Information Thrashing in to Encrypted H.264/AVC Videotape Streams by Secret Code Exchange

Ms.N.Zahira Jahan, MCA., M.Phil., Mr. S.Sugankumar,
Associate Professor, Final year,
Department of Computer Applications, Nandha Engineering College, Erode, India.

Abstract *— Digital videotape at times requirements near be stored and processed in an encrypted layout to keep sanctuary and seclusion. Used for the principle of comfortable information with tampering appreciation, it is basic to complete information thrashing in these encrypted video. Into this system, information thrashing into encrypted sphere personal of decryption protects the privacy of the restful .In reckoning, it is additional inventive present decryption follow by data thrashing along with encryption. Into this serious newspaper, a original format of information thrashing openly in the encrypted description of H.264/AVC videotape river is projected, which include the subsequent three part, videotape encryption, information embed, and information removal. By analyzing the possessions of H.264/AVC code, the codeword's of intraprediction and the codeword's of enduring coefficients are encrypted with torrent ciphers. Then, a in turn hider may surround supplementary data in the encrypted domain by using code word substitution procedure, without perceptive the original videotape contented. In instruct to adapt to different application scenarios, information extraction can be done whichever in the encrypted domain or in the decrypted domain. Videotape dossier extent is harshly conserved even after encryption and data embed. Tentative grades comprise established the probability in addition to good organization of the planned methods.*

Keywords-*Data hiding, encrypted domain, H.264/AVC, codeword substituting.*

I. INTRODUCTION

Cloud computing have develop into an imperative knowledge development, which be capable of supply very much competent subtraction with large-size storage space resolution meant for record information.

Agreed with the intention of shade forces could be a magnet for supplementary attack with be helpless en route for dishonest classification administrator, it is beloved with the intention of the videotape comfortable is reachable in encrypted appearance. The competence of drama information thrashing honestly in encrypted H.264/AVC videotape streams would avoid the seepage of record comfortable, which Manuscript received November conventional server. Natural Science Foundation under code.

II. PLANNED METHOD

During this segment, a narrative inspiration of information thrashing in the encrypted report of H.264/AVC video exists, which includes three parts, i.e., H.264/AVC video encryption, data embedding and data extraction. The comfortable possessor encrypts the innovative H.264/AVC video stream by average river ciphers among encryption key to manufacture an encrypted videotape stream. Afterward, the information-hider (e.g., a cloud server) container insert the added information addicted to the encrypted videotape stream by means of using open sesame substitute system, devoid of significant the inventive videotape pleased. On the receiver beginning, the hidden data extraction can be skilled whichever in encrypted or in decrypted description.

The planned configuration is shown in Fig. 1, somewhere the encryption and information embed are depict in measurement (a), in addition to the information pulling out and videotape decryption are revealed in element.

1. Encryption of H.264/AVC Videotape River

Video encryption repeatedly requires so as to the method be present instance competent toward assemble the condition of existent time and arrangement fulfilment. It is not realistic to encrypt the complete packed in video bit stream approximating the conventional cipher do since of the pursue two constraint, i.e., plan conformity and computational charge. Instead, just a whole of videotape information is encrypted to pick up the effectiveness even as unmoving achieves passable security. The key quandary is subsequently how to

decide on the understanding in order to encrypt. According to the examination given in, it is level headed to encrypt both spatial in order and proposal information (MVD) during H.264/AVC encoding.

During this document, an H.264/AVC tape encryption format through first-rate routine together with safekeeping, competence, along with design agreement is planned. Through analyzing the material goods of H.264/AVC codec, three perceptive parts (i.e., IPMs, MVDs, and residual coefficients) are encrypted with stream ciphers. compare with the planned encryption algorithm is performed not throughout H.264/AVC encoding excluding in the H.264/AVC packed in sphere of pressure. Into this case, the bit stream will be modified directly. Discriminatory encryption in the H.264/AVC packed collectively province has been already on hand on situation-adaptive inconsistent measurement lengthwise coding (CAVLC) and context-adaptive binary reckoning coding (CABAC).We have enhanced and better the preceding planned come nearby encrypting more sentence structure 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 H.264/AVC obedient and can be decoded by any typical-compliant H.264/AVC decoder, but the encrypted videotape information is treat absolutely special compared to plaintext videotape data. In fact, performing the format compliant. Diagram of proposed scheme. (a) Video encryption and data embedding at the sender end. (b) Data extraction and video display at the receiver end in two scenarios.

Encryption directly on the squashed bit stream is tremendously difficult as the internal states of the encoder have to be sealed, otherwise the lingering information is interpreted falsely which may easily lead to format desecration. In H.264/AVC, each Intra_4 × 4 luminance blocks are predicted opening its spatially neighbouring sample. exclusively, H.264/AVC offers nine calculation modes (0-8) for Intra_4×4 luminance blocks.

TABLE I

MACROBLOCK TYPES FOR I SLICES AND VARIABLE LENGTH OF CODEWORD IN H.264/AVC [17]

mb_type	Name of mb_type	Intra16x16 PredMode	Chroma CBP	Luma CBP	Codeword
1	I_16x16_0_0_0	0	0	0	010
2	I_16x16_1_0_0	1	0	0	011
3	I_16x16_2_0_0	2	0	0	00100
4	I_16x16_3_0_0	3	0	0	00101
5	I_16x16_0_1_0	0	1	0	00110
6	I_16x16_1_1_0	1	1	0	00111
7	I_16x16_2_1_0	2	1	0	0001000
8	I_16x16_3_1_0	3	1	0	0001001
9	I_16x16_0_2_0	0	2	0	0001010
10	I_16x16_1_2_0	1	2	0	0001011
11	I_16x16_2_2_0	2	2	0	0001100
12	I_16x16_3_2_0	3	2	0	0001101
13	I_16x16_0_0_1	0	0	15	0001110
14	I_16x16_1_0_1	1	0	15	0001111
15	I_16x16_2_0_1	2	0	15	000010000
16	I_16x16_3_0_1	3	0	15	000010001
17	I_16x16_0_1_1	0	1	15	000010010
18	I_16x16_1_1_1	1	1	15	000010011
19	I_16x16_2_1_1	2	1	15	000010100
20	I_16x16_3_1_1	3	1	15	000010101
21	I_16x16_0_2_1	0	2	15	000010110
22	I_16x16_1_2_1	1	2	15	000010111
23	I_16x16_2_2_1	2	2	15	000011000
24	I_16x16_3_2_1	3	2	15	000011001

The calculation method of the at this time well thought-out block E is denoted as Mode E. If Mode E is equal to MPME, only one bit is needed to signal the prediction mode. When Mode E and MPME are changed, the password is composed of one flag bit “0”and three bits fixed-length code.

In each codeword is encrypted with a pseudorandom progression which is generate via a standard protected cipher (e.g., RC4) single-minded by an encryption key E_Key2. Bitwise XOR procedure is still utilized as the encryption scheme.

From what described above, it is obvious that the length of the encrypted codeword is the same as the original one. designed for the format acquiescence in the decode process, the encrypted IPMs of blocks in the first row furthermore in the opening support should encompass the decidable value, since not all modes are available along the top and the missing margins of each surround due to the lack of neighbours.

TABLE II

MVDs AND CORRESPONDING EXP-GOLOMB CODEWORDS

MVD	*code_num*	*codeword*
0	0	1
1	1	010
-1	2	011
2	3	00100
-2	4	00101
3	5	00110
-3	6	00111
4	7	0001000
-4	8	0001001
5	9	0001010
-5	10	0001011
6	11	0001100
-6	12	0001101
7	13	0001110
-7	14	0001111
8	15	000010000
-8	16	000010001
9	17	000010010
-9	18	000010011
…	…	…

In our proposal, if the IPM\ after encryption is not available for a border block, then the IPM encryption of this block will be skipped. This further indicates that IPM encryption is not secure an adequate amount of in some specific locations and should be used in combination with last encrypting method. In summary, IPM encryption implies varying the fake mode to a further one lacking violating the semantics and bit stream conformity.

2. Activity Vector Difference (MVD) Encryption:

In charge to look after mutually consistency information and activity information, not only the IPMs but also the motion vectors must be encrypted. In H.264/AVC, motion vector calculation is supplementary performed on the motion vectors, which yields MVD. In H.264/AVC baseline contour, Exp Gloomy entropy coding is used to encode MVD. The codeword of Exp Gloomy is constructed as[M zeros] where I NFO is an M-bit field carrying information.

Table II shows the values of MVDs and correspondingExp-Golomb codewords. The last bit of the codeword is encrypted by applying the bitwise XOR operation with a standard stream cipher, which is unwavering by an encryption key E_Key3. According to Table II, the last bit encryption may change the sign of MVD, but does not change the

Length of the codeword and satisfies the format compliance. With the intention of is, the resulting cipher texts are still valid Exp-Glooms codes. For example, the codeword's corresponding to "2" and "−2" are "00100" and "00101", respectively, which have the equivalent duration. It should be distinguished that whilst the value of MVD is equal to 0, its equivalent codeword "1" keeps unchanged during the encryption process.

3. Remaining Information Encryption:

In order to keep high security, another type of sensitive data, i.e., the outstanding data in both I-frames and P-frames must be encrypted. In this section, a novel system for encrypting the outstanding information based on the description of codeword is accessible in element.

In H.264/AVC 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:

{Cafe token, Sign of Trailing Ones,

Level, Total zeros, Run be}

The explicit purpose of each sentence structure component is described in. on the way to keep the bit stream compliant, not all syntax elements can be made to order all through encryption process. Meant for illustration, Coef f symbol, total zeros, and Run bed ore should remain unchanged. Consequently, persistent data encryption can be proficient by modifying the codewords of Sign_of_TrailingOnes and Level.

The encoded with a single bit. Bit "0" is assigned for +1 and bit "1"is assigned for −1. The codeword of Sign_of_TrailingOnes is encrypted by applying the bitwise XOR operation with a standard stream cipher, which is determined by an encryption key E_Key4. The codeword for each Level is made up of a prefix (level_prefix) and a suffix (level_suffix) as

Table III shows Levels with different suffix Length and corresponding code words. The last bit of the codeword is encrypted by applying the bitwise XOR operation with a standard stream cipher, which is determined by an encryption key E_Key5. According to Table III, the last bit encryption may change the sign of Levels, but does not affect the length of the codeword and satisfies the format compliance. For example, when is equal to 1, the codewords corresponding to "2" and "-2" are "010" and "011", respectively, which have the same length. It should be noted that when suffix Length is equal to 0, the codewords should keep unchanged during the encryption process.

III. INFORMATION EMBEDDING

Though few method have been proposed to embed information into H.264/AVC bit stream directly on the other hand, these methods cannot be implemented in the encrypted domain. In the encrypted bit stream of H.264/AVC, the proposed data embedding is accomplished by substituting eligible codewords of Levels in Table III. Since the sign of Levels are encrypted, data beating should not affect the sign of Levels. Besides, the codewords substitution should gratify the following three limitations. First, the bit stream after codeword substituting must remain syntax disobedience so that it can be decoded bystander decoder. Second, to keep the bit-rate unmovable, thesubstituted codeword should have the same size as the original codeword. Third, information hiding does cause visual degradation but the shock 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 secret word should keep close to the value of Level corresponding to the original codeword. In addition, the codewords of Levels within P-frames are used for data hiding, while the codewords 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.

According to the analysis prearranged above, we can see that there are no corresponding substituted when is equal to 0 or 1, as revealed in Table III. When is equal to 0, we cannot find a pair of codewords with the same size. When is equal to 1, one codeword also cannot be substituted by another codeword with the same size, this substitution would change the sign of Level. Then the codewords of Levels which suffix Length is 2 or 3 would be

TABLE III

LEVELS AND CORRESPONDING CODEWORDS

suffixLength	*Level*(>0)	*Codeword*	*Level*(<0)	*Codeword*
0	1	1	-1	01
	2	001	-2	0001
	3	00001	-3	000001
	4	0000001	-4	00000001
1	1	10	-1	11
	2	010	-2	011
	3	0010	-3	0011
	4	00010	-4	00011
	5	000010	-5	000011
	6	0000010	-6	0000011
	7	00000010	-7	00000011
	8	000000010	-8	000000011
2	1	100	-1	101
	2	110	-2	111
	3	0100	-3	0101
	4	0110	-4	0111
	5	00100	-5	00101
	6	00110	-6	00111
	7	000100	-7	000101
	8	000110	-8	000111
	9	0000100	-9	0000101
	10	0000110	-10	0000111
	11	00000100	-11	00000101
	12	00000110	-12	00000111
	13	000000100	-13	000000101
	14	000000110	-14	000000111
3	1	1000	-1	1001
	2	1010	-2	1011
	3	1100	-3	1101
	4	1110	-4	1111
	5	01000	-5	01001
	6	01010	-6	01011
	7	01100	-7	01101
	8	01110	-8	01111
	9	001000	-9	001001
	10	001010	-10	001011
	11	001100	-11	001101
	12	001110	-12	001111
	13	0001000	-13	0001001
	14	0001010	-14	0001011

(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. divided into two opposite codes paces denoted as C0and C1 as shown in Fig. 2.

The codeword assigned in C0and C1 are associated with dual hidden information “0” and “1”.

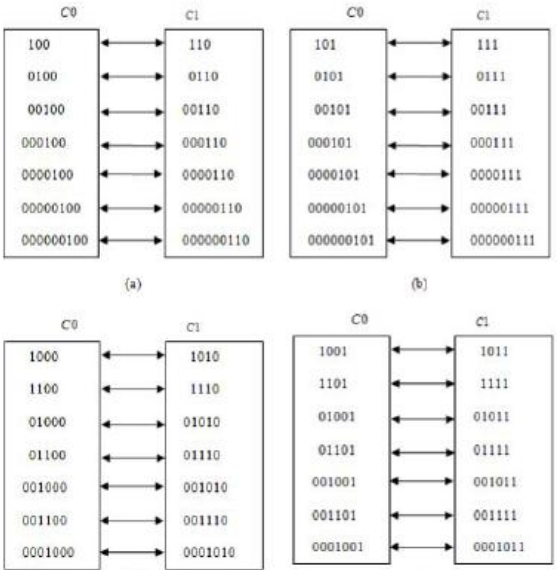

Fig. 2. CAVLC codeword mapping.

assume the additional data that we want to embed is a binary sequence denoted as B = {b(i)|i = 1, 2, · · · , L, b(i) ∈{0, 1}}. Data hiding is performed directly in encrypted bit stream through the following steps.

Step1. In order to enhance the security, the additional data is encrypted with the chaotic pseudo-random sequence P = {p(i)|i = 1, 2, · · · , L, p(i) ∈ {0, 1}} to generate the to-be-embedded sequence W = {w(i)|i = 1, 2, (i) ∈{0, 1}}. The sequence P is generated by using logistic map with an initial value i.e., the data hiding key. It is very difficult for anyone who does not retain the data hiding key to recover the additional data.

Step2. The codewords of Levels are obtained by parsing the encrypted H.264/AVC bitstream.

Step3. If current codeword belongs to codeword is "1000"which belongs to C0 as shown in Fig. 2(c). If the data bit "1" needs to be embedded, the codeword "1000" should be replaced with "1010". Otherwise, if the data bit "0" needs to embedded, the codeword "1000" will keep unchanged.

Step4. Choose the next codeword and then go to Step3 for data hiding. If there are no more data bits to be embedded, the embedding process is stopped.
Suppose the to-be-embedded data is "1001", the CAVLC codeword of Level parsing from H.264/AVC is"01 010 00100 00100 0001011 0000100" and the encryption torrent is "10111", an example of data embedding based on
Codeword mapping is shown in Fig. 4(a).

```
Procedure
  if (data bit= =0)
  {
        if (the codeword belongs to C0)
           The codeword is unmodified;
        else if (the codeword belongs to C1)
           The codeword is replaced with the corresponding codeword in C0.
  }
  else if (data bit= =1)
  {
        if (the codeword belongs to C1)
           The codeword is unmodified;
        else if (the codeword belongs to C0)
           The codeword is replaced with the corresponding codeword in C1.
  }
```

C. Data Extraction

In this scheme, the hidden data can be extracted either in encrypted or decrypted domain, as shown in Fig. 1(b). Data extraction process is fast and simple. We will first discuss the extraction in encrypted domain followed by decrypted domain.
1) 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. Data extraction in encrypted domain guarantees the feasibility of our scheme in this case. In encrypted domain, as shown in Fig. 1(b), encrypted video with hidden data is directly sent to the data extraction module, and the extraction process is given as follows.

Step1: The Levels are firstly identified by parsing the encrypted bit stream.

Step2: If the codeword belongs to codes pace C0, the extracted data bit is "0". If the codeword belongs to codespace C1, the extracted data bit is "1".

Step3: According to the data hiding key, the same chaotic pseudo-random sequence P that was used in the embedding process can be generated. Then the extracted bit sequence could be decrypted by using P to get the original additional information. Since the whole process is entirely operated in encrypted domain, it effectively avoids the leakage of original video content. An example of data extraction in encrypted domain is shown in Fig. 4(b). Users wanton decrypt the video first and extract the hidden data frothed decrypted video. For example, an authorized user, whichowned the encryption key, received the encrypted video with unhidden 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

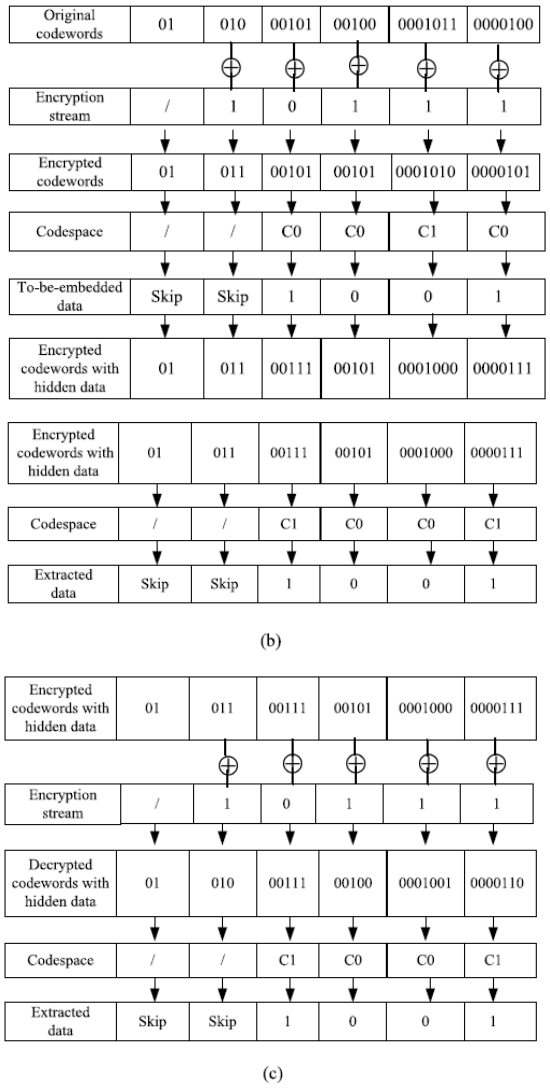

Fig. 4. An example of data embedding and extraction. (a) Data embedding. (b) Data extraction in encrypted domain. (c) Data extraction in decrypted domain case. As shown in Fig. 1(b), the received encrypted video with hidden data is first pass through the decryption module. The whole process of decryption and data extraction is given as follows.

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

Step2: The codeword 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 encryptedcodewords can be decrypted by performing XOR operation
with generated encryption streams, and then two XOR operations cancel each other out, which renders the original plaintext. Since the encryption streams depend on the encryption keys, the decryption is possible only for the authorized users.
After generating the decrypted codewords with hidden data, the content owner can further extract the hidden information.

Step4: According to Table III, the last bit encryption may change the sign of Level. However, as shown in Fig. 2, the encrypted codeword and the original codeword are still in the same code spaces. If the decrypted codeword of Level belongs to code space C0, the extracted data bit is "0". If the decrypted codeword of Level belongs to codes pace C1, the extracted data bit is "1".

Step5: Generate the same pseudo-random sequence P 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. An example of data extraction in decrypted domain
.

III. Experimental Grades

The projected numbers hiding scheme has been implemented in the H.264/AVC reference software version JM-12.2. Six well-known standard video sequences (i.e., Stefan, Table, Tempter, Mobile, Hall, and News) in QCIF format (176 × 144) at the frame rate 30 frames/s are used for simulation. The first 100 frames in each video sequence are used in the experiments. The GOP (Group of Pictures) structure is "IPPPP: one I frame followed four P frames".

A. Protection of Encryption Algorithm

For the proposed video encryption scheme, the security includes both cryptographic security and perceptual security. Cryptographic security denotes the security against cryptographic attacks, which depends on the ciphers adopted by the scheme. In the proposed scheme, the secure stream cipher (e.g., RC4) is used to encrypt the bit stream, and chaotic pseudo-random sequence generated by logistic map is used to encrypt the additional data. They have been proved to be secure against cryptographic attacks. Perceptual security refers to whether the encrypted video is unintelligible or not.

Generally, it depends on the encryption scheme's properties. For example, encrypting only IPM cannot keep secure enough, since the encrypted video is intelligible. The proposed scheme encrypts IPM, MVD and residual coefficients, which keeps perceptual security of the encrypted video. The demonstration is shown in Figs. 5 and 6. An original frame from each video is depicted in Fig. 5, and their corresponding encrypted results are depicted in Fig. 6. Other frames have a similar effect of encryption.

Due to space limitations, we do not list the results of all frames. It should be mentioned that not every video can be degraded to the same extent. The perceptual quality of high-motion videos with a complex textured background becomes much more scrambled after encryption than that of low-motion videos with a static background. The reason is that there are less residual coefficients and MVDs in low-motion videos that are available for encryption. In general, scrambling performance of the described encryption system is more than adequate.

B. Optical Class of Stego Videotape

The encrypted video containing hidden data provided by the server should be decrypted by the authorized user. Therefore, the visual quality of the decrypted video containing hidden data is expected to be equivalent or very close to that of the original video. By modifying the compressed bit stream to embed additional data, the most important challenge is to maintain perceptual transparency, which refers to the modification of bit stream should not degrade the perceived content

Quality only the code words of Levels within P-frames are modified for data hiding. Simulation results have demonstrated that we can embed the additional data with a large capacity into P-frames while preserving high visual quality. The encrypted and decrypted video frames with hidden data are shown in Figs 7 and 8 respectively.

IV. 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. In this paper, an algorithm to embed additional data in encrypted H.264/AVCbitstream is presented, which consists of video encryption, data embedding and data extraction phases. The algorithm can preserve the bit-rate exactly even after encryption and data embedding, and is simple to implement as it is directly performed in the compressed and encrypted domain, i.e., it does not require decrypting or partial decompression of the video stream thus making it ideal for real-time video applications. The data-hider can embed additional data into the encrypted bit stream using codeword substituting, even though he does not know the original video content. Since data hiding

Is completed entirely in the encrypted domain, our 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, which provides two different practical applications. 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 smallThe heading of the Acknowledgment section and the References section must not be numbered.

REFERENCES

[1] W. J. Lu, A. Varna, and M. Wu, "Secure video processing: Problems and challenges," in *Proc. IEEE Int. Conf. Account., 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. Zhen and J. W. Huang, "Walsh-Hadamard transform in the homomorphism 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-1–68191E-9, 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," *IEEETrans. Inf. Forensics Security*, vol. 8, no. 3, pp. 553–562, Mar. 2013.

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

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

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

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