International Journal on
Advanced Science
Engineering
Information Technology

Vol. 2 (2012) No. 1
ISSN: 2088-5334

Video Encryption Based on Chaotic Systems in the Compression Domain

Ali Abdulgader, Kasmiran Jumari, Mahamod Ismail, Tarik Idbeaa

Department of Electrical, Electronic & Systems Engineering, University Kebangsaan Malaysia, Selangor, Malaysia
Email : a752006@yahoo.com, kbj@eng.ukm.my, mahamod@eng.ukm.my, tidbeaa@yahoo.com

Abstract— **With the development of the internet and multimedia technology digital video encryption has attracted a great deal of research interest in the recent few years in applications. In this paper, we propose a method to encrypt video data. The proposed algorithm is based on the MPEG video coding standard. It selectively encrypts some DCT coefficients in the I frame, B frame and P frame in MPEG video compression by using chaotic systems. The key in this paper is chaotic sequence based on logistic mapping. It can produce the pseudo-random sequences with good randomness. The experimental results based on chaotic maps prove the effectiveness of the proposed method, showing advantages of large key space and high-level security. The proposed algorithm was measured through a series of tests and achieved good results. The results indicate that the algorithm can be implemented for video encryption efficiently and it provides considerable levels of security.**

Keywords— **MPEG video Compression, video encryption, chaotic systems.**

I. Introduction

Internet and digital media applications are rapidly growing which make the requirement of secure transmission of data correspondingly increase. The security of digital images and videos has become more and more important. Since digital video transmission systems usually includes a compression part that aims to reduce the transmitted bit rate, the cryptography techniques have to be carefully designed to avoid probable adverse impact on the compression efficiency, and on the compression format. Several video encryption techniques have been proposed in the past decade. These techniques can be classified into three types: Spatial Domain, Bitstream Domain and Frequency Domain. While encryption standard algorithms like AES (Advanced Encryption Standard) or DES (Data Encryption Standard) can be used to encrypt the video data, it has two main drawbacks. First, since multimedia data is usually large and requires real-time processing, DES and AES incur significant overhead. Second, these techniques hide the synchronization information in the video stream, making it impossible to adaptively shape the stream to match available network resources and making it harder to recover from transmission errors in the network. This paper proposes a selective encryption algorithm to encrypt video data and increase level of security. The MPEG video is composed of a sequence of Group of Pictures (GOPs) [15]. Each GOP consists of three types of frames: I frames (Intra coded frames), P and B frames (forward predictive and bi-directional predictive coded frames known as Inter frames). The typical steps performed in MPEG video compression include motion estimation, DCT transform, quantization, and entropy encoding as shown in Fig.1.

II. Previous Work

Several video security techniques based on the MPEG video compression standard have been proposed [3], [4], [12]. These techniques can be classified into three types: encryption techniques before, during and after compression domain. In the encryption schemes before compression domain, the encryption technique applies to the original data directly. In the encryption schemes after compression domain, the encryption technique applies to encrypt the code words of compressed Bitstream. To reduce the amount of processing overhead, the encryption schemes during compression domain technique has been proposed, which exploits the results of DCT or DWT transformation and quantization stages of the compression process.

The encryption schemes before compression domain: there are some encryptions algorithms have been proposed to encrypt the video in the RGB colour space using secret linear transformations before video compression. It can keep the compression efficiency of the video codec. But as shown in Li and Bhargava scheme [1], the scheme is not secure enough against brute force attacks because its key space is not large.

Another encryption schemes before compression domain was proposed by Socek and Magliveras [2]. It encrypts the raw video before compression by using the sorting permutation to keep the compression efficiency. In this scheme the first frame is compressed by the video codec and transmitted over the secure channel. Then the sorting permutation of the first frame is applied to the second frame. The sorting permuted pixels of the second frame are encoded by the video codec and transmitted over the insecure channel. After that the sorting permutation of the second frame is applied to the third frame. This process is repeated till the end of the video sequence. The scheme is not secure enough against known-plaintext attacks because the adversary can recover all frames that follow the known frame.

The selective encryption schemes after compression domain select and encrypt some important parts in compressed video bitstream. Some selective encryption algorithms have been proposed to encrypt only header information of bitstream as in the SECMPEG algorithm [6]. The SECMPEG algorithm contains different levels of security. At the first level the SECMPEG algorithm encrypts only the headers information from the sequence layer to the slice layer. At the second level, the SECMPEG algorithm encrypts parts of the I-blocks and the headers information. At the third level, SECMPEG encrypts all I-frames and all I-blocks in P-frames and B-frames. Finally, the fourth level SECMPEG algorithm encrypts the whole MPEG sequence. SECMPEG algorithm describes a new syntax which is Conflict with the MPEG syntax. It cannot be used for the perceptual encryption because headers contain the information .The MPEG decoder cannot recognize the encrypted bitstream.

Another selective encryption algorithm is the Aegis algorithm by Maples and Spanos[5], [12]. The Aegis algorithm increases only the number of I-frames and encrypts of it the bitstream. These schemes are not compatible with standard MPEG for the same reason in SECMPEG algorithm.

Another encryption schemes after compression domain was proposed by Qiao and Nahrstedt [4]. They consider the MPEG stream as a sequence of bytes. The algorithm divides a chunk of the MPEG video stream into two byte lists, an odd and an even list. Then the XOR operation performs to encrypt these lists with a generated binary key. This algorithm is 47 % computationally faster than the DES (data encryption standard). This algorithm is suitable for some real-time applications because the security level is good enough for some very sensitive applications like video on demand. But it has drawbacks; it still needs to go through all I-frames, which is computationally expensive.

Another selective encryption scheme was proposed by Wen, M. Severa, W. Zeng [7]. Their algorithm permutes the codewords of the MPEG bitstream. However, this method loses some error resiliency. Moreover, in order to get code words, the algorithm has to analyze the syntax of the MPEG bitstream. Therefore, it increases the complexity of the algorithm.

Different kinds of encryption schemes during compression domain have been proposed. These schemes selectively encrypt the resulting data of the DCT transformation process or motion estimation process.

Shi and Bhargava suggest some encryption schemes, which take the DCT coefficients of every block and encrypt the sign bits of DC and AC coefficients and all the motion vectors [8] - [10]. The schemes XOR these sign bits with a generated binary key. Although this scheme is simple and keeps the bitstream format-compliant not changed. The encrypted bitstream is weak when the attacker gets some plaintexts of the encrypted video data.

Another encryption scheme during compression domain was proposed by Tang, which is called Block Shuffle [11]. In this algorithm the DCT coefficients in 8 x 8 blocks in video frame are shuffles by using a permutation table instead of the original zigzag order of MPEG. Few processing overhead was presented from this algorithm; it replaces the original zigzag order with a random order. Therefore, a large amount of bit overhead occurs after the entropy coding stage.

III. Proposed encryption and decryption algorithm

Video encryption schemes aim to make the digital video unknown to unauthorized users. In this paper, the process of encrypting video data is based on selection of the DCT coefficients to be encrypted. The steps required in the proposed encryption process are as the following. The video frames and secret keys are input into the encryption process. The output of the encryption process is encrypted video frames as shown in Fig.3.

A. Chaotic Mapping and Key Generation

The chaotic encryption can be developed by using properties of chaos including deterministic dynamics, random behaviour and non-linear transforms. The key in this paper is chaotic sequence based on logistic mapping [13]. Chaotic systems have the character of pseudo-randomness and they are sensitive to initial conditions; it is also a nonlinear series having a complicated structure and is difficult to predict. A dynamical system that is researched widely is Logistic mapping, defined as:

$$x_{n+1} = f\left(\lambda, x_n\right) = \lambda x_n \left(1 - x_n\right) \quad \lambda \in \left[0,4\right] \tag{1}$$

λ is known as a parameter. Chaotic sequence of this video encryption system is generated by Logistic mapping when $\lambda = 3.65399$. The input and output of the Logistic model are in the range (0, 1). In this paper we propose a pseudo random number generator by using two chaotic logistic maps

$$x_{n+1} = [\lambda \; x_n \left(1 - x_n\right)] \bmod 1 \tag{2}$$

$$y_{n+1} = [\lambda \; y_n (1 - y_n)] \bmod 1 \tag{3}$$

$$z_{n+1} = [\lambda \; z_n + y_{n+1} + x_{n+1}] \bmod 1 \tag{4}$$

starting from random independent initial conditions $(x_0, y_0, z_0 \in (1,0)$ and $(x_0 \neq y_0)$ .The pseudo random number sequence is generated by alternative outputs of both the chaotic logistic maps. Mod 1 limits the data values which are above 1 to the range of (0, 1).

Another key used in the permutation stage to change the position of DC components is based on this equation:

$$x = (1 + 77 * i + 3) \bmod M \tag{5}$$

where M is the number of DC components in every frame and x is the new position of i^{th} DC components.

B. Algorithm Steps

1. When a frame of size $N \times N$ pixels enters, it is divided into non-overlapping blocks of size 8 x 8.
2. Then the DCT transform and quantization processes are applied to each block. The result is blocks of DCT coefficients.
3. Blocks of the DCT coefficients are arranged in a 1-D zig-zag sequence to perform the encryption and encoding process.
4. The DC and AC coefficients are selected to be encrypted.
5. The secret key is generated by using the chaos scheme.
6. The DC and some AC coefficients are encrypted by using XOR operation with the secret key.
7. The positions of the encrypted DC components between blocks are changed in video frame by using the key generated from the pseudo-random sequence.
8. Entropy coding is done. The output is encrypted mpeg bitstreams.

The decryption process is the reverse of the encryption process described above. The decoder takes the encrypted mpeg bitstreams as input to entropy decoding; then the decryption process is applied to each frame. By generating the secret key using the same chaos scheme the DC and AC components are decrypted. The output of the decoder is an original mpeg video frame.

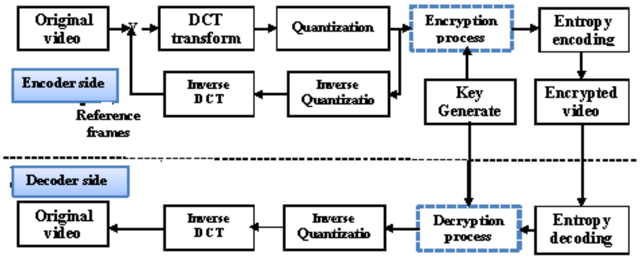

Fig. 1 The block diagram of video compression and encryption/decryption algorithm [15].

IV. RESULT AND DISCUSSION

The proposed encryption process described above has been tested with several video files of the size 5.55MB, 150 frames which are in uncompressed format. Fig. 2 show one of the original video frames named Suzie [14].

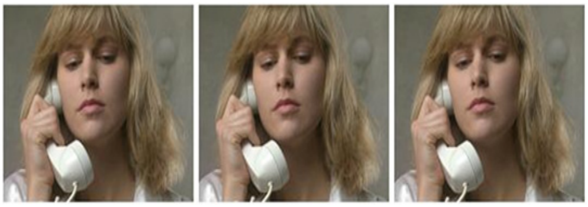

Fig. 2 Example of an original video frame.

The original video frames are first divided into 8 x 8 non-overlapping blocks and the DCT transform is applied on each block as described in section 2. Then the resulting 2-D transform coefficients are re-arranged into a 1-D sequence by using a zigzag technique. The result after the zigzag is one high frequency coefficient and *n* low frequency coefficients. The DC and some AC components of DCT coefficients are chosen to be encrypted because these components have large energies and encrypting these components makes the frame unintelligible. The video frame is encrypted using a secret key.

A. The DC Components Encryption and Permutation

After applying the encryption process only to the DC coefficient in each frame we have obtained the results as shown in Fig. 3 (a). The results show that, if only the DC component has been encrypted, it is very easy to access the visual information of the frame by simply replacing the encrypted DC coefficient with some constant values (for example 0 byte) as shown in Fig. 3 (b). This is because the DC coefficient is of high frequency. In this case, to increase level of security, the permutation technique takes place with encryption process by using different secret keys to change the position of the DC coefficient in each video frame. The results show that the information of the original frame is not visible as shown in Fig. 3 (c).

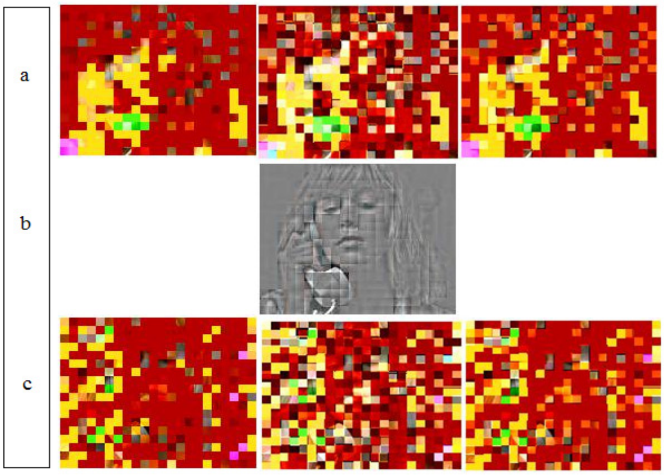

Fig. 3 Example of (a) Encryption of the DC component. (b) DC component with constant value. (c) Encryption and permutation of the DC component

B. The Selective Encryption of the DC and AC components

To increase the level of security in our algorithm, the AC component is selected to be encrypted. After applying the encryption process only to the AC coefficients in each frame and permuting the DC component without encrypting it, we have gotten results as shown in Fig. 4(a & b). Because DC coefficients are kept unencrypted, some details of the contents in the frame can be recognized. To increase the security the DC component must be encrypted also as shown in Fig. 4(c). Fig. 4(d) shows an example of the decrypted video frames.

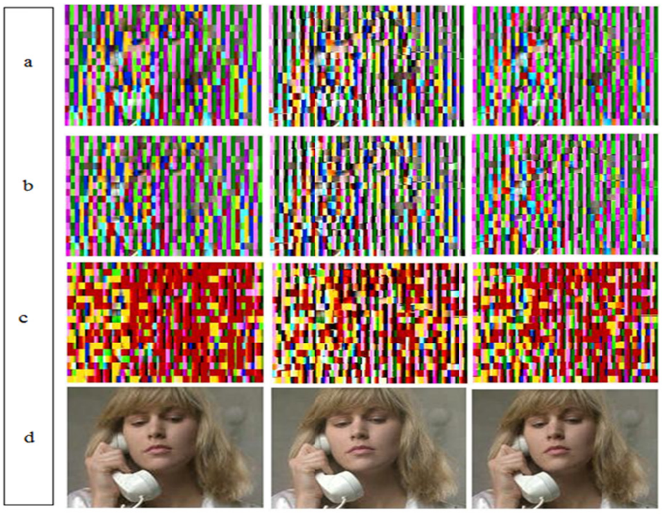

Fig. 4 Example of: (a) AC components encryption.(b)The permutation of DC components and encryption of all AC. (c) Encryption of DC and AC components in all frames. (d) Decrypted video frames.

C. Security Analysis

In this part, we discuss security analysis and performance of the proposed video encryption algorithm such as key space analysis and PSNR (peak signal to noise ratio) and the HVS (Human Visual System) to prove that the proposed algorithm is secure.

1) *Key Space Analysis*: The strong point of the proposed algorithm is the generation of the key sequence by using chaos mapping. The key space should be large to make brute-force attack unfeasible. In the proposed algorithm, we use 2 keys; one is for encrypting DCT coefficients and another key is used for the permuting of DC components of the video frame. For the first process the initial condition $(x_0, y_0, z_0 \in (1,0)$ is considered as the secret key because it is used to generate key sequences. For the permutation process we use another equation to generate the key, depending on the number of DC components in each video frame. With any change in the initial condition of the chaotic maps it will produce different keys; then it is most difficult to decrypt the video frames without using the same initial condition.

2) *Performance Analysis*: two factors have used to estimate the performance of video encryption algorithm. These factors are the PSNR (peak signal to noise ratio) and the HVS (Human Visual System). In our case, the PSNR was used not for measuring the quality for the whole video frames but it was calculated for all frames to measuring the different between original video frames and the encryption video frames. The mathematical expression has approved that our proposed algorithm is very useful. Where PSNR of a frame will be defined as:

$$PSNR = 10\log\left(\frac{255^2}{MSE_f}\right) \qquad (6)$$

$$MSE_f = \frac{1}{N}\sum_{n=0}^{N-1}\left(x_f(n) - \hat{x}_f(n)\right)^2 \qquad (7)$$

where MSE denotes the mean square error between the original video frame x_f and the encrypted video frame $\hat{x}_f$

Fig.5 shows the relationship between number of video frames and MSE and PSNR. The PSNR for original video frames before encryption process was in the range between 29.88 dB up to 54.03 dB for each video frame and for all the encrypted frames the PSNR is given between 26.44 dB up to 27.3 dB. In most cases, the correlation between the pixels in the original and encrypted frame is relatively big. The other factor that used as mentioned to investigate the performance of the implemented method was HVS. In this method, HVS is used as an indicator to measure the visibility in encrypted frame where as shown in Fig.4 when applying encryption process to AC components only, some details in the encrypted frame can be recognized. In other hand, applying encryption process and permutation approach to DC and AC components make the details in the encrypted frame invisible.

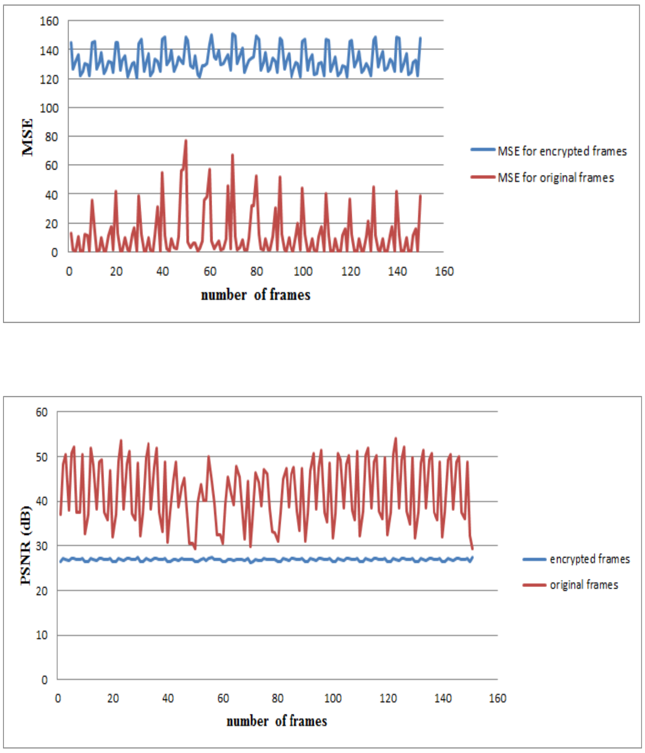

Fig. 5 The MSE and PSNR of the original video frames and encrypted frames.

V. Conclusions

In this paper we have proposed a video security scheme, which includes the encryption and permutation methods. Using the chaotic system in the digital video encryption provides greatly increased safety parameters in the encryption video algorithm, because of the sensitivity of the chaotic system to the initial condition. The encryption and decryption are done at I, P and B frames by XOR operation with secret keys. The proposed scheme encrypts the DC and AC components of MPEG video sequences and provides enough security to video frames. In addition, it is most difficult for any cryptanalyst to attempt to decrypt the MPEG video without permission.

Acknowledgment

The authors would like to acknowledge the support from the Department of Electrical, Electronic & Systems Engineering for this work and also to thank the anonymous reviewers for their constructive comments. This project is supported by University Grant fund UKM-OUP-ICT-36-182/2010.

References

[1] Li S, Chen G, Cheung A, Bhargava B, Lo KT. "On the design of perceptual MPEG-video encryption algorithms*". IEEE Transactions on Circuits and Systems for Video Technology* .2007.

[2] Socek D, Magliveras S, C´ ulibrk D, Marques O, Kalva H, Furt B. "Digital video encryption algorithms based on correlation preserving permutations". *EURASIP Journal on Information Security,* January 2007.

[3] P. Melih and D. Vadi, "A MPEG-2-transparent scrambling technology," *IEEE Trans. Consum. Electron.* vol.48, no.2, pp.345–355, May 2002.

[4] L. Qiao and K. Nahrstedt, "Comparison of MPEG encryption algorithms," Int. J. Compus. Graph., Special Issue on Data Security in Image Communications and Networks, vol.22, no.3, pp.437–448, 1998

[5] G.A. Spanos and T.B. Maples, "Security for real-time MPEG compressed video in distributed multimedia applications," *IEEE 15th Annual International Conference on Computers and Communications*, pp.72–78, 1996.

[6] J. Meyer and F. Gadegast, "Security mechanisms for multimedia data with the example MPEG-1 video," http://www.gadegast.de/frank/doc/secmeng, 1995.

[7] J. Wen, M. Severa, W. Zeng, M. Luttrell, and W. Jin, "A format compliant configurable encryption framework for access control of multimedia," *IEEE Workshop on Multimedia Signal Processing,* pp.435–440, Cannes, France, Oct. 2001

[8] C. Shi and B. Bhargava, "Light-weight MPEG video encryption algorithm," Multimedia 98, pp.55–61, 1998.

[9] C. Shi and B. Bhargava, "An efficient MPEG video encryption algorithm," *17th IEEE Symp. on Reliable Distributed Systems, IEEE Computer Society*, pp.381–386, West Lafayette, Indiana, USA, Oct. 1998.

[10] C. Shi and B. Bhargava, "A fast MPEG video encryption algorithm," ACM Multimedia'98, pp.81–88, 1998.

[11] L. Tang, "Methods for encrypting and decrypting MPEG video data efficiently," Proc. Fourth ACM Int. Multimedia Conf., pp.219–229, 1996

[12] Shiguo Lian, *Multimedia content encryption: Techniques and application,* 1st ed., Taylor and Francis Group,Boca Raton. London, New York, 2009.

[13] http://en.wikipedia.org/wiki/List_of_chaotic_maps

[14] http://trace.eas.asu.edu/yuv/index.html.

[15] L.Hanzo,P.J.Cherriman and J.Streit, *Video Compression and Communications From Basics to H.261, H.263, H.264, MPEG4 for DVB and HSDPA Style Adaptive Turbo-Transceivers,* 2nd ed,John Wiley & Sons Ltd West Sussex PO19 8SQ, England,2007.