Robust and fast selective encryption for HEVC videos

Mokhtar Ouamri, and Kamel Mohamed Faraoun

***Abstract* — Emerging High efficiency video coding (HEVC) is expected to be widely adopted in network applications for high definition devices and mobile terminals. Thus, construction of HEVC's encryption schemes that maintain format compliance and bit rate of encrypted bitstream becomes an active security's researches area. This paper presents a novel selective encryption technique for HEVC videos, based on enciphering the bins of selected Golomb–Rice code's suffixes with the Advanced Encryption Standard (AES) in a CBC operating mode. The scheme preserves format compliance and size of the encrypted HEVC bitstream, and provides high visual degradation with optimized encryption space defined by selected Golomb–Rice suffixes. Experimental results show reliability and robustness of the proposed technique.**

***Index Terms* — High efficiency video coding, Golomb–Rice code, Context-adaptive binary arithmetic coding, advanced encryption system.**

I. INTRODUCTION

High efficiency video coding (HEVC) is the latest video coding standard [1] developed by Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG) as a successor of H.264/AVC [2]. One of its primary objectives is to provide almost double compression efficiency at the cost of major computational complexity increase with respect to its predecessor H.264/AVC. It also support wide range of high definition video resolutions (from Full HD 1920x1080 to 4K Ultra HD and 8K Ultra HD) and several corresponding frame rates (30 FPS to 120 FPS).

The HEVC coding efficiency is optimized by improving the core of basic hybrid coding architecture of its predecessor H264/AVC by introducing several features and tools in all mains stages of compression including prediction, transformation, quantization, and entropy coding. Due to its decent coding performance, the emerging HEVC standard is expected to be widely adopted in network applications for HD devices and mobile terminals [3] such as ultra high-definition television UHDTV, streaming, and low delay communication. Security of such video applications is based on the protection of communicated HEVC videos using efficient encryption techniques, according to one of two possible encryption modes: a full encryption mode applying a global ciphering of the HEVC bitstream in detriment of the format compliance of the standard, and a selective encryption mode that only encipher some selected parts of the video data (transform coefficients, signs of motion vectors, syntax elements of entropy coder,…) without destroying the format compliance of the HEVC standard. In addition to HEVC format compliance, the selective encryption mode ensures the same bit rate ratio of encrypted bitstream as the original bitstream.

In order to apply selective encryption mode, a retrieval of the meaningful data to be encrypted must firstly be performed in order to get maximum visual degradation of the encrypted video sequences. Since the transform coefficients are the most widely employed as protected parts for selective encryption mode, we propose in this work to use the Golomb–Rice codes newly defined by the HEVC standard as selected parts to be protected.

The remaining of this paper is organized as follows: a brief description of HEVC structure with corresponding coding tools, coefficient level coding, and related encryption works are presented in Section II. Section III explain the procedure of level plaintext preparing and encryption/decryption. Section IV presents the different performed experiments with obtained result. Finally, conclusions are drawn in section V.

II. BACKGROUND AND RELATED WORKS

A. Description of HEVC structure and coding tools

The emerging high efficiency video coding (HEVC) is a new video coding standard, which introduces several tools and new concepts for a better compression of multiple existing picture resolutions, chiefly high definition (HD) and ultra high definition UHD spatial resolutions. To ensure a highest level of compression efficiency, the input video frame is first split into multiple coding-tree units (CTUs) with a maximum size of 64x64 pixels. A CTU is the basic unit of coding process and can be part of a slice (I_SLICE, P_SLICE, and B_SLICE) or a tile. Every CTU is a root of a quadtree structure that can be later divided into leaf level coding units (CUs) sharing the same mode of prediction. Each CU is partitioned further into prediction units (PUs), and can either uses intra-frame prediction (a whole of 35 modes are available) or inter frame prediction (uni-prediction or bi-prediction). The residual coding is done by partitioning each CU in a quadtree, and then defining the transform unit (TU) as a leaf of CU quadtree structure. The TU is the basic unit used for transform and quantization stage, and it has a dyadic block size varying from 4x4 to 32x32 samples. In addition to CTU, CU, PU, and TU,

Manuscript received May 25, 2014; revised December 15, 2014.

M. Ouamri is with the Department of Computer science, Djilalli Liabbes University, Sidi Bel Abbès, Algeria. (e-mail: amokhtar124@yahoo.fr).

K,M. Faraoun is with the Department of Computer science, Djilalli Liabbes University, Sidi Bel Abbès, Algeria. (e-mail: kamel_mh@yahoo.fr).

various new features are introduced in different stages of encoding/decoding operation of HEVC standard, and are explicitly highlighted in [1].

The CABAC (Context adaptive binary arithmetic coding) is the only one standard supported for entropy coding [3] in the HEVC, and it is an improved and simplified straightforward extension of the CABAC used in H.264/AVC standard [4]. The three main operations involved by the CABAC engine are depicted in Fig. 1: a binarization to decompose the non-binary syntax elements into a sequence of bins, a context modeling, and a binary arithmetic coding (BAC).Two operation modes are invoking from CABAC engine: a regular mode where the context model is required for coding bins, and a bypass mode where bins are coded with equi-probability. Many techniques were added to improve the throughput [3], including reducing context coded bins, grouping bypass bins together, grouping bins that use the same contexts together, reducing context selection dependencies, and reducing the total number of signaled bins. Furthermore, the HEVC define a new set of CABAC coded syntax elements for describing the properties of CU, PU, TU, and Loop filter. The draft of HEVC [6] presents several different binarization processes including unary coding, truncated unary coding, k^{th} order Exp-Golomb (EG_k) coding, Golomb–Rice coding and fixed length coding.

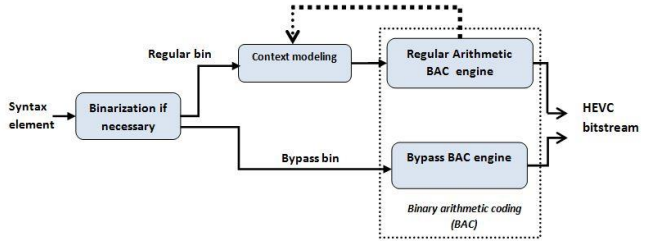

Fig. 1. Bloc diagram of CABAC engine with three key operations: binarization, context modeling and binary arithmetic coding (BAC).

B. Overview of coefficient level coding

A transform unit of size N x M consists of at least N x M quantized transform coefficients. The transform blocks TBs for TUs larger than 4x4 are decomposed into 4x4 sub-blocks unit, when each sub-block contains 16 consecutive quantized coefficients, encoded in inverse diagonal scan order. To encode coefficients level of each sub-block, five syntax elements are used to represent the coefficient's level information within the sub-block if it contains one or more non zero quantized transform coefficients: significant_coeff_flag, coeff_abs_level_greater1_flag, coeff_abs_level_greater2_flag, coeff_sign_flag, and coeff_abs_level_remaining. Table I describes the semantic of each cited syntax element.

Adaptive context models through regular mode is employed for encoding significant_coeff_flag, coeff_abs_level_greater1_flag and coeff_abs_level_greater2_flag, when remaining syntax elements coeff_sign_flag and coeff_abs_level_remaining are encoded by low complexity bypass mode.

The HEVC utilizes only Golomb–Rice code and k^{th} order Exp-Golomb (EG_k) for coeff_abs_level_remaining binarization [7] as depicted in Fig. 2. A Golomb–Rice code is an optimal code for representing a symbol value n and is defined by the quotient q=[n/m] and the remainder p=n-q.m, whereas m is a rice parameter. Quotient represents the prefix part binarized with a unary code, and remainder represents the suffix part composed by a fixed length bins. The Exp-Golomb code of symbol value n is obtained by concatenation of prefix and suffix codeword. The prefix is the unary code of $l(n)=\log_2((n/2^k)+1)$, whereas the suffix is calculated by $n+2^k(1-2^{l(n)})$.

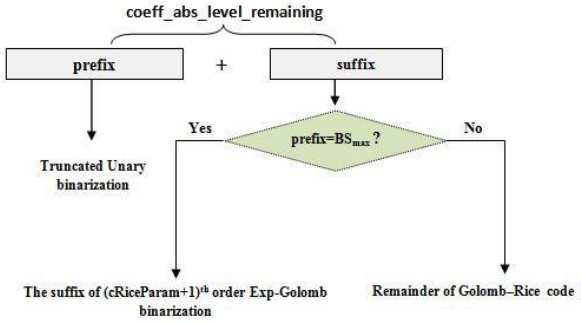

Fig. 2. Binarization of coeff_abs_level_remaining : prefix is binarized with truncated unary code, and suffix is binarized either by Golomb–Rice code or by Exp-Golomb code

TABLE I
DESCRIPTION OF CABAC SYNTAX ELEMENTS EMPLOYED FOR LEVEL CODING

Syntax element	Description
significant_coeff_flag	indicates the significance of each coefficient
coeff_abs_level_greater1_flag	indicates whether the coefficient amplitude is larger than one for each non zero coefficient
coeff_abs_level_greater2_flag	indicates whether the coefficient amplitude is larger than two for each coefficient with amplitude larger than one
coeff_sign_flag	indicates sign information of the nonzero coefficients
coeff_abs_level_remaining	indicates remaining absolute level value

According to [6] and [7], binarization of coeff_abs_level_remaining consists of a prefix part and a suffix part, and it depends on two parameters: cRiceParam and cTRMax. The cRiceParam parameter is ranging from 0 to 4 and it changes depending on the previously coded coeff_abs_level_remaining. Firstly, the truncated unary binarization is invoked to derive the prefix binstrings by binarizing the part Min(4, coeff_abs_level_remaining >> cRiceParam). If the prefix bin string is equal to a predefined bit string noted BS_{max}, then the suffix bin string is derived using the Exp-Golomb binarization for the suffix part (coeff_abs_level_remaining - cTRMax) with an order set equal to cRiceParam + 1, otherwise, the suffix string is the remainder of Golomb–Rice coding process specified by the

binary representation of the value computed by: (coeff_abs_level_remaining- ((coeff_abs_level_remaining>> cRiceParam) << cRiceParam) with a fixed length equal to cRiceParam.

The HEVC employs Golomb–Rice codes for short symbol's values, and Exp-Golomb codes for representing long symbol values. The Coefficients coeff_sign_flag are regrouped and encoded together for each sub-block, and are signaled before the coeff_abs_level_remaining of non zeros coefficients.

C. Related works

Selective encryption is a new trend for format-compliant content protection, privacy and security. It consists to choose a subset of data as protected part to be encrypted, and to let the remaining data unencrypted as a public part. The last decade is characterized by a number of important works on video selective encryption.

Various schemes have been suggested in compressed video especially for the H.264/AVC standard which uses CABAC and CAVLC for entropy coding. According to the protected part selected to be encrypted, the works proposed in [8] and [9] encrypt the DCT coefficient syntax elements (Non-zeros level and signs), while those in [10] and [11] scramble the intra mode prediction and the motion vector difference.

In [12] , the author proposes one of the first works allowing secure HEVC encoding by selective encryption. His works adopts a realization of HEVC level encoding with a binarization of coeff_abs_level_remaining using Golomb–Rice code for short symbol, and with zeroth Exp-Golomb (EG_0) for long symbol. The work focuses firstly on searching a set of dyadic codes of coeff_abs_level_remaining that can be encrypted without altering the format compliance of the HEVC bitstream. Then, the author prepares a plaintext formed by dyadic codes of coeff_abs_level_remaining and levels sign, and encrypted the resulting using AES-CFB mode. Unfortunately , since coeff_abs_level_remaining is binarized as described in [6], the scheme proposed in employment of Zafar's technique [12] cannot be used with the latest realization of the HEVC standard.

Several recent works deal with HEVC encryption using different approaches. In [13], the author encrypt three selected bitstream elements, namely intra prediction mode difference, motion vector difference sign, and residual sign. In [14], the residue data, intra-prediction modes, inter-prediction modes and motion vectors are key elements that are selected to encrypt to keep the security. Similar works can also be found in the works [15-17].

In this work, we present an original technique for selective encryption allowing the protection of HEVC video, and permitting to preserve the format compliance of HEVC bitstream.

III. THE PROPOSED APPROACH

Among the five syntax elements used for representing each 4x4 sub-block coefficients of the transform unit, only coeff_sign_flag and coeff_abs_level_remaining suffixes can be used to form the encryption space. Exp-Golomb code and Golomb–Rice code are utilized for binarizing coeff_abs_level_remaining, when the first one is employed to encode less frequent symbols with an order equal to cRiceParam+1, and the Golomb–Rice code is widely used to encode symbols having high probability of occurrence and is the most demanded by HEVC encoder for increasing compression ratio. The Golomb-Rice's suffix is an optimal code formed by a binary representation with length equal to cRiceParam bins (for example if cRiceParam is equal to 3, the binary representation of suffix is formed by three bins) .When cRiceParam value is greater or equal to 1, any modifications that affect the suffix binary representation through the same fixed length of bins will not alter the format compliance of Golomb–Rice code because here suffix means merely a remainder of division having the same fixed length of bins. Consequently, we defines the encryption space ES by the set of different binary representations of the Golomb–Rice suffixes of coeff_abs_level_remaining having length greater or equal to 1.

In order to preserve the format compliance of HEVC bitstream, it is necessary to avoid modifying DCs coefficients of TUs, since they represent the average of the TUs energy, and it is preferred to obviate the first and the last coefficient of each reverse scan pattern in each sub-block especially when TUs range in size from 8x8 to 32x32.

It is generally preferred to secure HEVC video only in low delay mode employing a particular order for decoding process, since in other modes, the encryption in random access scheme can alter the format compliance as the HEVC decoder choose any part of the frame at random.

The principle aim of the proposed technique is to keep the format compliance and the bit rate of the encrypted bitstream without any alterations. The decoded frames of the encrypted bitstream must have high visual degradation compared to the original plain frames. The protected parts chosen in proposed selective encryption technique are defined by the set of elements belonging to the encryption space ES.

A. Preparing of plaintext of coefficients levels

For each intra predicted slice I_SLICE, if a transform unit TU is divided into N sub-block named sub-$block_1$,sub-$block_2$,…, sub-$block_N$ consecutively, then only sub-$block_1$ can contain DC coefficient. In addition, if the reverse scanning order in each sub-block starts with a significant coefficient noted $Coef_1$ and proceeds to another significant coefficient noted $Coef_2$ ($Coef_2$ can be DC of TU in sub-$block_1$), then we can encrypt all coefficients of each sub-block except the two coefficients $Coef_1$ and $Coef_2$. Selected coefficients will be denoted further by ACEs.

Fig. 3 shows an example of a TU_{8x8} decomposed into 4 sub-block. In this example, we suppose that all coefficients are non zeros, so we can encrypt all coefficients except those colored in white. The binary plaintext noted plaintext1 is constructed by concatenating all bins of coeff_abs_level_remaining suffixes of the ACEs belonging to ES. This is done by appending into a plaintext P_1 all cRiceParam bins of the binary

representation of each Golomb–Rice suffix found when cRiceParam value is greater or equal to 1.

The number of Golomb–Rice suffixes of coeff_abs_level_remaining syntax elements qualified to be encrypted varies from one 4x4 sub-block to another, and there is at least 14 elements per sub-block. If a sub-block contains L Golomb–Rice suffixes, and if we define L_{MAX} as a number non zero less than L, then any change affects the first L_{MAX} Golomb–Rice suffixes in inverse scan order will modify the whole sub-block entirely. For such reason, we optimized the plaintext P_1 by choosing the bins belonging to the first L_{MAX} Golomb–Rice suffixes found for each 4x4 sub-block, when the range of L_{MAX} ranges from 1 to 14 and defined as a secret parameter.

B. Encryption and decryption procedure

For all intra predicted slice (I_SLICE), we encrypted its corresponding constructed plaintext P_1 in a CBC mode [18]. First we divided the plaintext into a sequence of n+1 consecutives blocks (X_1, X_2,...., X_n, Y) where $n \geq 0$. The length of each block X_i is fixed to 128bit, whereas the length of remainder plaintext Y can be less than 128. A random initial vector $IV \in \{0,1\}^{128}$ is then chosen at random, and the first ciphered block C_1 is generated by applying the AES cipher to $X_1 \oplus IV$ (denotes or-exclusive bit to bit operation) using a secret key K_1 with having a length of 128bits. For i=2...n, each remaining ciphered block C_i is obtained by encrypting $C_{i-1} \oplus X_i$ using AES and the key K_1. The latest block cipher C_{n+1} is specially computed by $C_{n+1}=Y \oplus AES(K_2,IV)$, when K_2 is a secret key on 128 bits ,that it different from K_1 to keep high privacy of the CBC-AES mode. Finally, the ciphertext is obtained by concatenating the blocks C_1, C_2,...., C_n, C_{n+1}.

After encrypting the blocks, selected suffix bin of the plaintext are replaced by correspondent bin in the ciphertext before the BAC compression. The decryption can be performed after BAC decompression by the deciphering suffixes bins of the decoded ciphertext of the. Each ciphertext block is then replaced by its corresponding decrypted plaintext bins.

Since the proposed scheme encipher only Golomb-Rice suffixes, the format of the encoded video sequence is preserved. In addition, since encryption using AES-CBC mode preserves the size of the plaintext, the scheme also preserves the size of encrypted video frames. In the next section, several experiments and performances evaluations are performed on the proposed scheme, with comparisons to some existing video encryption schemes.

IV. EXPERIMENTS AND OBTAINED RESULTS

The proposed scheme is implemented and tested using the HEVC reference software HM v10.0 [19]. In all experimental tests, encryption/decryption are performed simultaneously with encoding/decoding by embedding corresponding modules into the reference software. Simulation results described in this section were processed on benchmark video sequences of different sizes including WQVGA(416×240), WVGA(832×480), SD(1280×720), HD(1920×1080), UltraHD(2560×1600) and 4K UHD(3840×2160) illustrated in Table II with corresponding frame-rates. Table III shows the common encoding parameters used in all tested modes of the reference software (low delay and random access).

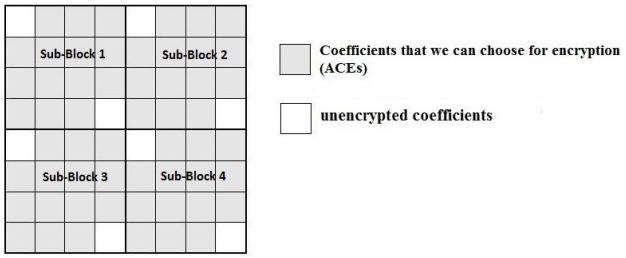

Fig. 3. Transform unit T_{8x8} with 4 sub-block: gray cases mean selected coefficients for encryption.

TABLE.II.
BENCHMARK VIDEO SEQUENCES USED TO SIMULATE THE PROPOSED SCHEME

Video sequence	Resolution	Frame-rate
BasketballPass	416×240	50
BasketballDrill	832×480	50
Johnny	1280×720	60
BasketballDrive	1920×1080	50
Traffic	2560×1600	30
YachtRide	3840×2160	120

TABLE.III.
THE SET OF ENCODER PARAMETERS USING DURING EXPERIMENTS WITH CORRESPONDING VALUES

Parameter	Description	Used value
MaxCUWidth	Maximum coding width in pixel	64
MaxCUHeight	Maximum coding height in pixel	64
MaxPartitionDepth	Maximum coding unit depth	4
IntraPeriod	Intra frame period	8
GOPSize	Size of the GOP's structure	8
InputBitDepth	8 bit per pixel	8

A. Parameters evaluation

The first experiment was performed in low delay mode, where each GOP is composed from an intra frame (I frame) followed by 7 predicted frame (P frame). Fig. 4 shows the first frame of the original benchmark sequences without encryption, and Fig. 5 shows the decoded frames at a quantization parameter QP=18 after encryption with secret parameter L_{MAX}=14. It is apparent that the commercial values of decoded frames are fully destroyed with high visual degradation, and perceptual content is completely disguised. Fig. 6 shows the 9th, 11th, and 14th original frames of BasketballDrill sequence encoded and encrypted at QP=24 with secret parameter L_{MAX}=14, with the corresponding decoding results. Only the 9th frame is encrypted because it is and intra-frame whereas 11th and 14th frames are P frames. It is clear from illustrated results that the visual protection of intra

intra frame propagates to its following P frames.

Fig. 4. The first original frame of: (a) BasketballPass,(b) BasketballDrill,(c) Johnny,(d) BasketballDrive,(e) Traffic and (f) YachtRide

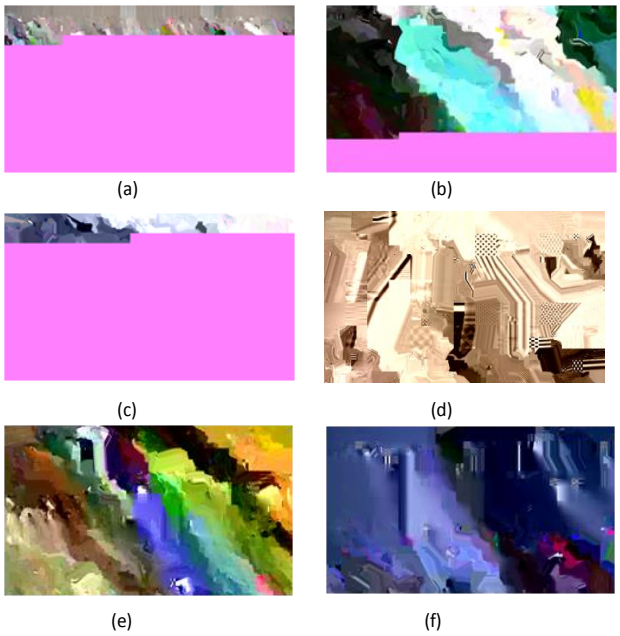

Fig.5. First decoded frame without decryption at QP=18 and L_{MAX}=14 : (a) BasketballPass, (b) BasketballDrill, (c) Johnny, (d)BasketballDrive, (e)Traffic and (f)YachtRide.

In a second experiment, we encrypted and encoded the first frame of BasketballDrill sequence at QP=24 while testing different values of L_{MAX} . Fig. 7 shows original frame used for encryption, with corresponding decoded frames (without decryption) when the secret encryption parameter L_{MAX} take the values 4, 8, and 12 respectively. Decoding results changes according to L_{MAX} , and we note that sufficient visual degradation is obtained for L_{MAX}=4, and all illustrated results prove that decoded frames don't share any outline objects with original ones.

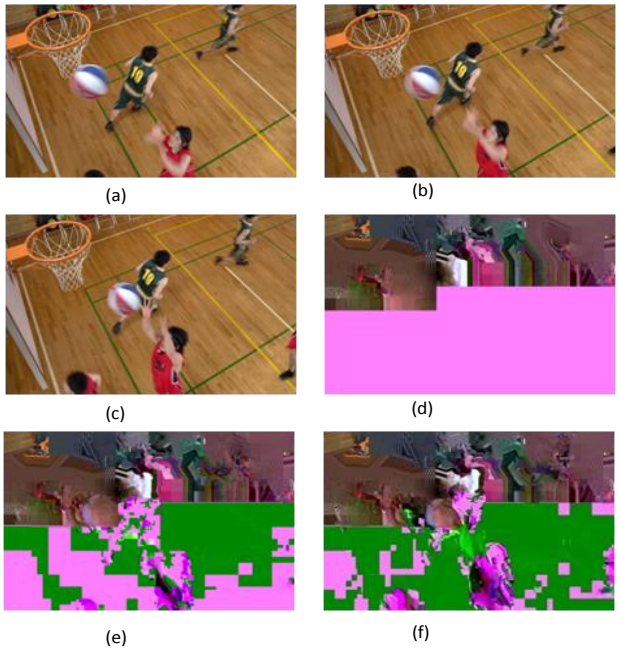

Fig. 6. Original and decoded frames without decryption at QP=24 and L_{MAX}=14 for the BasketballDrill sequence: (a),(b)and (c) are the 9th, 11th,and 14th original frames respectively, and (d),(e),and (f) are the 9th, 11th,and 14th encrypted frames decoded as I,P,P respectively.

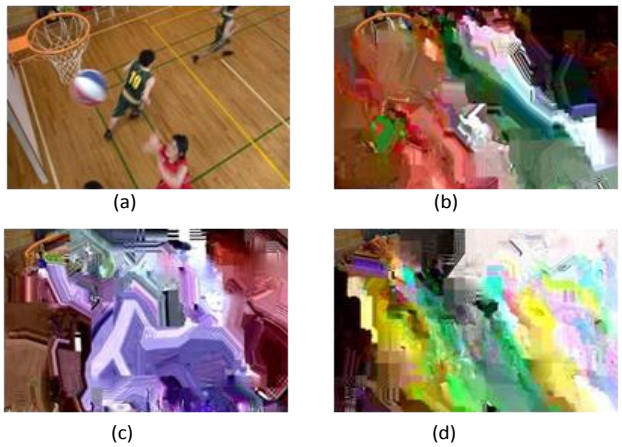

Fig.7. Original and encrypted frames decoded at QP=24: (a) original frame,(b)encrypted frame with L_{MAX}=4, (c) encrypted frame with L_{MAX}=8 and (d) encrypted frame with L_{MAX}=12.

B. Encryption space evaluation

Since the proposed technique is a selective encryption one, it is necessary to evaluate corresponding encryption space defined as the percentage of selected bits chosen for encryption from HEVC bitstream. We encrypted the 10 first frames from each benchmark sequence, with a secret

parameter L_{MAX}=14 in a two modes: firstly, in a low delay mode with each GOP composed by ones intra frame followed by 7 P-frames, and secondly in a random access mode where each GOP is composed by an intra frame followed by 7 B-frame. Table IV shows obtained results for the two tested mode. It is clear that encryption spaces for each sequence in all tested modes are very close to each other. The space's size of obtained results is less than 10% meaning that at least 10% bits of the bitstream provides sufficient high visual degradation.

The influence of QP and L_{MAX} parameters on encryption space variation is evaluated by encrypting the 10 first frame of the BasketballPass sequence. Obtained results show that encryption space increases proportionally for with L_{MAX} increasing values as shown in Table V. In contrast, the size of encryption space decreases with QP values increasing as is depicted in Table VI. Note that a reduced encryption space is obtained due to the encryption efficiency of HEVC compression.

When the proposed scheme is based upon the encryption of Golomb-Raise suffixes, it is important to outline that they have an important statistical frequency in HEVC encoded sequences. A statistical evaluation of codes employed to binarize the coeff_abs_level_remaining suffix is performed in order to evaluate the encryption space size. Table VII shows that the frequency of Golomb–Rice suffix is approximately 8 times higher than Exp-Golomb suffix for all tested sequences, which proves that the important weight of this selected code for encryption.

C. Reconstructed sequence's quality

In order to evaluate the quality of reconstructed video sequences, the simplest and most widely quality used metric is the peak signal-to-noise ratio (PSNR) expressed in decibel (dB). Table VIII lists the average PSNR for the intra decoded frames from all tested sequences encoded at QP=18 with L_{MAX}=14 in a low delay mode when using original HEVC without encryption (Original), and using the proposed selective encryption (Encrypted).

The most pertinent component that incorporates the most meaningful information is luminance component Y. Hence, it is apparent that for all tested sequences, PSNR values of luminance are less than 13dB, signifying that highest visual degradation is achieved for all tested sequences. The PSNR values of the remaining components U and V are reduced to roughly half of their original values.

The structural similarity index (SSIM) is another perceptual metric used to evaluate the quality of the proposed approach. This metric uses only luminance component for calculating (since human visual system is more sensitive to luminance than chrominance components), and it additionally evaluate the structural distortion between two frames. Table IX illustrates PSNR of luminance and the SSIM of decoded frames from BasketballPass sequence without decryption encoded at different QP values of 16, 18, 22, and 26 in a low delay mode. The results shows that for every QP value, a high visual degradation is obtained, when the PSNR values confirms a full distortion obtained using the proposed selective encryption since all values are less than 13dB. Additionally, all SSIM obtained values are less than 0.6 signifying that no visual structural correlation can be found between original and encrypted frames. Hence, the proposed selective encryption algorithm can be considered as a good encryption system with good confidentiality according to the criterions cited in [20].

A similar quality evaluation is performed with respect to the parameter L_{MAX} using several values equal to 3,4,5,6,10 and 14 encoded at QP=18 in a low delay mode. It is clear from corresponding results illustrated in Table IX that PSNR of luminance and SSIM are inversely proportional to the values of L_{MAX}, starting at 15.65dB of PSNR values, and 0.264 for SSIM ones. Best results can be achieved when encrypting the 14 possible coeff_abs_level_remaining suffixes.

D. Run-time and performances evaluation

Experimental simulations were performed on an Intel 2.3GHz Dual-Core T4500 processor with 3 Gb of memory. Table X shows timing results in millisecond (Ms) of encoding process for the first frame with and without encryption. Encoding is performed at QP=18 whereas the encryption is done using L_{MAX}=14. The difference between encoding and encryption time is negligible, and is estimated roughly as the processing time of extraction, encryption, and replacement of the plaintext before the BAC step. Encryption time can be further optimized by defining a novel implementation of HEVC encoder appropriate to the proposed algorithm.

TABLE.IV.
ENCRYPTION SPACE (IN PERCENTAGE) FOR BENCHMARKED SEQUENCES ENCRYPTED IN LOW DELAY AND RANDOM ACCESS MODE AT QP=18 WITH L_{MAX}=14

Sequences	Low delay space (%)	Random Access space (%)
BasketballPass	9.90	9.06
BasketballDrill	6.90	6.42
Johnny	7.81	6.43
BasketballDrive	3,41	2,28
Traffic	4,01	3,58
YachtRide	5,05	4,78

E. Encryption key space and plaintext's sensitivity

The proposed selective encryption algorithm uses 128 bits for the sub-key K_1, 128 bits for the sub-key K_2, and 16 to represent the value of the parameter L_{MAX} that is ranging from 1 to 14. The key space then contains $2^{128+128+4}$ possible key. Therefore, we consider that the key space is sufficiently large to permit robustness against exhaustive key search.

In order to evaluate sensitivity of the approach to plaintext variations, we encrypted the first frame of the "Johnny" sequence at QP=18 using L_{MAX}=14, K_1= A23412841234BFFF and K_2=5E198FE4128825AF then we encode in a low delay mode. The plaintext recuperated before encryption is submitted to a bit change in different places (begin, middle, and end) then encrypted again. Fig. 8 shows that for every bit change in the plaintext, decoded frame keeps a high degradation of visual content due to the randomness of

TABLE.V.
ENCRYPTION SPACE VARIATION OF FIRST ENCRYPTED FRAME FORM BASKETBALLPASS SEQUENCE ACCORDING TO QP VALUES

QP value	Encryption space (%)
18	9,90
20	9,12
24	8,22
30	4,10

TABLE.VI.
ENCRYPTION SPACE VARIATION OF FIRST ENCRYPTED FRAME OF BASKETBALLPASS SEQUENCE ACCORDING TO L_{MAX} PARAMETER'S VALUES

L_{max} value	Encryption space (%)
4	2,14
6	4,12
8	6,16
12	8,01

ciphertext resulting by encrypting all plaintexts in CBC-AES mode, which proves the robustness of the proposed algorithm.

F. Comparative study

TABLE.VII.
EXPERIMENTAL FREQUENCIES OF GOLOMB–RICE CODE AND EXP-GOLOMB CODE EMPLOYED FOR COEFF_ABS_LEVEL_REMAINING BINARIZATION IN THE HEVC ENCODING STANDARD

Sequences	Golomb–Rice code (%)	Exp-Golomb code (%)
BasketballPass	87,84	12,15
BasketballDrill	87,42	12,57
Johnny	87,12	12,87
BasketballDrive	87,41	12,58
Traffic	87,36	12,63
YachtRide	87,05	12,94

A comparative study is performed with some recent works on selective encryption of last video standards (H264/AVC, HEVC) proposed after 2010. A set of different criterions is used to evaluate and compare tested encryption algorithms. In Table 11, algorithms chosen for comparison are conform to the format of encrypted video standard, but they differ in several aspects like maintenance of compression rate, encryption domain, context modeling, encryption algorithm, and compression independence. While the scheme proposed in [12] selects coeff_abs_level_remaining suffixes and signs of transform coefficients to secure HEVC videos, and hence modify the context modeling used for BAC compression, the proposed scheme reduces the encryption space by encrypting only coeff_abs_level_remaining suffixes binarized by

TABLE.VIII.
PSNR OF DECODED FRAMES WITHOUT DECRYPTION (ORIGINAL) ,AND USING PROPOSED SELECTIVE ENCRYPTION (ENCRYPTED) IN LOW DELAY MODE AT QP=18 WITH L_{MAX}=14

	PSNR Y (dB)		PSNR U (dB)		PSNR V(dB)	
Sequence	Orig.	Enc.	Orig.	Enc.	Orig.	Enc.
BasketballPass	45.77	8.64	47.23	26.05	47.46	22.21
BasketballDrill	45.19	11.9	45.99	16.80	47.10	18.47
Johnny	46.26	9.33	49.44	21.09	49.87	23.29
BasketballDrive	46.45	9.93	46.35	9.21	48.16	13.76
Traffic	46.25	6.89	45.52	14.66	47.04	15.45
YachtRide	47.59	8.80	48.92	12.11	48.04	10.27

TABLE.IX.
VARIATION OF PSNR AND SSIM FOR FIRST DECODE FRAME OF BASKETBALLPASS SEQUENCE ACCORDING TO L_{MAX} AND QP VALUE

L_{max} Evaluation			QP Evaluation		
Parameter Value	PSNR Y (dB)	SSIM	Parameter Value	PSNR Y (dB)	SSIM
3	15.65	0.264	16	10.89	0.222
4	12.33	0.287	18	8.64	0.020
5	13.03	0.286	22	9.19	0.164
10	11.77	0.148	26	11.31	0.130
14	8.64	0.020	28	10.12	0.110

TABLE.X.
ENCRYPTION TIME (MS) ESTIMATED FOR FIRST FRAME ENCODING OF EACH BENCHMARK SEQUENCE WITH AND WITHOUT ENCRYPTION (QP=18)

Sequences	Encoding Time (Ms)	
	Without Encryption	With Encryption
BasketballPass	22.94	23.08
BasketballDrill	90.81	90.90
Johnny	162.62	162.69
BasketballDrive	444.15	444.23
Traffic	845.12	845.94
YachtRide	1520.10	1521.01

Golomb–Rice code without any change that affect the context modeling of BAC compression. This result in optimized encryption and enhanced format complacence with extremely effective encryption performances.

V. CONCLUSIONS

Emerging high Efficiency Video Coding standard HEVC presents new compression concepts such as Golomb–Rice codes that can be considered as good support to ensure security of selective encryption. We have presented in this paper a novel scheme of selective encryption based on the protection of Golomb–Rice suffixes (*coeff_abs_level_remaining*) using AES-CBC enciphering algorithm. We selected only suffixes of sub-blocks belonging to intra slice (I_SLICE), and the encryption is performed before binary arithmetic coding (BAC).

According to obtained results, we show that visual content of decoded frames from encrypted bitstream is very low for all video resolutions, implying that high visual degradation is attained using the proposed scheme. Decoding without errors confirms the format compliance of the encrypted bitstream, while the proposed approach permitted to obtain a reduced

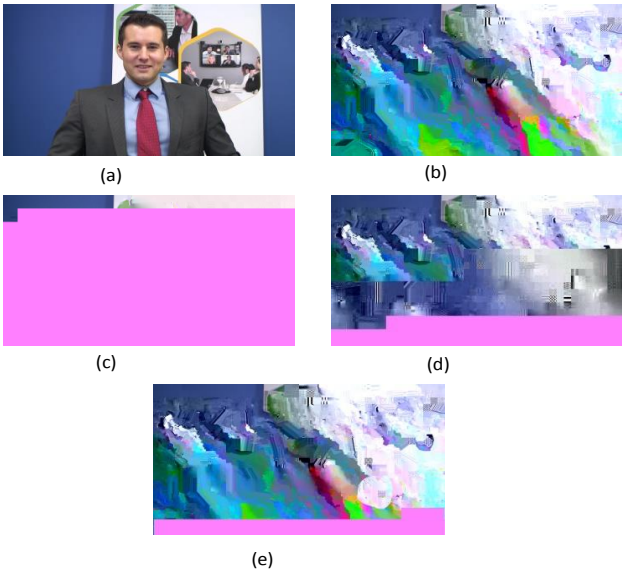

Fig.8. Decoded frame results for plaintext sensitivity evaluation: (a)original frame, (b) encrypted frame, (c)bit changed at beginning of plaintext, (d)bit changed at middle of plaintext, and (e) bit changed at end of plaintext.

encryption space formed by a minimal set of encrypted bits. We also compare the processing time of encoding with and without encryption, and we show that encryption overhead is negligible difference with respect to encoding one, implying that the scheme is suitable for real time applications.

To measure the distortion between original and encrypted frames, we utilized PSNR and SSIM metrics. Experimental results justify the high visual degradation obtained for all QP and L_{MAX} used values. Furthermore, the plaintext sensitivity is benchmarked against bit change in several locations of the encrypted sequence, and decoded results show that the proposed scheme provides high sensitivity.

We rewind that the scheme depends on compression quality of HEVC. Thus, better results can be achieved for low QP values (less than 24) since the size of the encryption space is inversely proportional to QP values. We note that the proposed technique is one of the first selective encryption techniques characterized by format compliance and optimized encryption space for the HEVC encoding standard.

TABLE.XI.
COMPARATIVE ANALYSIS BETWEEN OUR APPROACHES AND PRIOR SELECTIVE ENCRYPTION ALGORITHMS

Encryption algorithm	Compression ratio maintained	Encryption domain	Context modeling	Encryption algorithm	Compression independence
Wei et al (H.264/AVC)[21]	No	NALUs	No	RC4 + XOR	Yes
O.-Y. Lui (H.264/AVC) [9]	Yes	DCT coefficient (Sign of T1, Non-zero level)	No	Chaos + XOR	Yes
Shahid et al. (H.264/AVC) [8]	Yes	DCT coefficient (Sign of T1, Non-zero level)	No	AES+XOR	Yes
Shahid et al. (HEVC) [12]	Yes	coeff_abs_level_remaining suffix + signs	YES	AES+XOR	Yes
Wang et al. (H.264/AVC) [14]	Yes	DCT coefficient (Macro-blocks)	No	Permutation	No
Proposed algorithm	Yes	coeff_abs_level_remaining suffix	No	AES-CBC	Yes

V. REFERENCES

[1] Sullivan, G. J., Ohm, J., Han, W. J., & Wiegand, T. (2012). Overview of the high efficiency video coding (HEVC) standard. Circuits and Systems for Video Technology, IEEE Transactions on, 22(12), 1649-1668.

[2] Wiegand, T., Sullivan, G. J., Bjontegaard, G., & Luthra, A. (2003). Overview of the H. 264/AVC video coding standard. Circuits and Systems for Video Technology, IEEE Transactions on, 13(7), 560-576.

[3] ISO/IEC JTC1/SC29/WG11 Vision, Application, and Requirements for High Performance Video Coding (HVC), MPEG Document, N11096, Kyoto, JP (Jan. 2010).

[4] Sze, V., & Budagavi, M. (2012). High throughput CABAC entropy coding in HEVC. Circuits and Systems for Video Technology, IEEE Transactions on,22(12), 1778-1791.

[5] Marpe, D., Schwarz, H., & Wiegand, T. (2003). Context-based adaptive binary arithmetic coding in the H. 264/AVC video compression standard. Circuits and Systems for Video Technology, IEEE Transactions on, 13(7), 620-636.

[6] High efficiency video coding, ITU-TRec.H.265 and ISO/IEC 23008-2 (MPEG-H, Part 2), Apr.(2013), version 1.

[7] Sole, J., Joshi, R., Nguyen, N., Ji, T., Karczewicz, M., Clare, G., ... & Duenas, A. (2012). Transform coefficient coding in HEVC. Circuits and Systems for Video Technology, IEEE Transactions on, 22(12), 1765-1777.

[8] Shahid, Z., Chaumont, M., & Puech, W. (2011). Fast Protection of H. 264/AVC by Selective Encryption of CAVLC and CABAC for I and P frames. Circuits and Systems for Video Technology, IEEE Transactions on, 21(5), 565-576.

[9] Lui, O. Y., & Wong, K. W. (2013). Chaos-based selective encryption for H. 264/AVC. Journal of Systems and Software, 86(12), 3183-3192.

[10] Park, S. W., & Shin, S. U. (2008, September). Efficient selective encryption scheme for the H. 264/scalable video coding (SVC). In Networked Computing and Advanced Information Management, 2008. NCM'08. Fourth International Conference on (Vol. 1, pp. 371-376). IEEE.

[11] Wang, X., Zheng, N., & Tian, L. (2010). Hash key-based video encryption scheme for H. 264/AVC. Signal Processing: Image Communication, 25(6), 427-437.

[12] Shahid, Z. Puech, W., Visual Protection of HEVC Video by Selective Encryption of CABAC Binstrings, Multimedia, IEEE Transactions on , vol.16, no.1, pp.24,36, Jan. 2014
[13] Van Wallendael, G., Boho, A., De Cock, J., Munteanu, A., & Van de Walle, R. (2013, January). Encryption for High Efficiency Video Coding with video adaptation capabilities. In Consumer Electronics (ICCE), 2013 IEEE International Conference on (pp. 31-32). IEEE.
[14] Wang, Q., & Wang, X. (2014, May). A new selective video encryption algorithm for the H. 264 standard. In Progress in Informatics and Computing (PIC), 2014 International Conference on (pp. 275-279). IEEE.
[15] Zhang, X., & Qiu, B. (2014). Fast Mode Decision and Encryption Policy in H. 264/AVC Frame-skipping Transcoding. Journal of Computers, 9(5), 1201-1208.
[16] Nithya, B., & Radharani, S. (2014). Scanned Document Compression Using High Efficiency Video Coding (HEVC) Standard.International Journal, 3(11).
[17] Zhou, J., Liu, X., Au, O. C., & Tang, Y. Y. (2014). Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation.
[18] Dherbecourt, Y. M., Herodin, J. M., & Vidrascu, A. (1996). U.S. Patent No. 5,583,940. Washington, DC: U.S. Patent and Trademark Office.
[19] SHM reference software: https://hevc.hhi.fraunhofer.-de/svn/svn_SHVCSoftware/
[20] Dubois, L., Puech, W., Blanc-Talon, J., 2012. Reduced selective encryption of intra and inter frames of H.264/AVC using psychovisual metrics. In: 19th IEEE Inter-national Conference on Image Processing, pp. 2641–2644.
[21] Wei, Z., Wu, Y., Ding, X., & Deng, R. H. (2012). A scalable and format-compliant encryption scheme for H. 264/SVC bitstreams. Signal Processing: Image Communication, 27(9), 1011-1024.

Mokhtar Ouamri received his engineer degree from the University of Sciences and Technologies of Oran (USTO), Oran, Algeria, in 2007, and the M.S. degree from the University of Sciences and Technologies of Oran (USTO), Oran, Algeria, in 2010, both in computer science. Since December 2011, he is PhD candidate in computer science, at Djillali Liabes University (UDL), Algeria. He is currently Assistant Professor at University of Ibn-Khaldun, Tiaret, Algeria. His research interests are in the fields of multimedia compression/security, image/video processing and analysis, video surveillance (detection, tracking, event detection, and storage) ,and multimedia communication.

K.M. Faraoun was born in Sidi Bel abbes, Algeria, in February 23, 1978. He received his master's degree in computer science at the computer science department of Djilali Liabbes University- Sidi-Bel-abbes – Algeria in 2002, his Ph.D degree in computer science, in 2006, and his HDR degree in computer science and intelligent systems, in 2009 From UDL-University. His current research areas include computer security systems; cryptography; genetic algorithms; cellular automata; evolutionary programming and information theory. Dr. Faraoun is currently a Full professor and a teacher at the computer sciences department of UDL-University, he teaches Information Theory and Cryptography. He has published several papers in many international journals.