http://dx.doi.org/10.7236/JIIBC.2013.13.3.119
JIIBC 2013-3-16

적응 산술 부호화를 이용한 고화질 영상 암호화 전략

Enhanced Image Encryption Scheme using Context Adaptive Variable Length Coding

심갑용, 이말레[*]

Gab-yong Shim, Malrey Lee

요 약 실시간 부호 매김과 비디오데이터 부호화 변환을 위한 비디오 부호화 방법은 비디오 압축 과정을 가진 통합 부호화 기법을 사용한다. 본 연구에서는 접근이 허용되지 않은 비디오 데이터를 가진 사람이 접근을 하지 못하도록 비디오 데이터 부호화 기법에 대해서 연구 하였다. H.264 엔트로피 코딩에 대한 연구와 H.264 CAVLC 부호화 방법을 이용하여 진보된 비디오 부호화 알고리즘에 대한 연구이다. 특히 보다 더 강한 보안 부호화 프레임을 만들기 위하여 혼합 알고리즘을 제안하였다. 제안한 방법은 비디오 데이터 암호화 기능과 압축률이 호전됨을 실험을 통하여 알 수 있었다.

Abstract o achieve real-time encryption and video data transcoding, current video encryption methods usually integrate encryption algorithm with video compression course. This paper is devoted to discussing the video encryption technology, by encrypting to avoid unauthorized person getting video data. This paper studied the H.264 entropy coding and proposed of CAVLC video encryption scheme which is combined with the process of entropy coding of H.264 CAVLC encryption scheme. Three encryption levels are proposed. In addition, a scrambling method is also proposed which makes the encrypted frames more robust in anti crack. This method showed more robust video data encryption function and compressive rate .

Key word : Digital Video Encryption, Surveillance, Security, Encryption

Ⅰ. Introduction

With range enlarging of digital video application, the demand of video data safety is getting more and more. Likewise, video data safety is also needed in human detection system.

Video data encryption algorithms roughly fall into four kinds: full encryption, selective encryption, DCT permutation encryption and entropy encryption. Full encryption algorithm treats video data as common bit stream. This algorithm provides the highest Security, but it is time-consuming for mass data, so it cannot meet requirement of a real-time system such as human detection system. Selective encryption is also called partial encryption. In the process of video encoding, sectional data is encrypted. Compared with full

[*]정회원, 전북대학교 영상정보신기술연구센터 (교신저자)
접수일자 2013년 5월 16일, 수정완료 2013년 6월 12일
게재확정일자 2013년 6월 14일

Received: 16 May 2013 / Revised: 12 June 2013 /
Accepted: 14 June 2013
[*]Corresponding Author: mrlee@chonbuk.ac.kr
Center for Advanced Image and Information Technology, Chonbuk National University, Korea

encryption, selective encryption processes less data, and provides better real-time, but the format information of video data is changed so that the video data no longer has operability. In DCT permutation encryption, DCT coefficients make up a binary code stream and XOR with secret key.

It provides fast encryption speed but low security. In the process of entropy encoding, both coding and encryption are achieved.

Video data is huge, and importance of every part is different. For meet request of security and real-time, it is necessary to design algorithm on the basis of video data features. H.264 lays emphasis on high performance data compression and high reliability of data transmission, while error code-handling capacity is enhanced. Therefore high security and real-time is achieved.

II. H.264 Encoding Standard

H.264/MPEG-4 AVC is a block-oriented motion-compensation-based codec standard developed by the ITU-T Video Coding Experts Group (VCEG) together with the ISO/IEC Moving Picture Experts Group (MPEG).The intent of the H.264/AVC project was to create a standard capable of providing good video quality at substantially lower bit rates than previous standards, without increasing the complexity of design so much that it would be impractical or excessively expensive to implement. There are 2 parts in H.264 encoding system, one is Video Coding Layer (VCL), and the other is Network Abstraction Layer (NAL)[1].

VCL is used to describe and define Video Content. It includes Motion Compensation, transform and quantization and entropy coding. NAL is in charge of data encapsulation, frame format, logical channel control, and stop bit definition. A typical NAL packet consists of packet header information, segment structure information and VCL data load information. NAL gets encoded video data from access layer port, and packs VCL data under requirement of external network. NAL supports Circuit Switching (CS) and Internetwork Packet Exchange (IPX). Data header and data message can be mapped to most network protocols accurately. Encoding and channel are split by NAL, which enhanced flexibility of complex channels. By the layered structural design, compression, encapsulation and priority control of the video data are strengthened.

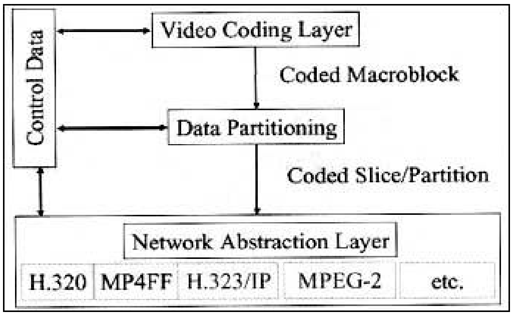


그림 1. H.264 계층구조
Fig. 1. H.264 Layered Structure

H.264 standard divided into three profiles: baseline profile, main profile and extended profile.

Baseline profile is mainly used in Interactive Application Communication, its technical features are: only operating I slice and P slice; using Intra-Prediction and interframe prediction to build pictures; using loop filter to reduce blocking effects; Context-Adaptive Variable-Length Coding (CAVLC) is used in entropy coding part; Slice is independent of each other and can be transmitted to decoder in random order; redundant slices is transmitted to recover in data error.

III. Encryption Scheme Based on CAVLC

In this chapter, an encryption based on CAVLC is proposed, and Figure 2 shows the flow of the scheme.

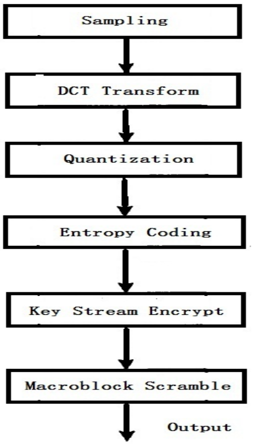


그림 2. 부호결합흐름도
Fig. 2. Encryption Combining with Coding Flow Chart

1. CAVLC Encoding Process

1.1 Process Description

1) Number of nonzero coefficient

Most of the coefficients turned to 0 by residual transforms, only a few nonzero coefficients left. Then residual coefficients are scanned in Zigzag, nonzero coefficients are on the left, and 0 are on the right. The last several nonzero coefficients are often ±1. ±1 are which continuously emerge from right to left in order (there could be several 0 but no nonzero coefficients). Number of ±1 never exceeds 3; if not extra ±1 will be treated as common nonzero coefficients. If last nonzero coefficient is not ±1, the number of ±1 is 0. The number of nonzero coefficients and ±1 become to a syntax element coeff_token.

2) Sign of ±1

Trailing ±1 are coded in 1 bit, +1 is coded into 0, −1 is coded into 1 from right to left.

3) Coding other nonzero coefficients

Except trailing ±1, other nonzero coefficients are coded from right to left. The code words consist of prefix and suffix. Suffix length means absolute value of the coded coefficient. If number of nonzero coefficient is more than 10 and number of trailing ±1 is less than 3, then suffix _length will be initialized into 1. Otherwise, suffix _length will be initialized into 0.

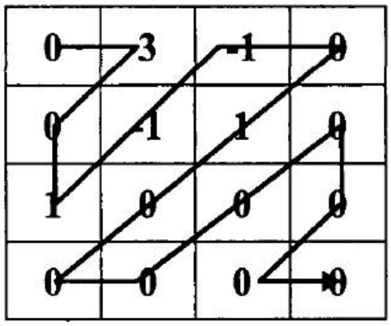
4) Coding number of 0 on the left of last nonzero coefficient Relevant code word is in code table of H.264 standard [13].

5) Coding every number of 0 before nonzero coefficients

Zero_left means number of 0 on the left of nonzero coefficient. Run_before means number of 0 between the nonzero coefficients and the next.

1.2. Coding Sample

From the above explanation, a specific example is showed [6]



0	3	-1	0
0	-1	1	0
1	0	0	0
0	0	0	0

그림 3. 4*4잔여계수 행렬
Fig. 3. 4*4 Residual Coefficient Matrix

In figure 3, by Zigzag scanning, the sequence is: 0,3,0,1,−1,−1,0,1,0,0,0,0,0,0,0,0

By the coding step:

(1) Number of nonzero coefficients is 5, number of trailing ±1 is 3(1,−1,−1). In the H.264 standard code table, corresponding code word is "0000100".

(2) The three trailing ±1 are coded from right to left. The coded bit is "011"

(3) Two nonzero coefficients are 1 and 3, suffix_ length is initialized into 0. From right to left, 1 is coded first, and calculates prefix to get 0

without suffix, then prefix is used to get code word “1” in code table, so 1 is coded into “1” and suffix_length is change to 1. 3 is coded to calculate prefix 2 and suffix 0, then prefix is used to get code word “001” in code table, suffix joints 001 to get the coded bit “0010”

(4) The last nonzero coefficient is “1”, there are three 0 on its left, the code word is “111” in the code table

(5) On the left of the first nonzero coefficient, there are three 0, there is one 0 between it and the next nonzero coefficient, so code word is “10” in the code table. Similarly, other code words of nonzero coefficients are “1”, “1” and “01”. The last nonzero coefficient-3 need not to be coded.

표 1. 코드화 단계

Table 1. Code Step

Step	Code element	Code word
(1)	(5,3)	0000100
(2)	1	0
	-1	1
	-1	1
(3)	1 : prefix=0, suffix_length=0	1
	3 : prefix=2, suffix_length=1, suffix=0	0010
(4)	Three 0 are on the left of last nonzero coefficient	111
(5)	1 : zero_left=3, run_before=1 (3,1)	10
	-1 : zero_left=2, run_before=0 (2,0)	1
	-1 : zero_left=2, run_before=0 (2,0)	1
	1 : zero_left=2, run_before=1 (2,1)	01
	3 : zero_left=1, run_before=1(1,1)	no code

So this 4*4 block coded bit stream is: 000010001110010111101101.

2. CAVLC Encryption Scheme

2.1 Scheme Description

By analyzing the process of CAVLC entropy coding, it is known that the number of trailing ±1 and nonzero coefficient can be abstracted. So in step (2) there is no need to encrypt syntactic element, otherwise the coefficient of the block cannot be decoded. Similarly, in step (4), the number of 0, which are on the left of last nonzero coefficient, cannot be encrypted. Consequently, encryption objects can only be sign of trailing ±1, nonzero coefficients and number of 0 before nonzero coefficient. Meanwhile, the coded motion vector will be encrypted on order to get higher security. According to this route, a video encryption scheme is presented.

Level 1 : encryption of trailing ±1.

There are three trailing ±1 in a 4*4 block at most, and 1 bit means a sign of trailing ±1. So the signs can be encrypted by a certain algorithm:

$$y = SEA(x)$$

x means the sign, SEA means encryption function, y means cipher text. Trailing ±1, which is high-frequency component of coefficient transformation, is on right of the Zigzag scanning sequence. So this encryption level can only scramble the video.

Level 2 : Besides level 1, encrypt suffix and run_before in step (3).

The last bit of suffix relates to plus-minus property of the nonzero coefficient. If the bit is 0, the nonzero coefficient is plus; if the bit is 1, the nonzero coefficient is minus. By encrypting suffix, plus-minus property of the nonzero coefficient is changed; meanwhile its absolute value is also changed. The encryption method of suffix and run_before is:

$$y = Encrypt(x), x = suffix \text{ , run\_before}$$

Encrypt is encryption function, y is cipher text. The bit of secret key is suffix_length. Encryption of run_before will change the coefficient position in sequence, thus inverse transform generate more deviation.

Level 3 : besides level 2, encrypt motion vector difference (MVD)

Exp-Golomb Coding is used for coding MVD. Prefix is M zero and 1, suffix is M bit INFO:

$$[Mzero]1[INFO]$$
$$Y = Encrypt(INFO)$$

Y is cipher text. Because H.264 used intraframe and interframe prediction technology, in case motion prediction of one block makes mistakes, then it causes avalanche effect, and there will be more confusion in the subsequence images. So MVD encryption makes further improvement of security.

Encryption functions can be neatly configured to achieve faster encryption speed and higher security.

2.2 Encryption Sample

This section gives encryption operation on level 2 for the block in 3.1.2. Stream cipher is used for encryption function. In sign encryption SEA, it combines 1 bit secret key and trailing ±1 by using XOR operator, so is the suffix. Run_before minus 1 bit secret key, if run_before is changed then zero_left need to be update.

Set secret key sequence as: 0, 0, 1, 1, 1, 0, 0, 1…

The first three bits are used to encrypt signs of trailing ±1 (011), the encrypted code stream is 010; and then one bit is used to encrypt the second suffix and gets 0011; the last four bits are sued to encrypt run_before. The encrypted bit stream is: “0000100010100111111111101”

표 2. 레벨2 부호화 예
Table 2. Sample of Leve2 Encryption

Step	Original code	Secret key	Encrypted code
(1)	0000100	No encrypt	0000100
(2)	0	0	0(0⊕0)
	1	0	1(1⊕0)
	1	1	0(1⊕1)
(3)	1	No encrypt	1
	0010	1	0011(1⊕suffix)
(4)	111	No encrypt	111
(5)	10(3,1)	1	11(3,0)
	1(2,0)	0	11(3,0)
	1(2,0)	0	10(3,1)
	01 (2,1)	1	1(2,0)
	No code	No encrypt	

IV. Simulation Analysis and Experiment

1. Experimental Result

From Figure 4 to Figure 7 are the frames of foreman_qcif in three levels encryption; The frame encrypted in level 1 can be recognized; in l level 2, the frame is difficult to distinguish; in level 3 the frame is beyond recognition. So are the scrambled frames.

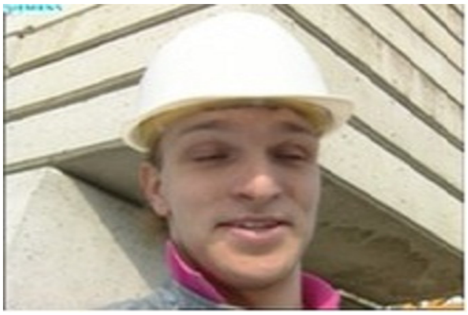


그림 4. 표준 프레임
Fig. 4. Foreman_qcif Standard Frame

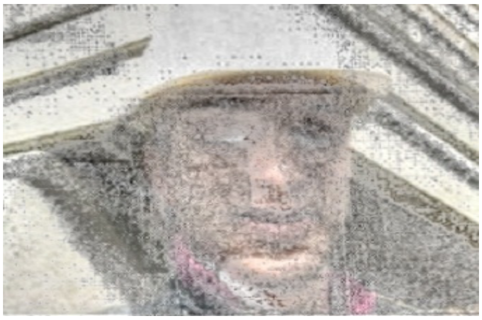


그림 5. 레벨 1 부호화
Fig. 5. Foreman_qcif Level 1 Encryption

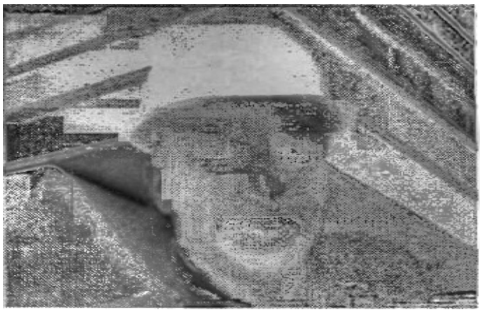


그림 6. 레벨2 부호화
Fig. 6. Foreman_qcif Level 2 Encryption

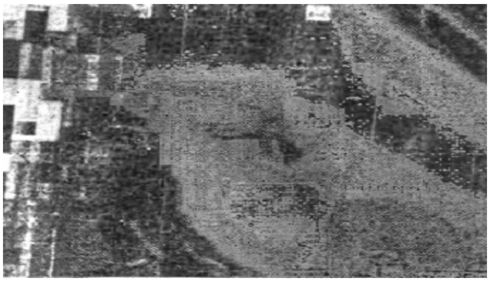


그림 7. 레벨 3 부호화
Fig. 7. Foreman_qcif Level 3 Encryption

2. Signal-to-Noise

The SNR of the encrypted frame luminance component is less than 15. Commonly, when SNR is less than 23 people feel obvious distortion. In table 3 SNR of the frames is showed.

표 3. 레벨 2 부호화 예
Table 3. Sample of Leve2 Encryption

	SNR(Y)	SNR(U)	SNR(V)
Foreman_qcif	10.75	19.60	29.13
Carphone_qcif	9.62	21.57	28.04

3. Project analysis

Video width is W; video height is H. In level 1 encryption, there are at lost three trailing ±1 in a 4*4 block. Maximal time of sign encryption is

((W*H/(4*4)+1/2*W*H/(4*4))*3;
in level 2 suffix encryption time is
((W*H/(4*4)+1/2*W*H/(4*4))*13;
in level 3 the encryption time is
((W*H/(4*4)+1/2*W*H/(4*4))*17, secret key is
((W*H/(4*4)+1/2*W*H/(4*4))*73 bit.

In QCIF (176*144), encryption time is 40392 and length of secret key is 173448 bit which has a low complexity of encryption operation.

4. Security Analysis

Security depends on stream cipher which can resist Know-Plaintext Attack. On resisting of exhaustive attack, P and B frames used most of encryption operation, so it is hard to crack P and B frames. It has to crack

((W*H/ (4*4) +1/2*W*H/ (4*4)) 4 secret keys to crack I frame. And in scrambling the macro block, a qcif frame has 11*9 macro blocks, it needs
((W*H/ (4*4) +1/2*W*H/ (4*4)) 4*99! secret keys to crack I frame.

5. Coding Compression Analysis

In level 1, the encryption only changed the sign of trailing ±1 which is the code stream in entropy coding. So the code length is not changed. In level 2, value of suffix is changed while the bit length of suffix is not changed so that code length is not changed. In level 3, same bit length secret key encrypted the coded suffix, so code length is not changed. There is no influence of compression efficiency.

6. Maneuverability Analysis

The encrypted code stream still meets the H.264 standards, so the maneuverability is persevered.

V. Conclusion and Future Work

In[7], it combined encryption with compression and included both CAVLC and QTC (quantized transform coefficients).[8] used permutation code and DES encryption algorithm.[9] presented a chaotic VEA which is not just in view of H.264.[10] presented method of SCAN-based permutation encryption algorithm. [11] proposed a transparent scrambling in DCT domain.

In recent years GPU and CPU Performance increase exponentially, multistage encryption can be achieved in real time video system.

In the paper, key point of CAVLC video encryption is that video data is encrypted in the process of entropy coding. After that, by macro block scrambling the scheme has a very high performance in security and while the system can meet the complexity of encryption. Different encryption levels can be used in many applications.

In future, the main emphasis is that on the basis of video data features, the combining of entropy coding and encryption. In aspect of key stream synchronization and key distribution, they remain to be further studied.

Also video encryption algorithm based on region of interest remains to be studied. Such as human face encryption or moving things encryption which is very useful in Video surveillance.

In addition, the scheme is based on H.264 and used entropy coding of

H.264. However, as the standards of video coding continues to develop, there is possibility to achieve the objective which not only encodes video but also encrypts video.

References

[1] Thomas Stockhammer, Miska M. Hannuksela, and Thomas Wiegand, "H.264/AVC in Wireless Environments," IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, Vol. 13, pp. 657-673, July. 2003.

[2] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol., Vol. 13, pp. 560 - 576, July 2003.

[3] M. Cai, Y. Li, F. Tang and L. Yan, "Research on Video Encryption Scheme of H.264 Multi-level Security Network," Communications Technology, Vol. 43, pp. 75-77, Jan. 2010.

[4] S. G. Lian, J. S. Sun and Z. Q. Wang, "A secure image or video transmission scheme based on image 1 library," Control and Decision, Vol. 19, pp. 827-830, July. 2004.

[5] C. Chen and S. Y. Yu, "Analysis and Improvement of Entropy Coding in H.26L," Communications Technology, Vol. 11, pp. 1-6, Nov. 2002.

[6] L. Tang, "Methods for encrypting and decrypting MPEG video data efficiently," Proceedings of the Fourth ACM International Multimedia Conference, Boston, MA, pp. 219-230, 1996.

[7] X. Y. Bao, J. G. Jiang and Y. Li, "A New Encryption Scheme for H. 264 Real-Time Video Transmission," ACTA ELECTRONICA SINCA, Vol. 34, pp. 2099-2102, Nov. 2006.

[8] Q. Zhang, J. M. Wu and H. X. Zhao, "Efficiency Video Encryption Scheme Based on H.264 Coding Standard and Permutation Code Algorithm," World Congress on Computer Science and Information Engineering, pp. 674-678, 2009.

[9] F. Chiaraluce, L. Ciccarelli, E. Gambi, P. Pierleoni, and M. Reginelli, "A new chaotic algorithm for video encryption," IEEE Trans. Consumer Electron., Vol. 48, pp. 838 - 844, Nov. 2002.

[10] S. S. Maniccam and N. G. Bourbakis, "Image and video encryption using SCAN patterns," Pattern Recognition., Vol. 37, pp. 725-737, Apr. 2004.

[11] C. Wang, H-B. Yu, and M. Zheng, "A DCT-based MPEG-2 transparent scrambling algorithm," IEEE Trans. Consumer Electron., Vol. 49, pp. 1208-1213, Nov. 2003.

[12] S. Wenger, "H.264/AVC over IP," IEEE Trans. Circuits Syst. Video Technol., Vol. 13, pp. 645 - 656, Jul. 2003.

[13] G. Bjøntegaard and Karl Lillevold, "Context-adaptive VLC (CVLC) coding of coefficients", document JVT-C028, JVT of ISO/IEC MPEG & ITU-T VCEG, 3rd Meeting, Fairfax, Virginia, USA, 6-10 May, 2002.

저자 소개

Gab-yong Shim(정회원)

Gab-yong Shim received a Ph.D. from the ChonBuk National University. He has been a associate professor at the ChonBuk National University in Korea. He had many experience in the filed and has many publications in various areas of Accounting and Computer Science, concentrating on Artificial Intelligence, Decision making system, Accounting and so on.

Malrey Lee(정회원)

Malrey Lee received a Ph.D. in Computer Science from the University of Chung-Ang. She has been a Professor at the ChonBuk National University in Korea. She has over seventy publications in various areas of Computer Science, concentrating on Artificial Intelligence, Robotics, Medical Healthcare, Software Engineering, Multimedia, Game, and so on.