Data Hiding in encrypted H.264/AVC Video Streams

Nikita Pawar
Department Of CSE
Saraswati College Of Engineering
Kharghar,
Navi Mumbai

Nayana Patil
Department Of CSE
Saraswati College Of Engineering
Kharghar,
Navi Mumbai

Babasaheb Suryavanshi
Department Of CSE
Saraswati College Of Engineering
Kharghar, Navi Mumbai

Rina Bora
(Assistant Professor)
Department Of CSE
Saraswati College Of Engineering
Kharghar,Navi Mumbai

ABSTRACT

Digital video sometimes needs to be stored and processed in an encrypted format to maintain security and privacy. For the purpose of content notation and/or tampering detection, it is necessary to perform data hiding in these encrypted videos. In this way, data hiding in encrypted domain without decryption preserves the confidentiality of the content. In addition, it is more efficient without decryption followed by data hiding and re-encryption. In this paper, a novel scheme of data hiding directly in the encrypted version of H.264/AVC video stream is proposed, which includes the following three parts, i.e., H.264/AVC video encryption, data embedding, and data extraction. By analyzing the property of H.264/AVC codec, the codewords of intraprediction modes, the codewords of motion vector differences, and the codewords of residual coefficients are encrypted with stream ciphers. Then, a data hider may embed additional data in the encrypted domain by using codeword substitution technique, without knowing the original video content. In order to adapt to different application scenarios, data extraction can be done either in the encrypted domain or in the decrypted domain. Furthermore, video file size is strictly preserved even after encryption and data embedding.

Keywords

Data Hiding, Encrypted domain, H.264/AVC.

1. INTRODUCTION

Cloud services may attract more attacks and are vulnerable to untrustworthy system administrators, it is desired that the video content is accessible in encrypted form.

The capability of performing data hiding directly in encrypted H.264/AVC video streams would avoid the leakage of video content, which can help address the security and privacy concerns with cloud computing.

With the increasing demands of providing video data security and privacy protection, data hiding in encrypted H.264/AVC videos will undoubtedly become popular in the near future. Obviously, due to the constraint of the underlying encryption, it is very difficult and sometimes impossible to transplant the existing data hiding algorithms to the encrypted domain.

An algorithm is used to embed additional data in encrypted H.264/AVC bit stream, which consists of video encryption, data embedding and data extraction phases. The data-hider can embed additional data into the encrypted bit stream using codeword substituting, even though we does not know the original video content. Since data hiding is completed entirely in the encrypted domain, here we can preserve the confidentiality of the content completely.

The first challenge is to determine where and how the bitstream can be modified so that the encrypted bitstream with hidden data is still a compliant compressed bitstream. The second challenge is to insure that decrypted videos containing hidden data can still appear to be of high visual fidelity. The third challenge is to maintain the file size after encryption and data hiding, which requires that the impact on compression gain is minimal. The fourth challenge is that the hidden data can be extracted either from the encrypted video stream or from the decrypted video stream, which is much more applicable in practical applications. Based on the analysis given above, we propose a novel scheme to embed secret data directly in compressed and then encrypted H.264/AVC bitstream.Firstly, by analyzing the property of H.264/AVC codec, the codewords of IPMs,the codewords of MVDs, and the codewords of residual coefficients are encrypted with a stream cipher. The encryption algorithm is combined with the Exp-Golomb entropy coding and Context-adaptive variable-length coding (CAVLC), which keeps the codeword length unchanged.

Then, data hiding in the encrypted domain is performed based on a novel codeword substituting scheme. In contrast to the existing technologies discussed above, the proposed scheme can achieve excellent performance in the following three different prospects:

- The data hiding is performed directly in encrypted H.264/AVC video bitstream.

- The scheme can ensure both the format compliance and the strict file size preservation.

- The scheme can be applied to two different application scenarios by extracting the hidden data either from the encrypted video stream or from the decrypted video stream.

In Section III, we describe the proposed scheme, which includes three parts, i.e., H.264/AVC video encryption, data embedding and data extraction. Finally in Section IV, conclusion is drawn.

2. RELATED WORK

W. Hong, T. S. Chen, and H. Y. Wu [6] have proposed that most of the work on reversible data hiding focuses on the data embedding/extracting on the plain spatial domain. The five MSBs of each pixel of the decrypted image will be identical to those of the cover image.According to the data-hiding key, it is easy for the data hider to reversibly embed data in the encrypted image.Thus the data hider can benefit from the extra space Emptied out in previous stage to make data hiding process effortless.Su-Wan Park*, Sang-Uk Shin*[11] describes the efficient selective encryption scheme for H.264/AVC. The proposed scheme encrypts with three domains in hierarchical layers using different keys: The intra prediction modes, the motion vector difference values and the sign bits of the texture data. The features of the proposed scheme are as follow. The proposed scheme offers

1)The computational efficiency by encrypting domains selectively according to each layer type, 2)The time efficiency through the light-weight encryption, 3)The security through the use of mutually different keys, and 4)The format compliance by utilizing the H.264/AVC structure and method.

Tamer Shanableh[23] proposes two data hiding approaches using compressed MPEG video. The first approach hides message bits by modulating the quantization scale of a constant bitrate video. A payload of one message bit per macroblock is achieved. A second order multivariate regression is used to find an association between macroblock-level feature variables and the values of a hidden message bit. The regression model is then used by the decoder to predict the values of the hidden message bits with very high prediction accuracy. The second approach uses the flexible macroblock ordering feature of H.264/AVC to hide message bits.

Macroblocks are assigned to arbitrary slice groups according to the content of the message bits to be hidden. A maximum payload of three message bits per macroblock is achieved. The proposed solutions are analyzed in terms of message extraction accuracy,message payload, excessive bitrate and quality distortion.Comparisons with previous work reveal that the proposed solutions are superior in terms of message payload while causing less distortion and compression overhead.

3. PROPOSED SCHEME

In this section, a novel scheme of data hiding in the encrypted version of H.264/AVC videos is presented, which includes three parts, i.e., H.264/AVC video encryption, data embedding and data extraction. The content owner encrypts the original H.264/AVC video stream using standard stream ciphers with encryption keys to produce an encrypted video stream.

Then, the data-hider can embed the additional data into the encrypted video stream by using codeword substituting method, without knowing the original video content. At the receiver end, the hidden data extraction can be accomplished either in encrypted or in decrypted version.

3.1 Encryption of H.264/AVC Video Stream:

Video encryption requires that the scheme be time efficient to meet the requirement of real time and format compliance. It is not practical to encrypt the whole compressed video bitstream like what the traditional ciphers do because of the following two constraints, i.e., format compliance and computational cost. Alternatively, only a fraction of video data is encrypted to improve the efficiency while still achieving adequate security.

The key issue is then how to select the sensitive data to encrypt. According to the analysis given in, it is reasonable to encrypt both spatial information (IPM and residual data) and motion information (MVD) during H.264/AVC encoding. In this paper, an H.264/AVC video encryption scheme with good performance including security, efficiency, and format compliance is proposed.

An H.264/AVC video encryption scheme with good performance including security, efficiency, and format compliance is been proposed. By analyzing the property of H.264/AVC codec, three sensitive parts (i.e., IPMs, MVDs, and residual coefficients) are encrypted with stream ciphers.Selective encryption in the H.264/AVC compressed domain has been already put forth on context-adaptive variable length coding (CAVLC) and even on context-adaptive binary arithmetic coding (CABAC). We encrypt the codewords of IPMs, the codewords of MVDs, and the codewords of residual coefficients.

The encrypted bitstream is still H.264/AVC compliant and can be decoded by any standard-compliant H.264/AVC decoder, but the encrypted video data is treated completely different compared to plaintext video data.

3.1.1 Intra-Prediction Mode (IPM) Encryption:

According to H.264/AVC standard, there are four different types of intra coding are supported, which are denoted as Intra_4×4, Intra_16×16, Intra_chroma, and I_PCM. Four intra prediction modes (IPMs) are available in the Intra_16 ×16.

3.1.2 Motion Vector Difference (MVD) Encryption:

Further to protect both texture information and motion information, not only the IPMs but also the motion vectors should be encrypted. In H.264/AVC, motion vector prediction is further carried out on the motion vectors, which yields MVD.

In H.264/AVC baseline profile, Exp- Golomb entropy coding is used to encode MVD encryption may change the sign of MVD, but does not affect the length of the codeword and satisfies the format compliance.

3.1.3 Residual Data Encryption:

In order to keep high security, another type of sensitive data, i.e., the residual data in both the I-frames and P-frames should be encrypted. In this region, a novel method for encrypting the residual data based on the characteristics of codeword. In H.264/AVC baseline profile, CAVLC entropy coding is used to encode the quantized coefficients of a residual block.

3.2 Data Embedding:

In the encrypted bit stream of H.264/AVC, the proposed data embedding is accomplished by substituting eligible code words. On the other hand, the code words substitution should fulfill the following three limitations. First, the bit stream after codeword substituting must remain syntax compliance so that it can be decoded by standard decoder. Second, to keep the bit-rate unchanged, the substituted codeword should have the same size as the original codeword.

3.3 Data Extraction:

In this scheme, the hidden data can be extracted either in encrypted or decrypted domain. Data extraction process is fast and simple.

Scheme I: Encrypted Domain Extraction.
Scheme II: Decrypted Domain Extraction.

Scheme I: Encrypted Domain Extraction.

To protect privacy, a database manager may only get access to the data hiding key and have to manipulate data in encrypted domain. Data extraction in encrypted domain guarantees the feasibility in it

Scheme II: Decrypted Domain Extraction.

In scheme I, both embedding and extraction of the data are performed in encrypted domain. However, in some cases, users want to decrypt the video first and extract the hidden data from the decrypted video. For example, an authorized user, which owned the encryption key, received the encrypted video with hidden data. The received video can be decrypted using the encryption key. That is, the decrypted video still includes the hidden data, which can be used to trace the source of the data.

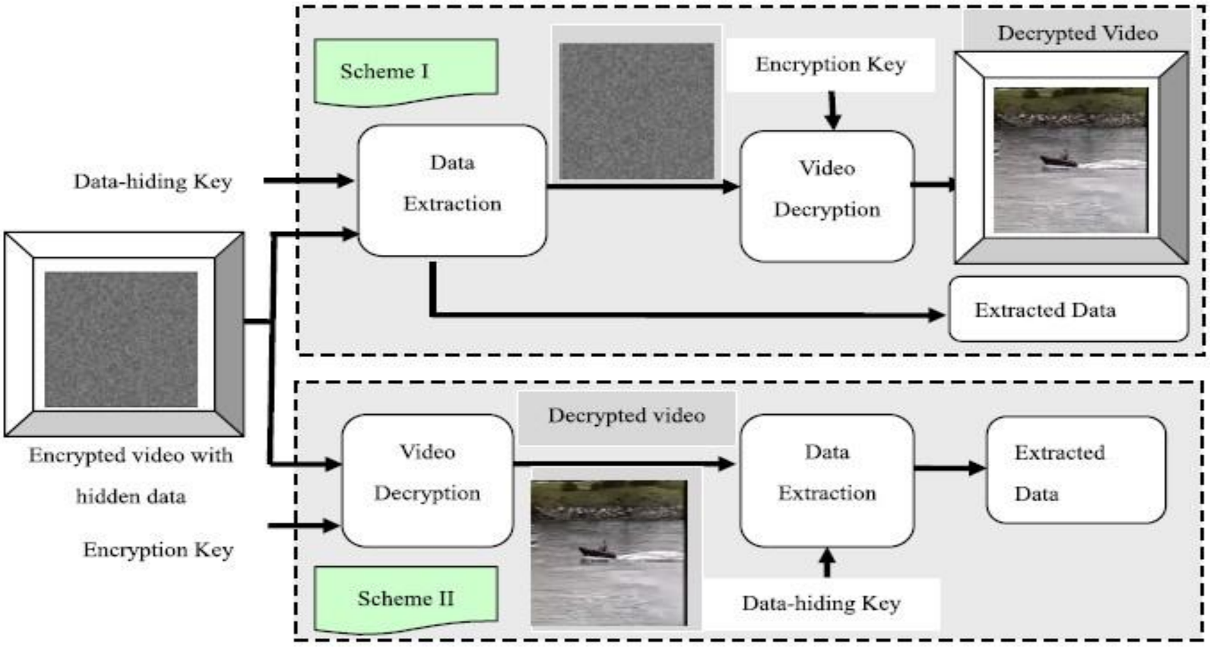

Fig 1: Data extraction in Encrypted and Decrypted domain

Fig 2: Original video frames

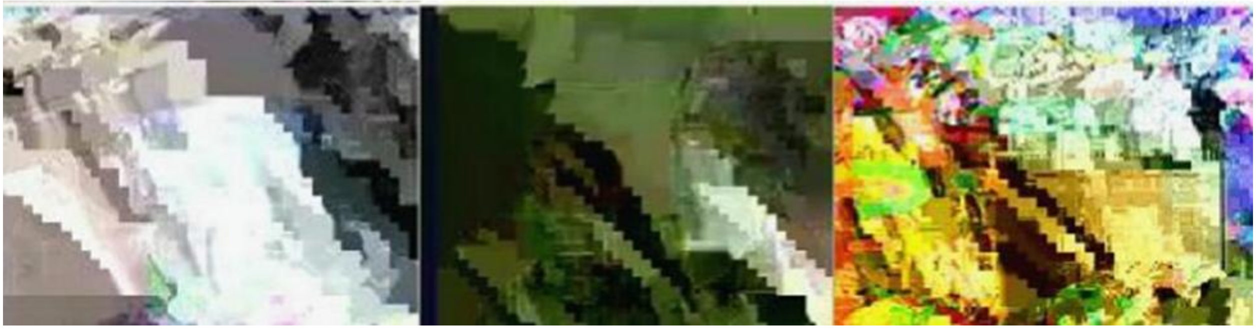

Fig 3: Encrypted video frame

The whole process of decryption and data extraction is given as follows.

Step1: Generate encryption streams with the encryption keys as given in encryption process.

Step2: The code words of IPMs, MVDs, Sign_of_TrailingOnes and Levels are identified by parsing the encrypted bit stream.

Step3: The decryption process is identical to the encryption process, since XOR operation is symmetric. The encrypted code words can be decrypted by performing XOR operation with generated encryption streams, and then two XOR operations cancel each other out, which renders the original plain-text. Since the encryption streams depend on the encryption keys, the decryption is possible only for the authorized users.

After generating the decrypted code words with hidden data, the content owner can further extract the hidden information.

Step4: The last bit encryption may change the sign of Level. The encrypted codeword and the original codeword are still in the same code spaces.

Step5: Generate the same pseudo-random sequence that was used in embedding process according to the data hiding key. The extracted bit sequence should be decrypted to get the original additional information.

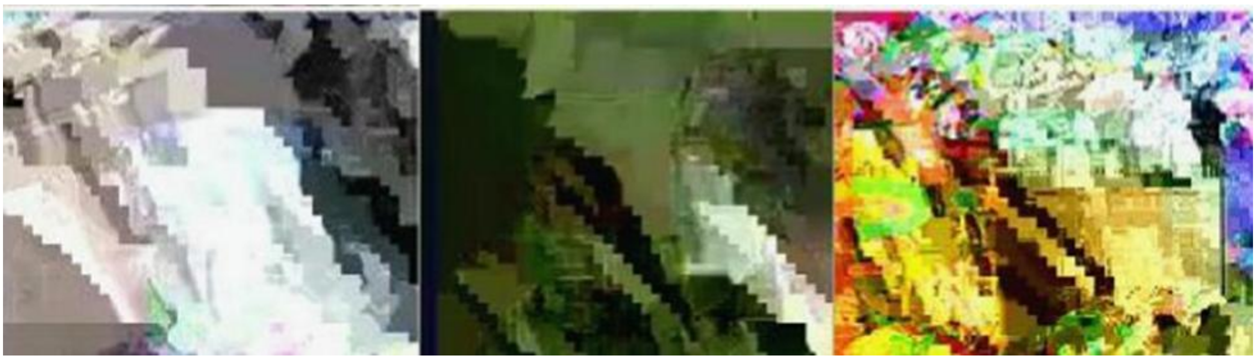

Fig. 4: Encrypted video frames with hidden data.

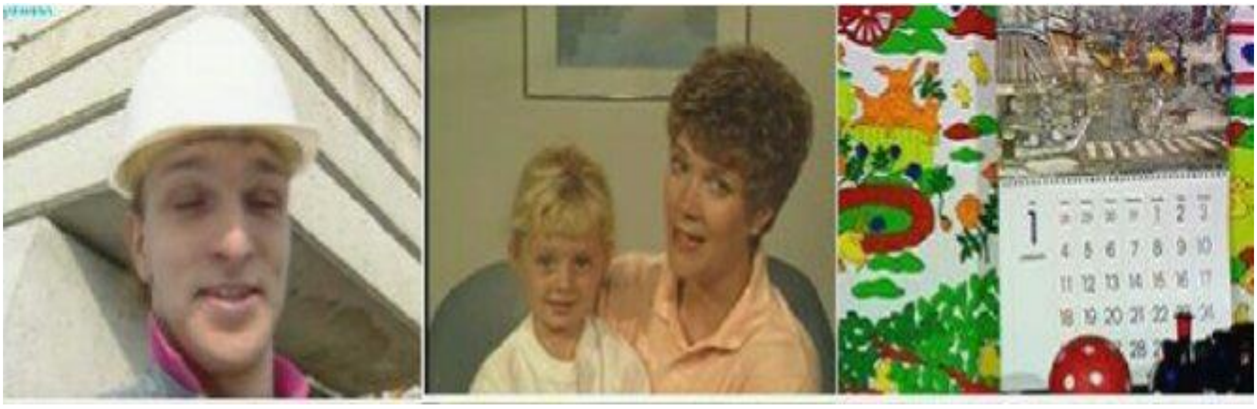

Fig. 5: Decrypted video frames with hidden data.

4. CONCLUSION

Data hiding in encrypted media is a new topic that has started to draw attention because of the privacy-preserving requirements from cloud data management. The data-hider can embed additional data into the encrypted bit stream using codeword substituting, even though does not know the original video content. Since data hiding is completed entirely in the encrypted domain, proposed method can preserve the confidentiality of the content completely. Another advantage is that it is fully compliant with the H.264/AVC syntax. Experimental results have shown that the proposed encryption and data embedding scheme can preserve file-size, whereas the degradation in video quality caused by data hiding is quite small.

5. REFERENCES

[1] W. J. Lu, A. Varna, and M. Wu, “Secure video processing: Problems and challenges,” in Proc. IEEE Int.Conf. Acoust. Speech, Signal Processing, Prague, Czech Republic, May 2011, pp. 5856–5859.

[2] B. Zhao, W. D. Kou, and H. Li, “Effective watermarking scheme in the encrypted domain for buyer-seller watermarking protocol,” Inf. Sci., vol. 180, no. 23, pp. 4672–4684, 2010.

[3] P. J. Zheng and J. W. Huang, “Walsh-Hadamard transform in the homomorphic encrypted domain and its application in image watermarking,” in Proc. 14th Inf. Hiding Conf., Berkeley, CA, USA, 2012, pp. 1–15.

[4] W. Puech, M. Chaumont, and O. Strauss, “A reversible data hiding method for encrypted images,” Proc. SPIE, vol.6819, pp.68191E-1–68191E9, Jan.2008.

[5] X. P. Zhang, “Reversible data hiding in encrypted image,” IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.

[6] W. Hong, T. S. Chen, and H. Y. Wu, “An improved reversible data hiding in encrypted images using side match,” IEEE Signal Process. Lett., vol. 19, no. 4, pp. 199–202, Apr. 2012

[7] X. P. Zhang, “Separable reversible data hiding in encrypted image,” IEEE Trans. Inf. Forensics Security, vol.7, no. 2, pp. 826–832, Apr. 2012.

[8] K. D. Ma, W. M. Zhang, X. F. Zhao, N. Yu, and F. Li, “Reversible data hiding in encrypted images by reserving

room before encryption," IEEE Trans. Inf. Forensics Security, vol.8,no.3,pp.553–562,Mar. 2013.

[9] A. V. Subramanyam, S. Emmanuel, and M. S. Kankanhalli, "Robust watermarking of compressed and encrypted JPEG2000 images," IEEE Trans. Multimedia, vol. 14, no. 3, pp. 703–716, Jun. 2012.

[10] S. G. Lian, Z. X. Liu, and Z. Ren, "Commutative encryption and watermarking in video compression," IEEE Trans. Circuits Syst. Video Technol., vol. 17, no. 6, pp. 774–778, Jun. 2007.

[11] S. W. Park and S. U. Shin, "Combined scheme of encryption and watermarking in H.264/scalable video coding (SVC)," New Directions Intel. Interact. Multimedia, vol. 142, no. 1, pp. 351–361, 2008.

[12] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 560–576, Jul. 2003.

[13] S. G. Lian, Z. X. Liu, Z. Ren, and H. L. Wang, "Secure advanced video coding based on selective encryption algorithms," IEEE Trans.Consumer Electron., vol. 52, no. 2, pp. 621–629, May 2006.

[14] Z. Shahid, M. Chaumont, and W. Puech, "Fast protection of H.264/AVC by selective encryption of CAVLC and CABAC for I and P frames,"IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 5, pp. 565–576, May 2011.

[15] M. N. Asghar and M. Ghanbari, "An efficient security system for CABAC bin-strings of H.264/SVC," IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 3, pp. 425–437, Mar. 2013.

[16] T. Stutz and A. Uhl, "A survey of H.264 AVC/SVC encryption,"IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 3pp. 325–339, Mar. 2012.

[17] Advanced Video Coding for Generic Audiovisual Services, ITU, Geneva, Switzerland, and Mar. 2005.

[18] J. G. Jiang, Y. Liu, Z. P. Su, G. Zhang, and S. Xing, "An improved selective encryption for H.264 video based on intra prediction mode scrambling," J. Multimedia, vol. 5, no. 5, pp. 464-472, 2010.

[19] I. E. G. Richardson, H.264 and MPEG-4 Video Compression: Video Coding for Next Generation Multimedia. Hoboken, NJ, USA: Wiley, 2003.

[20] D. K. Zou and J. A. Bloom, "H.264 stream replacement watermarking with CABAC encoding," in Proc. IEEE ICME, Singapore, Jul. 2010, pp. 117–121.

[21] D. W. Xu and R. D. Wang, "Watermarking in H.264/AVC compressed domain using Exp-Golomb code words mapping," Opt. Eng., vol. 50,no. 9, p. 097402, 2011.

[22] D. W. Xu, R. D. Wang, and J. C. Wang, "Prediction mode modulated data-hiding algorithm for H.264/AVC," J. Real-Time Image Process.,vol. 7, no. 4, pp. 205–214, 2012.

[23] T. Shanableh, "Data hiding in MPEG video files using multivariate regression and flexible macro block ordering," IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 455–464, Apr. 2012.