ISSN:2348-2079

Volume-5 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Efficient steganography in encoded video streams using motion vector difference

[1]Mrs K.E.Eswari M.C.A., M.Phill., ME., Associate Professor,
[2]Mr A.Rinshad Final MCA,
Department of MCA, Nandha Engineering College (Autonomous), Erode-52.
E-Mail ID: eswari.eswaramoorthi@nandhaengg.org, rinshad7070mft@gmail.com

Abstract - Generally, digital video sometimes are stored and processed in an encrypted format to maintain privacy and security. For the purpose of content notation, 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. This thesis proposes a novel scheme of data hiding directly in the encrypted version of AVI video stream, which includes the following three parts, i.e., AVI video encryption, data embedding, and data extraction. By analyzing the property of AVI codec and the code words of motion vector differences 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. Experimental results have demonstrated the feasibility and efficiency of the proposed scheme.

Index Terms—Data hiding, encrypted domain, H.264/AVC, codeword substituting.

1. INTRODUCTION

CLOUD computing has become an important technology trend, which can provide highly efficient computation and large-scale storage solution for video data. Given that 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; video streams would avoid the Leakage of video content, which can help address the security and privacy concerns with cloud computing [1]. In thesis study describing the becomes highly desirable to develop data hiding algorithms that work entirely on encoded bit stream in the encrypted domain However, there are some significant challenges for data hiding directly in compressed and encrypted bit stream. The first challenge is to determine where and how the bit stream can be modified so that the encrypted bit stream with hidden data is still a compliant compressed bit stream. 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.

The Proposed scheme is made up of image/video encryption, data embedding and data-

extraction/image-recovery phases. The content owner encrypts the original uncompressed image using an encryption key to produce an encrypted image. Then, the data-hider compresses the Least Significant Bits (LSB) of the encrypted image using a data-hiding key to create a sparse space to accommodate the additional data. At the receiver side, the data embedded in the created space can be easily retrieved from the encrypted image containing additional data according to the data-hiding key. Since the data embedding only affects the LSB, a decryption with the encryption key can result in an image similar to the original version. When using both of the encryption and data-hiding keys, the embedded additional data can be successfully extracted and the original image can be perfectly recovered by exploiting the spatial correlation in natural image/video.

2. LITERATURE REVIEW

The most watermarking schemes for copyright protection, a seller usually embed a watermark in multimedia content to identify a buyer [2]. When an unauthorized copy is found by the seller, the traitor's identity can be traced by the embedded watermark. However, it incurs both repudiation issue and framing issue. To solve these problems, some buyer seller watermarking protocols

have been proposed based on watermarking scheme in the encrypted domain. The enhanced scheme increases effective watermarking capacity, avoids additional overhead and overcomes an inherent defect that watermarking capacity depends on the probability distribution of input watermark sequence. Based on the security requirements of buyer–seller watermarking protocols, a new watermarking scheme in the encrypted domain with flexible watermarking capacity is proposed.[5] It improves the robustness of watermark sequence against image compressions and enables image tampering detection. Watermark extraction is blind, which employs the same threshold criterion and secret keys as watermark embedding. The secure and authenticated discrete reversible data hiding in cipher images deals with security and authentication [4]. In the first phase, a content owner encrypts the original uncompressed image using an encryption key. Then, a data hider may compress the least significant bits of the encrypted image using a data hiding key to create a sparse space to accommodate some additional data. With an encrypted image containing additional data, if a receiver has the data hiding key, receiver can extract the additional data though receiver does not know the image content. If the receiver has the encryption key, can decrypt the received data to obtain an image similar to the original one.

3. EXISTING METHODOLOGY

In existing system, the Motion Vector Difference (MVD) Encoding is carried out as follows. In order to protect both texture information and motion information, not only the IPMs but also the motion vectors should be encoded. In avi file, motion vector prediction is further performed on the motion vectors, which yields MVD. The values of MVDs are taken.

Data Embedding: In the encrypted bit stream of avi frames, the proposed data embedding is accomplished by substituting eligible code words of various Levels. Since the sign of Levels are encrypted, data hiding should not affect the sign of Levels.

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

3.1 Drawbacks

- It does not decoding or partial decompression of the video stream and the final video is always in encrypted format.

- Perturbing the raw data is not carried out and encryption mechanism of raw text data is not discussed.

4. PROPOSED SYSTEM ALGORITHMS

4.1 Video File Parsing

In this process, the video file's number of frames is found out and extracted using AviFil32.dll methods. The frames are saved in a folder.

4.2 Text Data Input and Perturbation

- Text message selection.
- Two random characters are inserted between each two consecutive characters in the text message and the message is perturbated (confused).

4.3 Encrypted Data Embedding

- Text data is selected.
- Bit sequences of the perturbated data is taken for hiding. Frame data of the video is encoded with different pixel values. Using the given data hiding key, the data embedding process is carried out with the given encrypted data.
- The encrypted data is made to hide inside the frames in the least significant bits.

4.4 Encrypted Data Extracting and Decryption

- The encrypted video with the hidden data is selected.
- For data extraction, Data-hiding key is given and the data is first extracted and then decrypted. Then with the video decryption key (same as encryption key), the video is decrypted and original video is obtained.
- First data extraction followed by Video decryption or Video decryption followed by data extraction.

5. RESEARCH METHODOLOGY

A novel scheme of data hiding in the encrypted version of AVI videos is presented, which includes three parts, i.e., AVI video encryption, data embedding and data extraction. The content owner encrypts the original AVI video stream using standard stream ciphers with encryption keys to produce an encrypted video stream. Then, the data-hider (e.g., a cloud server) 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.

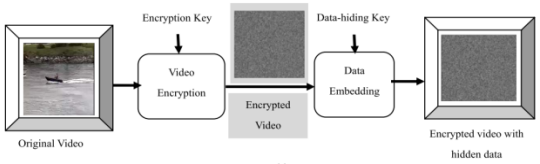


Fig 1 Video encryption and data embedding at the sender end.

5.1 Encryption Of Avi Video Stream

Video encryption often 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 bit stream 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.

In this study, an AVI video encryption scheme with good performance including security, efficiency, and format compliance is proposed. By analyzing the property of AVI codec, three sensitive parts (i.e., IPMs, MVDs, and residual coefficients) are encrypted with stream ciphers. Compared with the proposed encryption algorithm is performed not during AVI encoding but in the

AVI compressed domain. In this case, the bit stream will be modified directly. Selective encryption in the AVI compressed domain has been already presented on Context-Adaptive Variable Length Coding (CAVLC) and Context-Adaptive Binary Arithmetic Coding (CABAC). In this study, improved and enhanced the previous proposed approach by encrypting more syntax elements. We encrypt the code words of IPMs, the code words of MVDs, and the code words of residual coefficients. The encrypted bit stream is still AVI compliant and can be decoded by any standard-compliant AVI decoder, but the encrypted video data is treated completely different compared to plaintext video data.

5.1.1 Intra-Prediction Mode (IPM) Encryption:

According to AVI standard, the following four types of intra coding are supported, which are denoted as Intra_4 × 4, Intra_16× 16, Intra_chroma, and I_PCM. Here, IPMs in the Intra_4 × 4 and Intra_16 × 16 blocks are chosen to encrypt. Four intra prediction modes (IPMs) are available in the Intra_16 × 16. The IPM for Intra_16 × 16 block is specified in the mb_type (macroblock type) field which also specifies other parameters about this block such as coded block pattern (CBP).

5.1.2 Motion Vector Difference (MVD) Encryption:

In order to protect both texture information and motion information, not only the IPMs but also the motion vectors should be encrypted. In AVI, motion vector prediction is further performed on the motion vectors, which yields MVD. In AVI baseline profile, Exp-Golomb entropy coding is used to encode MVD. The codeword of Exp-Golomb is constructed as [M zeros] [I N FO], where I N FO is an M-bit field carrying information.

5.1.3 Residual Data Encryption:

In order to keep high security, another type of sensitive data, i.e., the residual data in both I-frames and P-frames should be encrypted. In AVI baseline profile, CAVLC entropy coding is used to encode the quantized coefficients of a residual block. Each CAVLC codeword can be expressed as the following format:

{Coef f token, Sign of T railing Ones, Level, T otal zeros, Run bef ore}

5.2 Data Embedding

In the encrypted bit stream of AVI, the proposed data embedding is accomplished by substituting eligible code words of Levels. Since the sign of Levels are encrypted, data hiding should not affect the sign of Levels. Besides, the code words substitution should satisfy 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.
- Third, data hiding does cause visual degradation but the impact should be kept to minimum. That is, the embedded data after video decryption has to be invisible to a human observer.

So the value of Level corresponding to the substituted codeword should keep close to the value of Level corresponding to the original codeword. In addition, the code words of Levels within P-frames are used for data hiding, while the code words of Levels in I-frames are remained unchanged. Because I-frame is the first frame in a Group Of Pictures (GOPs), the error occurred in I-frame will be propagated to subsequent P-frames.

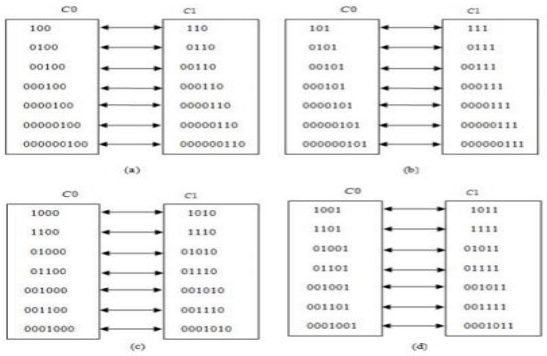


Fig. 3 . CAVLC codeword mapping. (a) su f f ix Length = 2& Level > 0. (b) su f f ix Length = 2& Level > 0. (c) su f f ix Length = 3& Level > 0. (d) su f f ix Length = 3& Level < 0

5.3 DATA EXTRACTION

Scheme I: **Encrypted Domain Extraction**

To protect privacy, a database manager (e.g., cloud server) may only get access to the data hiding key and have to manipulate data in encrypted domain.

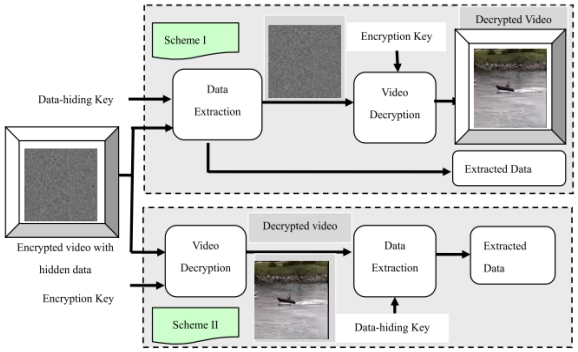


Fig 2 Data extraction and video display at the receiver end in two scenarios.

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. Data extraction in decrypted domain is suitable for this case.

Original codewords	01	010	00101	00100	0001011	0000100
Encryption stream	/	1	0	1	1	1
Encrypted codewords	01	011	00101	00101	0001010	0000101
Codespace	/	/	C0	C0	C1	C0
To-be-embedded data	Skip	Skip	1	0	0	1
Encrypted codewords with hidden data	01	011	00111	00101	0001000	0000111

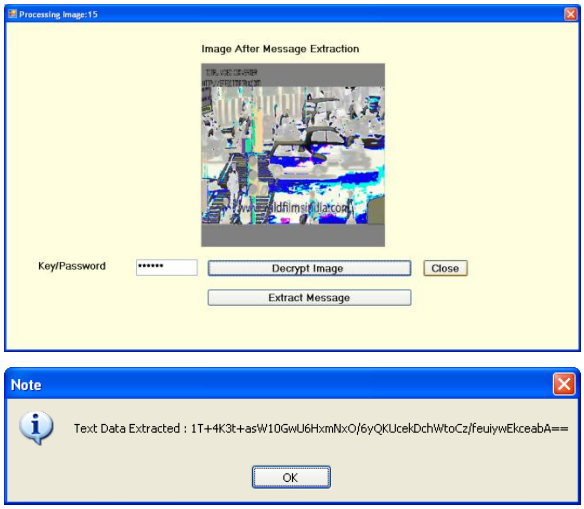
Fig 4: Data embedding

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

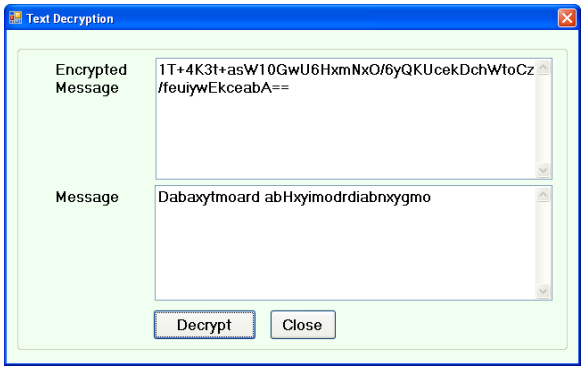
Step2: The code words of IPMs, MVDs, Sign_of_TrailingOnes and Levels are identified by parsing the encrypted bit stream.

6. RESULTS

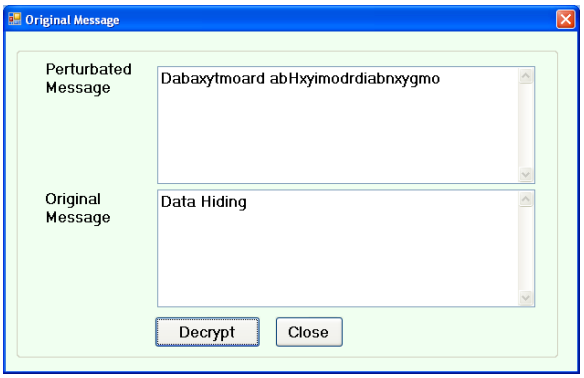
Extracting Image and Text



Text Decryption



Extracting Text



7. CONCLUSION AND FUTURE ENHANCEMENT

The reversible data hiding in encrypted image is investigated. Most of the work on reversible data hiding focuses on the data embedding/extracting on the plain spatial domain. But, in some applications, an inferior assistant or a channel administrator hopes to append some additional message, such as the origin information, image notation or authentication data, within the encrypted image though he does not know the original image content. And it is also hopeful that the original content should be recovered without any error after image decryption and message extraction at receiver side. A content owner encrypts the original image using an encryption key, and a data-hider can embed additional data into the encrypted image using a data-hiding key though he does not know the original content. With an encrypted image containing additional data, a receiver may first decrypt it according to the encryption key, and then extract the embedded data and recover the original image according to the data-hiding key. In the scheme, the data extraction is not separable from the content decryption.

In other words, the additional data must be extracted from the decrypted image, so that the principal content of original image is revealed before data extraction, and, if someone has the data-hiding key but not the encryption key, he cannot extract any information from the encrypted image containing additional data. At present, data hiding is completed entirely in the encrypted domain and the method can preserve the confidentiality of the content completely. With an encrypted video containing hidden data, data extraction can be carried out either in encrypted or decrypted domain. But, the video taken in the avi file only. In future various kinds of file formats can be taken for the entire process. Also, the data hiding process with no degradation in video quality can be carried out.

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

[4] 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.

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

[6] 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.

[7] P. J. Zheng and J. W. Huang, “Walsh-Hadamard transform in the homo- morphic encrypted domain

and its application in image watermarking," in Proc. 14th Inf.

[8] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images," Proc. SPIE, vol. 6819, pp. 68191E-1–68191E-9, Jan. 2008.

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

[10] 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.

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

[12] 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.

[13] 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.

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