ENHANCING VIDEO RECOVERY OF WATER-MARKED VIDEO AFTER ENCRYPTION

R.Vignesh, V.Prabhakaran, Dr.P.C.Kishore Raja, S.Pavithra

Abstract— The multimedia technology has a rapid development in various fields like medical, commercial, and defence. Therefore, security and privacy has become an important. In this paper the zigzag method based on block scrambling for the process of encryption and the compound mapping function is used for data hiding in encrypted video. The concept of applying the algorithm is explained in detail and its performance is also analysed. Its various parameters like performance in terms of speed and strength of encryption are compared with the original zigzag algorithm. The result enhances that the block based scrambling approach performs better and data recovery is lossless and highly privacy.

Index Terms— Video encryption, zigzag scrambling, video encryption, compound mapping, data recovery.

1 INTRODUCTION

The internet speed is becoming higher and higher, information passing through video or MPEG format becomes more popular than text or image. At the same time to ensure the MPEG data security the necessity of effective encryption algorithm becoming higher and higher.

Conventional cryptographic algorithms, which generally aim at encrypting text data, however, are not well suited for video encryption. This is due to the fact that conventional cryptographic algorithms cannot process the large volume of video data in real-time. Moreover, it is almost impossible to adapt them to special video application paradigms which pose special requirements that are never encountered when encrypting text data. For example, in video on demand (VoD) applications it is desired that the encrypted multimedia data are still partially perceptible still encrypted in order to be previewed for the purchase of the high-quality versions of the multimedia products and for the purpose of security and identity we are embedding the data in to the encrypted video This perceptual encryption and data hiding requires specific algorithms for encrypting the video data [1]. In some existing joint data-hiding and encryption schemes, a part of cover data is used to carry the additional message and the rest data are encrypted. For example [7], the intra-prediction mode, motion vector difference and signs of DCT coefficients are encrypted, while a watermark is embedded into the amplitudes of DCT coefficients. In [8], the cover data in higher and lower bit-planes of transform domain are respectively encrypted and watermarked. In [9], the content owner encrypts the signs of host DCT coefficients and each content-user uses a different key to decrypt only a subset of the coefficients, so that a series of versions containing different fingerprints are generated for the users. In these joint schemes, however, only a partial encryption is involved, leading to a leakage of partial information of the cover.

Furthermore, the separation of original cover and embedded data from a watermarked version is not considered. In [10] and [11], each sample of a cover signal is encrypted by a public-key mechanism and a homomorphic property of encryption is exploited to embed some additional data into the encrypted signal. But the data amount of encrypted signal is significantly expanded and the computation complexity is high. Also, the data embedding is not reversible. The current section deals with the basics of the MPEG video and the encryption methodologies. The following sections explain how and why the methodology used in this paper was arrived upon. The later sections deals in detail of the Zigzag algorithm and the block scramble algorithm also deals with data hiding the encrypted image and recovering the data in the receiver side without any loss. Towards the end the results obtained are displayed and discussed.

- *Mr. R.Vignesh is currently working as Assistant Professor in Electronics and Communication Engineering at Saveetha University, India, PH-9543724732 . E-mail: vigneshr79@gmail.com*
- *Mr.V.Prabhakaran is currently working as Assistant Professor in Energy and Environmental Engineering at Saveetha University, India, PH – 9884634252. E-mail : prabhakaranprof@gmail.com*
- *Dr.P.C.Kishore Raja is currently working as HOD & Professor in Electronics and Communication Engineering at Saveetha University, India, PH – 9444011815. E-mail : pckishoreraja@gmail.com*
- *Mrs. S.Pavithra is currently working as a Assistant Professor in Electrinocs and Communication Engineering at Saveetha University, India, PH – 9884246585. E-mail : pavithrra1286@gmail.com*

II. PROPOSED METHOD

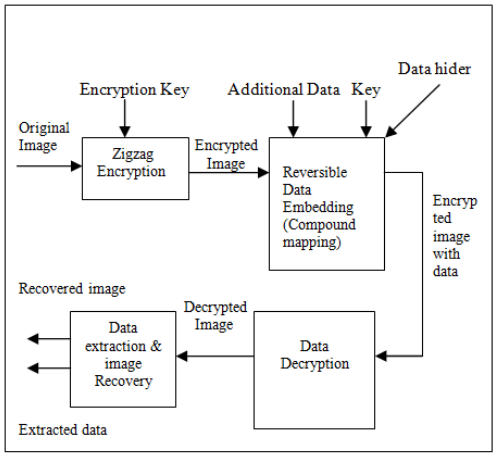

Fig.1.Block Of Proposed Method

A sketch of the proposed scheme is given in Fig. 1. A content owner encrypts the original uncompressed video using an encryption key to produce an encrypted video, and then a data-hider embeds additional data into the encrypted video using a data-hiding key though he does not know the original content. With an encrypted video containing additional data,

A receiver may firstly decrypt it using the encryption key, and the decrypted version is similar to the original video. According to he data-hiding key, he can further extract the embedded data and recover the original video from the decrypted version. The detailed procedures are as follows.

A. *Introduction to Video Encryption*

Video encryption or video scrambling is a powerful technique for the preventing unwanted interception and viewing of transmitted video, for example from a law enforcement video surveillance being relayed back to a central viewing centre.

Video encryption is the easy part. It is the unscrambling that's hard. There are several techniques of video encryption. However, the human eye is very good at spotting distortions in pictures due to poor video decoding or poor choice of video encryption hardware. So, it is important to choose the right video encryption else you video transmissions may be unsecure or the decoding video unviewable.

B. Need from a good video encryption method

- Everyone has a unique encryption key or code.
- The video encryption system should not try and decode non encrypted video transmissions [2].
- The encrypted signal should be positively identified by the decoder. The decoder should recognise the encrypted signal and only attempt to decode when fully validated.
- On screen status display and identification.
- Automatic configuration to any video standard.

C. Data Embedding

With the encrypted video, although a data-hider does not know the original video content, he can embed additional message into the video by modifying a small proportion of encrypted video. Firstly, the data-hider segments the encrypted video in to number of frames then each frame convert into a number of non-overlapping blocks sized by BxB. Each block is having the one additional bit For each block, pseudo-randomly divide the B^2 pixels into two sets B0 and B1 according to a data-hiding key. Here, the probability that a pixel belongs to B0 or B1 is 1/2. If the additional bit to be embedded is 0, flip the 3 least significant bits (LSB) of each encrypted pixel in Bo,

$$B_{I,J,K} = \overline{B_{I,J,K}} \qquad (I,J) \text{ Belongs to } B0$$

2. HYPOTHESIS

This section explains some of the relevant video encryption algorithm that is applicable for the scope of the project. They include the use of standard encryption formats like AES, RC5 on video to other more specific video encryption techniques. All of the below mentioned algorithms uses 128-bit encryption key which make them to overcome any text-only attack and provide good security against other attacks like perceptual.

A. *Scrambling technique*

Here the compressed video image is encrypted in the transform domain; this type of encryption is used for video compression like motion JPEG, Advanced Video Codec (H.264). This is a joint compression and encryption technique. Here the DC and AC coefficients of the Discrete Cosine Transform of the image is alone scrambled [3]. Mainly adapted in surveillance systems. Offers good encryption of the foreground images but for the stationary background it offers little or no encryption good for surveillance systems but not secure enough for other application like medical and defence. It offers full codec compliance.

B. *Real time video encryption using AES*

It uses the standard text encryption technique for video encryption. Instead of using AES on the whole video, it selectively encrypts the I-frame or DC and AC coefficient or all of the three depending on the level of security [4]. Since the amount of data that is to be encrypted the time taken for encryption is also greatly reduced. Thus it can be implemented for real time processing and encryption for low data rate videos. It is also offers full codec compliance, which means that any ordinary video with the supported standard codec can play the encrypted video without crashing.

C. *Fast and secure real-time video encryption*

Secure real-time video encryption algorithm [5] uses RC5 which the author mentions that can be substituted by any oth-

er standard encryption methods. But here only the I-frames are encrypted, as the I-frame contain the highest amount of data and all other frames are dependent on this frame to reconstruct them. Thus based on the above statement the encryption is considered to be secure. This algorithm is faster than AES encryption of the entire data but it is still slower than scrambling algorithm already mentioned. This algorithm also offers full codec compliance, which offers higher stability to the plug-in if played in a web browser, and does not end up crashing the plug-in causing the loss of user session and valuable data, when the encrypted video is accessed in an accidental manner.

D. Zigzag partitioning and swapping

This algorithm operates on MPEG video. This algorithm divides the input bit stream of the compressed video into equal size blocks of size all blocks are of equal size except for the last block. Using N=4 will divide the input sequence into blocks of 16. Then the partitioned blocks are rearranged based on the zig-zag rule [6]. This zigzag rule rearranges the data based on the data itself of the input key. This rearranged data is then swapped in a pseudo-random manner within a block. This algorithm offers high speed of encryption but is not standard codec compliant and it encrypts the whole video stream which is redundant. The algorithm is explained in detail in later section.

3. RESEARCH METHOD PRACTISED

In the last review at the end of the literature survey it was decided to combine the methods of block scrambling and selective encryption of video frames to obtain both fast results and also to maintain high levels of security.

The block scrambling is to be performed based on the reference [3], this uses the scrambling method described in the reference [6]. It is a combination of these two techniques in which the whole data set to be encrypted is segmented into smaller blocks and these blocks are scrambled using the Zigzag algorithm rather than all the bits as described in the original Zigzag algorithm. A multilevel scrambling was chosen for encryption. In addition to the block scrambling only the video data that has the highest information like the I-frames and the motion vectors are scrambled. This reduces the data to be encrypted and thus increasing the speed of encryption.

As far as the scope of the paper is concerned only block scrambling using the modified Zigzag algorithm is implemented. Here only one level of scrambling that is only inter block scrambling is performed. This is done to all the frames of the images not confined only to the I-frames. The detailed explanation for the modified Zigzag algorithm is explained the following section.

4. ZIGZAG ALGORITHM

This section explains the zigzag method for scrambling in detail below.

This algorithm operates on MPEG video. This algorithm divides the input bit stream of the compressed video into equal size blocks of size 16x16. Then the partitioned blocks are rearranged based on the zigzag rule [6]. This zigzag rule rearranges the data based on the data itself of the input key. This rearranged data is then swapped in a pseudo-random manner within a block. This algorithm offers high speed of encryption but is not standard codec compliant and it encrypts the whole video stream which is redundant

A. Original Zigzag method

A line or course that proceeds by sharp turns in alternating directions is known as zigzag rule. By this rule first bit of selected bits will be placed in right position. Again second bit will place in left and third bit will be placed in right position and so on. Here is an example of zigzag:

Let's select two bit BIT = "1010010100101011"
BITZigzag= [Empty]

First bit of BIT = 1
It will be placed in the right position of BITZigzag.
So, BITZigzag= "1"
Second bit of BIT = 0
It will be placed in the left position of BITZigzag.
So, BITZigzag = "01"

Third bit of BIT=1
It will be placed in the right position of BITZigzag.
So, BITZigzag = "011"

Forth bit of BIT= 0
It will be placed in the left position of BITZigzag.
So, BITZigzag = "0011"

After placing all bits BIT will become
BITZigzag = "1000110011000111"

B. Modified Zigzag Algorithm

1) *Read the given video file using matlab video read function.*
2) *Get the user key from the user of length of 32 characters.*
3) *Convert the input key into ASCII value.*
4) *Convert the ASCII value into binary with a resolution of 8 bits.*
5) *Extract the frames from the given video file.*
6) *Play the video file.*
7) *For each frame do the following steps*
    a. *Resize the frame to a size of 256x256*
    b. *Segment the frames into blocks of 16x16*
    c. *For each of the 16x16 blocks do the following*

i. *Rearrange the blocks in the form of an array of cell 1x256*
ii. *When a '0' appears in the key place the block under consideration before the previously operated block*
iii. *When a '1' appears in the key place the block under consideration after the previously operated block*
iv. *Rearrange the blocks to form the 256x256 image and save as frame*

8) *Play the encrypted video file*
9) *Stop.*

- ***Intra block scrambling***

Even the though the scrambling is performed, it was performed only to the blocks. But the data in individual blocks are still recognizable and this can solve by further scrambling the data inside the individual blocks. This block size can be varied based on the need, thus the computational load can be kept at an optimal level.

- ***Redundancy in Key***

If the key during the decryption the change in the decryption is seen only after the 1st bit that changes from the original key in the text-only attack, if that change occurs at a very later stage then most of the data that were decrypted before that occurrence will be decrypted properly.

To avoid this, encryption can be performed twice rather than once. In the second time the key should be reversed so as to compensate for the occurrence of the redundancy. This will add to the computational burden thus adapting an efficient method is important.

5. COMPOUND MAPPING ALGORITHM

A. Data Embedding procedure in encrypted video

Keeping generic lossless visible watermarking approach as the base paper, it implemented watermarking for image, the procedure has been extended to Encrypted video with that extraction module has one for module for extracting the watermark. Attacks against watermarking are regarded as common image recovery problems. To make watermark more useful, must care about its robustness against a various kind of possible attacks. These include robustness against noise addition such as salt & peeper, Gaussian noise and speckle noise and compression attack such as scaling and aspect ratio changes, rotation, cropping, row and column removal, filtering, cryptographic and statistical attacks, as well as insertion of other watermarks. Different types of geometric and signal processing attacks were checked against watermarked video and the PSNR value is calculated to find the quality of the video

Q - Original video source

Q1- encrypted (Q)

$$Q2 = Q1(F^{-1}{}_{B}(F_{A}(P)))$$

Encrypted image (Q1)

R-224 G-132 B-106	R-224 G-132 B-68	R-229 G-32 B-123
R-220 G-156 B-120	R-230 G-134 B-70	R-212 G-234 B-67

Data to hide (B)

R-24 G132 B-100	R-223 G-132 B-68	R-229 G-32 B-12
R-20 G156 B-119	R-30 G-13 B-70	R-212 G-34 B-67

Consider the pixel values for colour image video source A=(220,156,120) P=(230,134,70) B=(30,13,70) then Q2=(40,35,120) where a=Original encrypted image B-Data to hide P-Nearest pixel value in encrypted image Q2-encrypted image with additional data

Data Extraction and video Recovery

The video decryption has been done using the encryption key each block has been decrypted using the encryption key when having an encrypted video containing additional data, a receiver firstly generates according to the encryption key, and calculates the exclusive-or of the received data and to decrypt the video. The data can be extracted from the encrypted video by the compound mapping function by using this function can recover the original source video and can extract the data without any loss

6. RESULTS AND ANALYSIS

The video taken for the test is 30 frame video sequences, wherein each frame is of the size 256x256. In this paper only the 16x16 blocks have been scrambled and the intra block scrambling has also been performed.

A. Security

In the above figures the original frame and the encrypted frame are shown. This video has been encrypted using a 256-bit long key which is user entered. Since the key is 256 bits long it has enough security to resist any text-only attach, which is performed by generating all possible combination of the key and tested against the encrypted video stream.

Security against the text-only attack is widely considered that the time taken to decrypt the video stream should be greater than the play time of the video sequence as such.

B. Speed

A high quality video has a frame rate of 30. Thus the encryption algorithm should be fast enough to encrypt data at a rate equal or faster than 30 fps in order to transmit data in real-

time manner. The Block scrambling method takes less the 30ms to encrypt the data. Thus the encryption process can be implemented in real-time as the encryption time is less than the video playback time.

C. Loss less recovery

The mapping function used for each framed of the video then this function based upon the pixels valued manipulation so as how we are embedded the data we can extract the data without any loss

D. Overhead

Since only the data that is contained in the frames are encrypted without any addition of extra information, the overhead added due to encryption is almost zero. Thus the video does not change in size after encryption. But this method requires a larger key for the decryption of the encrypted data.

	10Frames	20Frames	30Frames
Zigzag Algorithm	1.4664 (146 ms/frame)	2.8173 (140 ms/frame)	4.2474 (141 ms/frame)
(Proposed)Block Scrambling Algorithm	0.2796 (28 ms/frame)	0.4773 (23 ms/frame)	0.6729 (22 ms/frame)

Table1 : The average time for encryption

Method	Legitimate recovery	Data size
Y.Hu[11]	37-38	Unlimited
Proposed	Loss less	Unlimited

Table 2 : Comparing reversible data hiding techniques

The table 1 shows that average times for encryption of both the Zigzag algorithm and the modified zigzag method (Block scramble method). In the zigzag method the numbers of values that are scrambled are in the order of the product of the number of frames, size of each frames and also the resolution of each pixel. The output of the video stream both original and encrypted, one of the frames is shown below.

The table 2 shows the legitimate recovery of the video peak signal to noise ratio and size of the data to hide Figure 2 shows that the original video frame it is a MPEG video format .Figure 3 shows that the encrypted frame using zigzag algorithm Figure .4 shows that the encrypted frame using Block Scrambled. Figure .5 shows that the data's to hide in the encrypted video frame. Figure.6 shows that the recovery of original video without any loss. Figure.7 shows that the extracted data. By this proposed method Although someone with the knowledge of encryption key can obtain a decrypted video and detect the presence of hidden data if he does not know the data-hiding key, it is still impossible to extract the additional data and recover the original video then at the time of recovery phase it is very much efficient to recover the original data without any loss.

Fig 1. Orginal Video frame before encryption

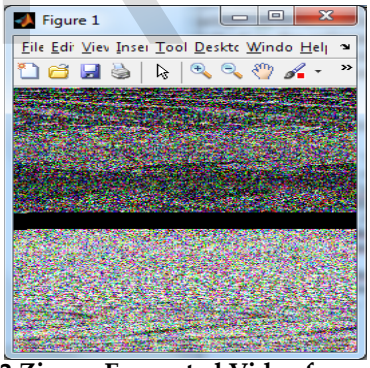

Fig.2 Zigzag Encrypted Video frame

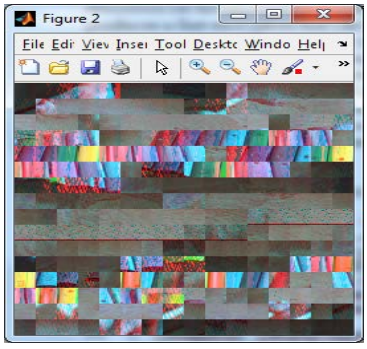

Fig 3. Block Scrambled Video Frame

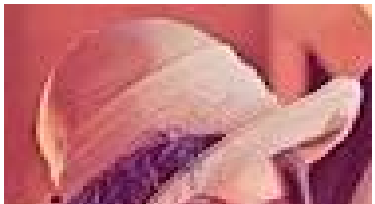

Fig 4. Data to Hide

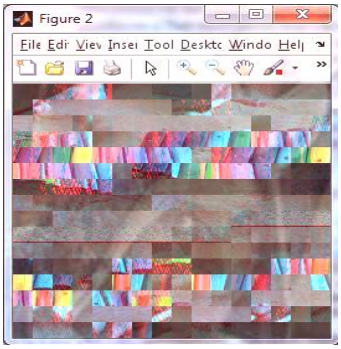

Fig 5. Encrypted Video with additional data

Fig 6. Recovered orginal video without loss

Fig 7. Extracted Data

While in the block scramble method the number of scramble is in the order of the product of the number of frames and the size of each frame. Thus the amount of computation is largely reduced and thus the computation time is highly improved, resulting in a reduction in time by 1/7 the time of the zigzag algorithm. The output of the Block Scramble method can be improved my reducing the size of the blocks. But strength of encryption equals that of the Zigzag algorithm as it uses a 256 bit key word.

5. CONCLUSION

In this work, a novel reversible data hiding scheme for encrypted video with a low computation complexity is proposed, which consists of video encryption, data embedding and video-recovery phases. The data of original video are entirely encrypted by a stream cipher. Although a data-hider does not know the original content, he can add additional data into the encrypted video by modifying a part of encrypted video data. With an encrypted video containing additional embedded data, a receiver may firstly decrypt it using the encryption key, and the decrypted version is similar to the original video content. According to the data-hiding key, the embedded data can be correctly extracted while the original video can be perfectly recovered. Although someone with the knowledge of encryption key can obtain a decrypted video and detect the presence of hidden data if he does not know the data-hiding key, it is still impossible to extract the additional data and recover the original video. the block based scrambling method is described for the encryption a video sequence. This method produces a fast encryption that can be implemented in real-time, with a time of 30 millisecond for encrypting a single frame. This encryption technique also does not add overhead due to encryption to the original video stream. And the additional data embedded if recovered loss less using the compound mapping function so this proposed work is very much efficient for security and privacy schemes

REFERENCES

[1] Fuwen Liu*, Hartmut Koenig, "*A survey of video encryption algorithms*", Elsevier publications, computers and security, June 2009

[2] Ujwala Potdar1, Prof. K.T.Talele, Dr.S.T.Gandhe, "*Perceptual Video Encryption for Multimedia Applications*", IEEE Second International Conference on Computer Engineering and Applications, 2010.

[3] FrédéricDufaux, TouradjEbrahimi,"*Scrambling for Privacy Protection in Video Surveillance Systems*",IEEE transactions on circuits and systems for video technology, *VOL. 18, NO. 8, 1168-1174*,August*2008*

[4] JayshriNehete, K. Bhagyalakshmi, M. B. Manjunath, ShashikantChaudhari, T. R. Ramamohan, "*A Real-time MPEG Video Encryption Algorithm using AES*",

[5] C. NarsimhaRaju, GanugulaUmadevi, KannanSrinathan and C. V. Jawahar, "*Fast and Secure Real-Time Video En-*

cryption", Sixth Indian Conference on Computer Vision, Graphics & Image Processing, January 2009

[6] A. F. M. SuaibAkhter, Saiful Islam, Md. JakirHossain, Rupam Deb, Dr. Md. BasirUddin, "*MPEG Encryption by Zigzag, Partitioning and Swapping*", IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.5, 116-120, May 2010

[7] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890– 896, Aug. 2003.

[8] Z. Ni,Y.-Q. Shi,N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354–362, 2006.

[9] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless gene LSB data embedding," *IEEE Trans. Image Process.*, vol. 14, no. 2, pp. 253–266, Feb. 2005.

[10] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Re versible image watermarking using interpolation tech nique," *IEEE Trans. Inf. ForensicsSecur.*, vol. 5, no. 1, pp. 187–193, 2010.

[11] Y. Hu, S. Kwong, and J. Huang, "An algorithm for re movable visible watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no.1, pp. 129–133, Jan. 2006.