

A straightforward, butcomputationally expensive approach for video encryption is to consider it as text data. There are several existing algorithms based on AES/DES/IDEA for secure transmission of video. They are very secure, but very slow owing to the large amount of data a video contains. The paper by P. Deshmukh et al[2]proposes a modified AES algorithm which achieves increased performancebut it doesn't account for correlation between pixels in the temporal domain.Many algorithms have been proposed where encryption of the video is done in the frequency domain. Discrete cosine transform (DCT) is used to convert the video from spatial to frequency domain. In a particular DCT block, most of the energy is concentrated in the DC coefficientsand very few AC coefficients. Thus frequency selective algorithms are used for encryption.The algorithm proposed by C. Narsimha et al[3] performed encryption of the first ten coefficients followed by permutation of DC and AC coefficients. The paper by Changgui Shi et al [4] proposed an encryption algorithm named video encryption algorithm (VEA) which uses simple XOR of the sign bits of the DCT coefficients using a secret key. The main disadvantage of domain conversion is the extensive amount of time needed for conversion. It also fails to account for the quantization losses in the frequency domain due to floating point arithmetic, which leads to artefacting and distortions on recovery of plaintext. They also don't use standard cryptographic algorithms, and hence their security is very low[5].The paper by Lian S. et al [6] proposes to encrypt the run length codes of DCT coefficients in the MPEG compression using chaotic run length algorithm. Sign bits of motion vectors are also encrypted using a different chaotic technique called as security enhanced chaotic stream cipher. Shang F. et al [7] also implemented a similar scheme with the difference that it also encrypts AC coefficients. The paper by Zhaopin Su et al[8] explains different types of video encryption algorithms in detail which include encryption using chaos theory and frequency domain encryption.

3. PROPOSED ALGORITHM

3.1 Chaotic Maps

The idea of using chaos for encryption is certainly not new and can be traced toClaude Shannon's paper in 1952. It suggests mixing of data and transformations which depends on their arguments in a sensitive manner[9].Chaos theory is a scientific discipline that focuses on the study of nonlinear systems that are highly sensitive to initial conditions that is similar to random behaviour and continuous system. They are deterministic yet they can be unpredictable in nature. This highly unpredictable and pseudo-random nature of chaotic output is the most attractive feature that leads to many novel applications[1].

3.1.1 Rectangular Chaotic Maps

The algorithm considers each frame as a colour image. They are nothing but two dimensional matrices of some height and width. Conventional chaotic algorithms like Baker's chaotic map and Arnold's cat map work on square matrices [9][10].

They can be used on a video frame provided the video frame is made square. Since video frames are inherently rectangular in size (aspect ratio ≠ 1:1), we need a rectangular chaotic map for encryption of data. A rectangular chaotic transform has been proposed in [11]which is an extension of the Arnold's cat map. It is given by equation (1). Let $gcd(m,n)$ be the greatest common divisor of *m* and *n*.

$$F : \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \; mod \begin{bmatrix} H \\ W \end{bmatrix} \qquad (1)$$

$$st. \begin{cases} gcd(a, p_h) = 1 \\ b\ (mod\ p_h) = 0\ or\ c(mod\ p_w) = 0 \\ gcd(d, p_w) = 1 \\ gcd\big((ad - bc), p\big) = 1 \end{cases}$$

where H and W are the height and width of the video frame respectively; (x, y) is the original pixel position, (x', y') is the mapped position of (x, y); $p = gcd(H, W)$, $p_h = H/p$and $p_w = W/p$. The matrix $A = [(a, c)^T, (b, d)^T]$ is called the transformation matrix of the chaotic transform.The properties and algorithm for effectively generating the coefficient matrix is elaborated in[11].*Property*Suppose A_0 is a coefficient matrix of 2D rectangular transform. For any positive integers e and f satisfying $gcd(e, H) = 1$and $gcd(f, W) = 1$, let

$$A_1 = \begin{bmatrix} e & 0 \\ 0 & f \end{bmatrix} \times A_0 \,, A_2 = A_0 \times \begin{bmatrix} e & 0 \\ 0 & f \end{bmatrix} \qquad (2)$$

Then A_1andA_2 are coefficient matrices of the 2D rectangular transform. Based on this property a coefficient matrix can be created using random integers which depends on the height and width of the frame. The random integers in the algorithm are generated using a pseudo random number generator (PRNG) which is seeded with a value obtained from a password entered by the user.

3.1.1.1 Generation of Transformation Matrix

(1) Generate 8 random integers$r_i (1 \le i \le 8)$ using a seed value. The last random number is i.e. r_8 is given as output as a new seed value. Set $p = gcd(H, W)$and$p_w = W/p$. Generate two integer sequences $\{h_0, h_1 \dots, h_{l-1}\}$ and $\{w_0, w_1 \dots, w_{l-1}\}$ such that $gcd(h_i, H) = 1$and$gcd(w_i, W) = 1, (1 \le i \le l)$. $l$is set to 40 in this algorithm.

(2) Let$b_0 = r_5$, $c_0 = p_w \times r_6$and $j = r_7(mod\ l)$. Construct a special matrix A_0 as,

$$A_0 = \begin{bmatrix} 1 & b_0 \\ c_0 & b_0 c_0 + w_j \end{bmatrix}$$

(3) Let $j_i = r_i(mod\ l),\ (1 \le i \le 4)$. Then calculate the final coefficient matrix using property described in (2) as follows,

$$A = \begin{bmatrix} h_{j1} & 0 \\ 0 & w_{j2} \end{bmatrix} \times A_0 \times \begin{bmatrix} h_{j3} & 0 \\ 0 & w_{j4} \end{bmatrix}$$

3.2 Implementation of Chaotic maps

The aim of using chaotic maps is to break the correlation amongst pixels in the video. In an image, chaotic maps are applied over pixel positions. In a video, there is a vast amount of data present. Permutation of pixels is a very computationally intensive process. Instead each frameis divided into macroblocks of size 8x8. A macroblock is smallest quantity in the video the algorithm is going to deal with instead of a pixel. Each macroblockis treatedakin to a pixel in the video and chaotic maps are used to permute macroblocks in each plane.One interesting feature of chaotic map is that if the map is applied over a sufficient number of times, the original matrix will be obtained. The map is applied iteratively. If the iterations is a value which is the period of the map, then it will bring back the original information, or if it closer to the period, the information might become easy to decipher. Hence we need to choose the number of iterations carefully before applying the map. It is difficult to predict what the period will be as it depends on the size of the map as well as the seed to the PRNG. Fig 1 and fig 2 shows the relation of chaotic map with iterations based on different value of seed. The graphs are nothing but a plot of cross correlation between the original matrix and its mapped version for a given number of iterations. Greater the value of the cross correlation, greater is the visual similarity between encrypted and original frame. Hence to avoid generation of a map with high correlation (low security), the following precautions are sufficient

(1) The number of iterations is a relatively large prime, 47 in the algorithm. If the value is too high, then the computation time will increase, as well as it could be a period of the map. If it is too small, then the level of randomness will not be sufficient if the map has a large period. If the value is a prime, it reduces the likelihood that it will be a multiple of the period.

(2) A new map, with a new, pseudo-randomly generated seed is used for each plane. This ensures that even if one of the maps has a low level of security, data will get scrambled in some other.

The graphs shown in fig 1 and fig 2 below plot the cross correlation of a mapped image to the original image, for different seed values and different dimensions of the map. It shows the periodic nature of the chaotic maps.

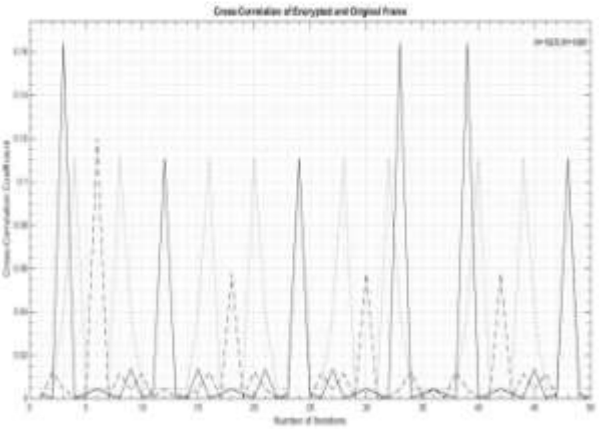


Fig 1: Cross Correlation for frame size1920x1080

3.2.1 Applying maps on frames

As mentioned earlier, every single frame of the video file is converted into planes having macroblocks of size 8x8. If the video resolution is$H \times W$, the planes which contain the macroblocks will have resolution $\left(\frac{H}{8}\right) \times \left(\frac{W}{8}\right)$.

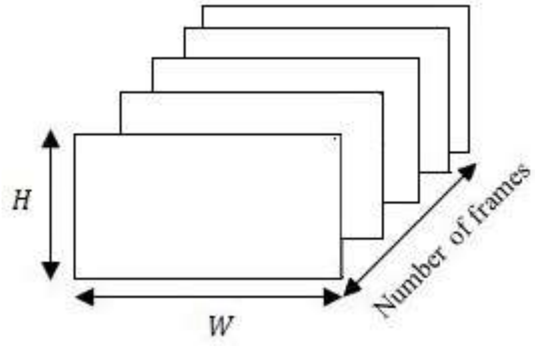


Fig 3: Representation of a video

The main intention is to apply chaotic maps on two domains of the video. The spatial domain, with resolution$\boldsymbol{H} \times \boldsymbol{W}$ and the temporal domain, resolution of $\boldsymbol{H} \times$ $\boldsymbol{number\,of\,frames}$. The mapping in these twodomains will introduce sufficient randomness in the video. Another advantage of this technique is that macroblocks in different parts of video are mixed and permuted many times to break the correlation between the macroblocks.Fig 4. Below shows a visual representation of the macroblocks in each frame of the video in a 3-D manner. The chaotic maps effectively shuffle the cubes (macroblocks) with each frame as well as between frames. This is somewhat similar to the process by which a Rubik’s cube would be shuffled.

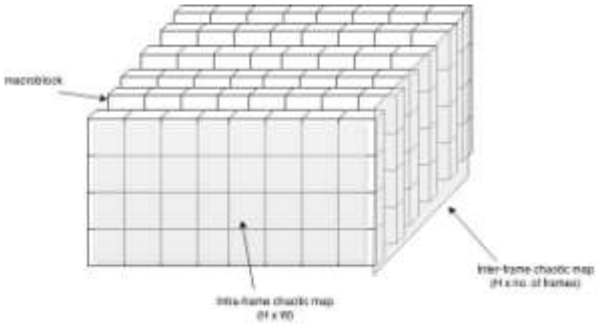


Fig 4: Mapping of frames

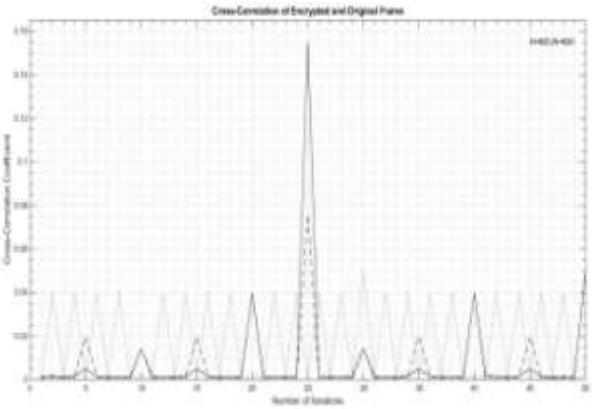


Fig 2: Cross Correlation for frame size800x600

Chaotic map of dimensions $H \times$ $number\,of\,frames$ is created using the same seed as in the case of previous maps. In a similar fashion, macroblocks of the RGB planes are shuffled in the vertical-temporal plane. Thus a total of 2 maps are created for encryption of the whole video. The maps are heavily dependent on the password entered by the user.Simple key expansion algorithms can be used to generate the random values for creating the chaotic maps.

3.3 Key Generation

Key generation is the process of producing keys for encryption and decryption. If the same seed is shared then it becomes a private key. Since the chaotic map heavily depends on seed values, it is necessary to have a mechanism which will generate the same seed for encryption and decryption. A password which is entered by the user has been used to form the seed. Modular exponentiation is used to expand the key into larger values. Exponentiation causes the key value to get larger, while the modular operation ensures it remains within acceptable limits of the PRNG. The inbuilt RNG in MATLAB only supports an input seed size of 2^{32}. Hence this forms an upper bound for the keys produced during the testing process. However, larger key spaces can be used by using a more robust RNG.

4. RESULTS AND PERFORMANCE ANALYSIS

All the blocks in the algorithm were implemented in MATLAB for testing and performance analytics. Each algorithmic step was run individually and tested for performance, as well as executed together for obtaining output videos and checking security. In the images shown below in fig. 5,6,7,the same frame in the original and encrypted video show little to no correlation. The intra-frame and inter-frame shuffling lead to a decorrelation in the spatial as well as temporal domains.Existing work, like the algorithm proposed by Rajpurohit. J et al[12]uses a key of a certain length to scramble frames of a video. This makes the encryption process extremely fast, but doesn’t account for the spatial redundancy in the frame and breaks only temporal redundancy. The algorithm proposed by us successfully breaks both the redundancies to make the video contents completely scrambled.

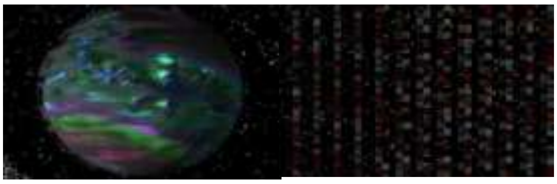


Fig 5: Original and encrypted frame for a 320x200 frame

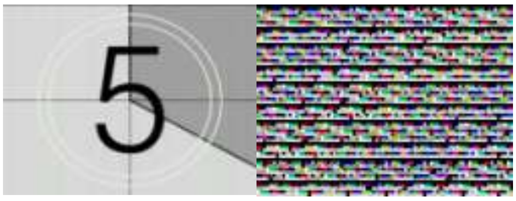


Fig 6: Original and encrypted frame for a 640x480 frame

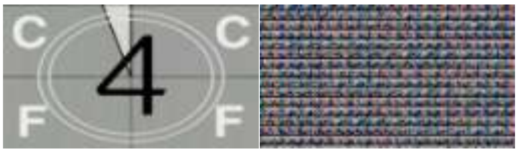
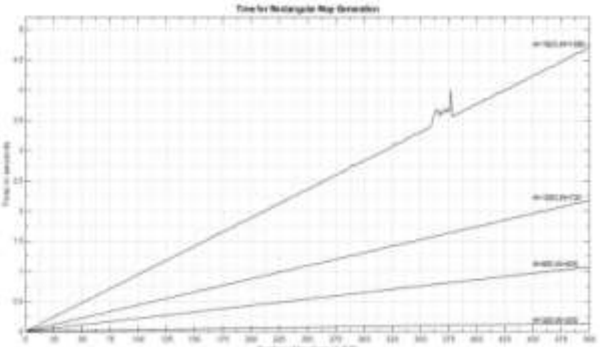


Fig 7: Original and encrypted frame for a 1280x720 frame

Video encryption is extremely processor intensive and time consuming because of the nature of the data, making it extremely challenging to work with. The process can be analyzed as a sum of its parts. The main section of the algorithm is the rectangular mapping. Following is a summary of the performance measured. All the codes were run on an Intel Core i7-3610QM running at 2.3 GHz with 6 GB of RAM. Figures 6,7 show the original and encrypted frames of some test videos on which the algorithm was performed.

4.1 Chaotic maps

Generation of the rectangular map depends on two factors, the dimensions of the frame and number of iterations. The graph in fig 8 shows the time for calculation of the map for different sizes and for iterations ranging from 1-500. Time taken is linearly related to the number of iterations. The larger the frame, the greater is the slope of the graph.



.

Fig 8: Time for rectangular map generation

4.2 Processing time

The processing time for the entire encryption process, for the videos in Fig. 5, 6 & 7, have been tabulated.

Table 1.Execution time for encryption and decryption

Sr. no.	Dimensions	Frame count	Encryption time	Decryption time
1	320x200	98	16.91s	16.10s
2	640x480	168	115.10s	115.9s
3	1280x720	168	624.95s	625.79s

The above processing time shows great promise for a fast, secure video encryption cipher. Using chaotic maps requires fewer rounds of encryption and therefore provides better security for lower processing times. This method can also be parallelized to achieve even faster encryption.

Further, by not working in the frequency or any other transform domain, we avoid the errors that creep into the decrypted video due to rounding errors in limited precision floating-point arithmetic. As all operations are purely in spatial domain, the decrypted video will have no errors or artefacting. This is especially useful in sensitive video applications like surveillance, where degradation in video quality can compromise the system function.

5. SCOPE FOR FUTURE WORK

The method discussed in this paper is in its nascent stages, and has a lot of scope for future work. Currently a unique map is only applied to each frame. The RGB channels in a color frame are all mapped in the same way. RGB planes within the frame can be mapped with separate chaotic maps. This can improve security as it further decorrelates the data. The algorithms for generation of chaotic maps as well as permutation of the macroblocks are hardware implementable. Hardware implementation can boost processing speed significantly. Mechanisms for integrity and authenticity check can be added using hashing techniques. This can be made to check for damaged or compromised nodes in the transmission network for the video. Further improvement in speed can be achieved by parallelization of the different processes. For example, map generation and macroblock permutation can be processed simultaneously. The mapping has been performed for entire video. Alternatively, videos can be broken down into chunks of more manageable length. Each chunk can be processed in parallel to other chunks. This will allow marked improvement in real time video encryption. Security can also be improved by providing some encryption to macroblock contents. This has not been performed currently due to the excessive increase in processing time. However with correct optimization, additional security can be achieved.

6. CONCLUSION

The encryption method described in this paper makes better use of redundancies specific to video data. It exploits them to encrypt the data faster than existing methods. The complexity involved is very less as compared to other algorithms referred. It has a lot of scope for future work and can be possibly be implemented in real time with suitable buffering and parallelization techniques. It is also important to note, that for the purposes of this paper, all algorithms have been applied to RGB frames. However if compressed video formats like MPEG are to be used, the same principles can be applied to the IPB or any other similar frame structure of the video format.

7. REFERENCES

[1] Salleh, M., Ibrahim, S., Isnin, I.F. 2003. Image Encryption Algorithm Based on Chaotic Mapping.

[2] Deshmukh, P., Kolke, V. 2014. Modified AES based Algorithm for MPEG Video Encryption.

[3] C.NarsimhaRaju, Srinathan, K., Jawahar, C.V. 2008. Real-Time Video Encryption Exploiting Distribution of the DCT coefficients. International Institute of Information Technology, Hyderabad.

[4] Shi, C., Bhargava, B. 1998. An Efficient MPEG Video Encryption Algorithm. Purdue University.

[5] Igorevich, R.R., Yong, H., Dugki Min, Eunmi Choi. 2010. A Study on Multimedia Security Systems in Video Encryption.

[6] Lian, S., Sun, J., Wang, Z., Dai, Y. 2004. A Fast Video Encryption Scheme Based on Chaos. International Conference on Control, Automation, Robotics and Vision, China.

[7] Shang, F., Sun, K., Cai, Y. Central South Uinversity, China. 2008. An Efficient MPEG Video Encryption based on Chaotic Cipher. Congress on Image and Signal Processing.

[8] Zhaopin Su, Lian, S., Zhang G., Jiang, J. 2011. Chaos-Based Video Encryption Algorithms. Hefei University of Technology, China.

[9] Jiri Fridrich. 1998. Symmetric Ciphers Based on Two-Dimensional Chaotic Maps. International Journal of Bifurcation and Chaos.

[10] Peterson, G. 1997. Arnold’s Cat Map

[11] Zhang, X., Fan, X., Wang, J., Zhao, Z. 2014. A Chaos Based Image Encryption Scheme Using Rectangular Transform and Dependent Substitution.

[12] Rajpurohit, J., Dr. Khunteta, A. 2013. A Scalable Frame Scrambling Algorithm for Video Encryption. IEEE conference on Information and Communication Technologies.