An H.264 Video Encryption Algorithm Based On Entropy Coding

Cai Mian, Jia Jia, Yan Lei
School of Electronics Information & Control Engineering, Beijing University of Technology, Beijing 100022,P.R.China
jiajia_bjpu@emails.bjut.edu.cn

Abstract

H.264 video coding standard supports a broad range of application,and how to guarantee its security has already become an urgent problem. In this paper, we propose a video encryption algorithm for H.264, which combines stream cipher algorithm with entropy coding process. It obtains video content security by encrypting codeword index to scramble the position of the codeword. Experiments show that, the proposed algorithm is able to provide compromise between security and complexity, has little effect on compression performance. It's believed to be applied in the fields of video conference, digital rights management, multimedia storage, etc.

1. Introduction

H.264, known as the newest international video coding standard, represents the state of the art in video compression for its network-friendly feature and excellent compression performance.H.264 can support a wide range of applications, such as video conference, video storage system, video on demand, and so on. With its development, the H.264-based products will become more and more popular in video market. So there will be a great desire for H.264 security methods.

The early methods for securing video information mainly depends on authentication control mechanism, the video content itself isn't encrypted.It will easily offer penetrators opportunities for illegal copying, pirating, falsifying and juggling in video transmission. However, most of current video encryption methods are limited in the area of MPEG standard. So,it's essential to research new video content encryption schemes suitable for H.264 standard.

In this paper, firstly we review former works about video encryption in recent years and introduce the theory of H.264 entropy coding.And then we propose a video encryption algorithm based on H.264 entropy coding. Finally we present the experimental results and make the performance analysis of this algorithm.

2. Background

Different from ordinary text data, video data are often of large quantity, special coding structure, and demand real time transmission. These characteristics put forward the problems in video security, real-time processing, compression performance, compliance of data format, etc. So how to design an effective scheme in terms of these characteristics is an urgent problem.

Current video encryption algorithms can be classified into three categories according to the relationship between encryption process and compression process: complete encryption algorithm, selective encryption algorithm and encryption algorithm based on entropy coding [1]. Complete encryption algorithm is often of high security, but encrypting all the data will bring great computational complexity. Selective algorithm, especially DCT-based selective algorithm, has nice features in real-time processing and compliance of video format, but encrypting DCT coefficients will change the statistical performance of entropy, that means, it will have negative effect on video compression performance. With the development of video coding technology, encryption algorithm should be well combined with compression process so as to satisfy the expectations in real-time transmission, error resilience and format maintenance. And that, because of the outstanding security and compression performance, the encryption algorithm based on entropy coding will surely become the research direction of video security algorithms.

2.1. Former works

The video encryption methods based on entropy coding have already been researched in the past years. Jingtao Wen and Mike Severa [2] have presented a format-compliant configurable encryption framework. In this framework,a compressed MPEG bitstream can be divided into information-carrying and not information-carrying portions, and the information-carrying fields are what need to be encrypted.Bits from these fields are extracted to be concatenated in an appropriate way.The concatenation will be encrypted by a common cipher algorithm such as DES, and then the encrypted bits need to be put back into their original positions.To maintain compliance, they assign a fixed length index to each codeword, encrypt the index concatenation, and map the encrypted index concatenation back to codeword.

Chunping Wu and C.-C.Jay Kuo [3] have proposed a methodology of fast encryption by modifying entropy coder in video compression system, called multiple Huffman tables (MHT) scheme. Train 2^k different Huffman tables by using a technique called Huffamn tree mutation,then generate a random vector as the key stream to encrypt the Huffman tables. Dahua Xie and C.-C. Jay Kuo [4] have proposed an enhanced MHT scheme. They use a one-way hush function *h(·)* to emulate a key hopper so as to overcome the choosen-plaintext attrack weakness. This scheme is proved to be to safer than MHT scheme.

The idea of encrypting compressed bitstream has already been researched and realized in software. However,as mentioned above, most of these encryption algorithms are on the basis of MPEG.H.264 entropy coding has its own characteristics,which is different form MPEG. It's nature to design a video encryption algorithm adapted to H.264 entropy coding.

2.2. The theory of H.264 entropy coding

Most of the video coding standards,including H.264, are based on a hybrid of motion compensation and transform coding, and entropy coding refers to the encoding of parameters, coefficients, or other numeric values using binary codes. Different from other video standards, H.264 presents two entropy coding methods: context-adaptive variable length coding (CAVLC) and context-adaptive binary arithmetic coding(CABAC). CAVLC, offered by baseline profile, is easy to realize and of low computational complexity. CABAC,offered by main profile, gives the most coding gains but is computationally expensive. What's more, H.264 baseline profile can support a wide range of video applications, such as wireless communication, video conference, visual telephone,etc. H.264 main profile can support digital video broadcasting, but it is not the mainstream of video application currently. So in this paper,we only discuss CAVLC,but don't take CABAC into consideration due to its coding complexity.

The process of H.264 CAVLC is specified in the following ordered steps [5]:

(1)Encode total number of all coefficients and trailing ones (*TotalCoeffs* and *TrailingOnes*).

(2)Encode signs of trailing ones.

(3)Encode non-zero coefficient levels except trailing ones (*Levels*).

(4)Encode total number of zeros before last non-zero coefficient (*TotalZeros*).

(5)Encode runs of zero coefficients before each non-zero coefficient (*Runbefore*).

There are five different codeword tables according to the five steps. In the encryption scheme proposed, we will use these codeword tables for reference.

3. The proposed encryption algorithm

In this paper, we propose a video encryption algorithm adapted to H.264 entropy coding. In the process of CAVLC, we use stream cipher algorithm to encrypt the codeword index so as to get a new index, then look up codeword table to determine the new codeword according to the encrypted index. The new codewords will be as the output data stream. This encryption method can help us get the new codewords different from the originals by scrambling the positions of the codewords in entropy codeword table.

As described above, CAVLC goes through five steps, each step has its own codeword table.However, it's not suitable to embed the encryption into all the five steps. The process of encoding non-zero coefficient levels includes encoding *level_prefix* and *level_suffix*. This process is so complex that encrypting *Levels* will easily result in invalid codewords. So we won't do encrypting operation in this process. Additionally, the number of *TrailingOnes* is small, it occupys a small minority of code stream. Encrypting the signs of *TrailingOnes* makes no sense to improve security performance. Therefore, we embed encryption into the process of encoding *TotalCoeffs*,*4×4 block TotalZeros*, *chroma DC 2×2 block TotalZeros* and *Runbefore*.

In the following parts, the process of encrypting codewords indices and the mechanism of key generation will be expressed in detail and the procedures relevant will be presented.

3.1. Encrypting the codewords indices

As described above,the encryption process can be illustrated as Figure 1.

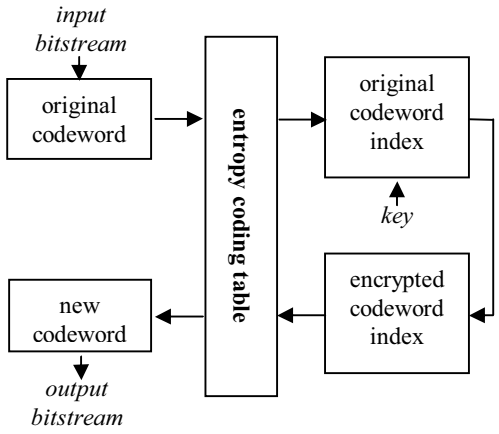

Figure 1. Codeword index encryption process

In this process, a problem should be taken into consideration, that is, the codewords in some fields of the codeword table are invalid. The encrypted index mapping back to the table may point to invalid codeword. So we need to judge whether the encrypted index is located in the invalid codeword fields. If this situation happens, we should do further operation.

Take *Runbefore* codeword table for example. In this table,if the position of codeword is under the condition: row≦7 & column≦15 or row≦6 & column≦row, the codeword is valid. Where "column" and "row" stand for the position index of the codeword. Assume a codeword P, with its position index (m,n) of the *Runbefore* table, it will be turned into C(m',n') after encryption. The process of judgement is shown below:

Input original codeword P(m,n)
m' = encrypt m
n' = encrypt n
If (n' >15) then
n' = n' mod 15
Else if (m' < =6) then
If (n' > m') then
n' = n' mod (m' + 1)
Else m' = m' mod 7
Else keep m' and n' as they are
Map the index (m',n') back to the Runbefore table
Output the new codeword C(m',n')

3.2. Key generation and distribution

In this paper, we use RC4 cipher algorithm to provide data confidentiality. RC4 is a variable key-size stream cipher with byte-oriented operations, it can be expected to run very quickly, and considerable secure. These advantages make it a classical public-key cipher algorithm. Through this algorithm we use the public keys to generate master keys, which are distributed to compression process. The pseudo-random byte based on non-linear data exchange will be XOR-ed with the codeword index. This process can provide adequate security to resist known-plaintext attack.

4. Performance analysis

4.1. Security analysis

As mentioned above:RC4 uses a variable length key form 1 to 256 bytes to initialize a 256-byte state table, and then generates a pseudo-random stream to be XOR-ed with the codewords indices. The RC4 key is often 128 bits at least, that means its key space is 2^{128}. We use the key stream to encrypt codeword tables four times. It's obviously to see that it will generate a new key stream in each encrypting operation. So for one codeword index, the key space is 2^{512} at least. Moreover,video data are of large volume,so it's nearly impossible to decrypt ciphertext at decoding terminal.

Take standard video sequence "Claire" for test. As shown in Figure 2,the first image is the 20th frame of original video sequence,others are the 20th frames of encrypted video sequences,but the encryption methods are different.The second image is based on selective encryption(only DC coefficients are encrypted) and the third one is based on entropy coding encryption. From the view of encrypted video quality, the third image is more illegible than the second one. Therefore, the security of this algorithm is satisfying.

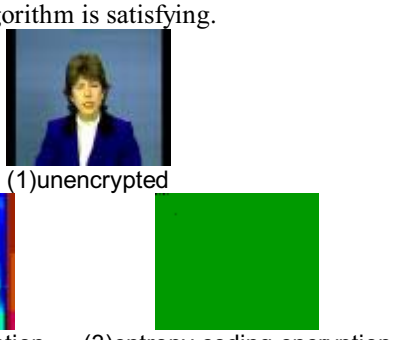

Figure 2. 20th unencrypted and encrypted frames of Claire

4.2. Computational complexity

The computational complexity of this algorithm mainly depends on encryption process and codeword

lookup process. From encrypting codeword index to mapping the index back to codeword table, the time cost is almost equal to the time looking up CAVLC codeword tables costs, so carrying out the encryption won't bring too much complexity. In addition,RC4 is an efficient algorithm, it takes a short time to generate and distribute the key, and it only use simple XOR operation. So, the computational complexity is low enough to support real-time processing.

4.3. Compression Performance

In the scheme we proposed, the cipher algorithm is embedded into CAVLC. What we encrypt are not the codewords, but the correlative codewords indices, and the codewords sent to the NAL unit are still from the codeword table, so it has very little effect on compression ratio. We use compression variation ratio Δr to test the compression performance of the video encryption algorithm. Δr can be defined as follow:

$$\Delta r = \frac{|r_1 - r_2|}{r_1} \times 100\%$$

r_1 represents video data quantity before encrypting, and r_2 represents vedio data quantity after encrypting. The comparing is shown in Table 1.(Video Format: qcif, Numuber of coded frames:100)

Table 1. Compression performance test

Video sequence	r_1 (bits)	r_2 (bits)	Δr (%)
Claire	106304	106376	0.068
Akiyo	100112	100400	0.288
Mother&Daughter	155792	156432	0.411

4.4. Rate distortion performance

Rate distortion (RD) curve provides a criterion of encoding quality. It's important to reconstruct the images after decryption and decoding.In this figure,the x-axis represents bit rate(Unit:Kbit/s), and the y-axis represents coding quality, which uses the Y-PSNR (Peak Signal-to-Noise Ratio of video luma value) to indicate(Unit:dB). At the same bit-rate, the higher curve point means the better reconstruction quality.

Take "Claire" for test (Video Format:qcif,Frequency for encoded bitstream:30Hz,QP=28).The RD curves of three encryption methods are shown in Figure 3.The curve '◇' is based on complete encryption algorithm, the curve '△' is based on our proposed algorithm,and the curve '✳' is based on selective algorithm (DC coefficients encrypted).As can be seen in the figure,the entropy coding algorithm shows the best coding quality. That is because, in this algorithm, the vedio format isn't changed, and adoption of stream cipher can well maintain the precision of bit stream control.

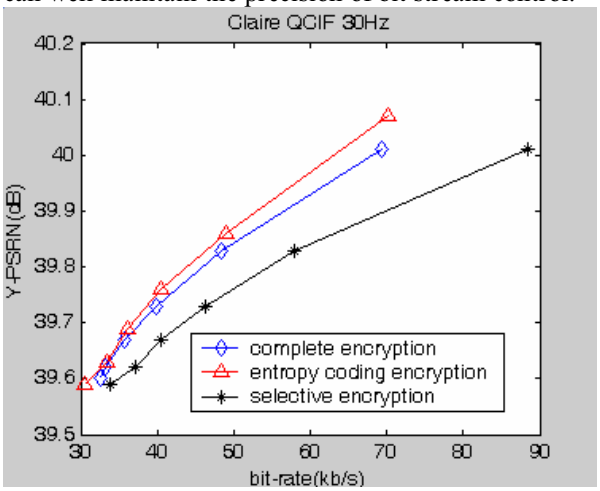

Figure 3. RD Curves of three encryption algorithms

5. Conclusion and future work

In this paper, we propose a video encryption algorithm based on H.264 entropy coding. This algorithm can make a trade-off between security and complexity, not compromise compression performance. It's suitable for the security of video conference, digital rights management, multimedia storage,etc.

This method is extended to be improved by optimizing cipher algorithm and changing encrypted position. In addition, H.264 has proposed two entropy coding methods. With the research and application of H.264 in digital video broadcasting, how to encrypt bitsteam in CABAC supported by main profile, should be taken into consideration. These are our future work.

References

[1]Lian Shi-guo, Sun Jin-sheng. Video Encryption and Its Development.*Information and Control*,Vol.33,No.5,2004,10.

[2]Jiangtao Wen,Severa, M., Wenjun Zeng. A Format-Compliant Configurable Encryption Framework for Access Control of Multimedia.*Multimedia Signal Processing*, 2001 IEEE Fourth Workshop on 3-5 Oct. 2001 Page(s):435 – 440

[3]Chung-Ping Wu and C.-C. Jay Kuo. Efficient Multimedia Encryption via Entropy Codec Design.*SPIE International Symposium on Electronic Imaging*, 2001, (San Jose, Ca, USA), Jan.2001, Proceedings of SPIE Vol.4314.

[4]Dahua Xie and C.-C. Jay Kuo.Enhanced multiple huffamn table(MHT) encryption sheme using key hopping. *Circuits and Systems*,2005,ISCAS 2005. IEEE International Symposium on 23-26 May 2005 Page(s): 5533 – 5536,Vol.6.

[5]T.Wieg. Draft ITU-T Recommendation H.264 and Draft ISO/IEC 14496-10 AVC.Joint Video Team of ISO/IEC JTC1/SC29/WG11&ITU-T SG16/Q.6 Doc.JVT-G050,2003.