Jurnal Teknologi

Full Paper

# LIGHTWEIGHT ENCRYPTION FOR HIGH EFFICIENCY VIDEO CODING (HEVC)

Mohammed A. Saleh, Nooritawati Md. Tahir, Habibah Hashim*

Faculty of Electrical Engineering, Universiti Teknologi MARA, 4045 UiTM, Shah Alam, Selangor, Malaysia

**Article history**
Received
*15 February 2016*
Received in revised form
*10 June 2017*
Accepted
*10 August 2017*

*Corresponding author
habib350@salam.uitm.edu.my

## Abstract

Video data are being compressed and distributed using one of several coding standards, among which the most recent and popular is the High Efficiency Video Coding (HEVC) standard. The threatening growth of security attacks, on the other hand, has brought security and privacy concerns to the attention of governments and people as well. In the absence of a reliable security system, shared multimedia data used on the public networks such as the internet will continue to be exposed to different types of attacks, making end-to-end encryption for video data a necessity to protect their sensitive information. Therefore, providing a reliable video security technique that complies to and fulfills the requirements of HEVC is pertinent. In this paper, a fast selective encryption approach is developed to provide protection for video bitstreams of HEVC, which can be used in real-time video applications, with low computational overhead and maintaining the standard's video bit rate. This approach employs the popular Advance Encryption Standard (AES) algorithm to encrypt selected elements in the horizontal intra prediction modes. Experimental evaluations confirm the provision of adequate security level of video information, with no bitrate increase, no increase in computational delay and no additional impact on the compression performance when compared to non-secure techniques, while also achieving a satisfactory trade-off between the encryption reliability, flexibility, and computational complexity. The security level of this method was found to be strongly secure against plaintext and brute force attacks.

*Keywords*: HEVC standard, real-time encryption, selective encryption, AES, video stream security

## Abstrak

Data video dimampat dan disebar menerusi beberapa piawai pengekodan antara yang terkini dan lebih popular adalah piawai High Efficiency Video Coding (HEVC). Walau bagimanapun, ancaman dari pertumbuhan angka cubaan menceroboh telah meningkatkan kebimbangan pihak berkuasa mahupun orang awam terhadap isu keselamatan dan privasi. Selagi sebuah sistem keselamatan yang bolehharap tidak diwujudkan, data multimedia pada rangkaian awam seperti internet akan terus terdedah kepada berbagai jenis cubaan menceroboh. Hal ini memerlukan data video dilindungi melalui proses penyulitan bagi memastikan maklumat sensitif yang terkandung di dalamnya terselamat. Oleh itu, penyediaan suatu teknik keselamatan video yang bolehharap yang dapat mematuhi dan memenuhi piawai HEVC menjadi satu keperluan relevan. Artikel ini menerangkan pembangunan suatu kaedah penyulitan selektif pantas bagi memberi perlindungan kepada deretan bit video HEVC, yang setreusnya dapat digunapakai ke atas aplikasi masa nyata dengan kos komputasi yang rendah, pada masa yang sama menepati keperluan kdara bit video yang ditetapkan oleh HEVC. Pendekatan tersebut menggunakan algoritma terkenal iaitu Advanced Encryption Standard (AES) untuk menyulitkan elemen-elemen terpilih dalam mod intra ramalan mendatar. Penilaian eksperimental mengesahkan bahawa tahap keselamatan bagi maklumat ideo yang diperoleh daripada pendekatan ini adalah memadai, di mana didapati tiada peningkatan dalam kadar bit, tiada penambahan dalam lengah komputasi dan tiada impak tambahan terhadap prestasi pemampatan apabila disbanding dengan pendekatan tanpa sekuriti. Pendekatan ini juga telah menghasilkan keseimbangan yang memuaskan di antara kebolehharapan penyulitan, fleksibiliti dan kerumitan komputasi. Tahap sekuriti pendekatan ini dipercayai dapat mematahkan cubaan ceroboh "known-plaintext" dan "brute force".

*Kata kunci*: piawai HEVC , penyulitan data masa nyata, penyulitan data terpilih, AES, sekuriti deretan ideo

## 1.0 INTRODUCTION

In tandem with the rapid widespread use of the Internet and telecommunication technologies, information sharing is gaining much importance, accompanied by increasing awareness of its security and privacy implications. Recently, end users became conscious about their sensitive information that can be shared through the enormous number of Internet social networks, especially after observing the increasing trends of security attacks [1], [2]. On the other hand, the number of video sharing applications is rapidly growing due to the revolution of the smart devices connected over the internet. According to Cisco, by 2019 mobile video sharing will occupy more than 69 percent of smart mobile data traffic, where the total global percentage of the smart mobile traffic will be 97 percent [3]. Since the volume of video data is high especially in high definition video, video compression standards have been developed in order to maintain the bandwidth requirements by minimizing the size of the video information [4]. Standardized techniques started as the analog videophone system in 1960 [5], and evolved to the High Efficiency Video Coding (HEVC) in 2013 [6].

Full data encryption of the multimedia content cannot endure the bandwidth limitation, computational time and memory constraints, because the video data storage is still high even after the compression process. Consequently, different encryption methods have been proposed to secure video data during the process of video compression.

The main goal of this paper is to propose a relevant scheme for encrypting video data, taking into account to save video bit rate and maintain the video quality as well as the computational overhead. Since a new video coding standard, namely HEVC, has been developed lately, new approaches to secure video stream is required [7], [8], [9]. In this paper, a new method to secure video information by selectively encrypting the sensitive syntax elements of HEVC has been proposed, in which intra prediction direction modes are encrypted by Advance Encryption Standard (AES) algorithm.

The paper is organized as follows: In introduction section (1.0), a brief overview of HEVC entropy coding is explained in Subsection 1.1. In Subsection 1.2, the state of the art in HEVC video encryption schemes is presented. The proposed encryption method is then described in Section 2.0. In Sections 3.0, the implementation of the experiment and the results are illustrated. Section 4.0 concludes the paper.

### Entropy Coding of HEVC

In modern video coding standards, the entropy coding technique is used to represent the video data in a small form. Entropy coding utilizes the statistical properties to compress video data, thus it is denoted as a lossless compression scheme [10]. There are two types of the entropy coding: Context Adaptive Variable Length Coding (CAVLC) and Context-Based Adaptive Binary Arithmetic Coding (CABAC) [10]. The CABAC algorithm was firstly used in the H.264/AVC standard, which was developed by the ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG) [11], [12]. The last versions of HEVC include only the CABAC that was firstly introduced in H.264/AVC standard [13].

Using the CABAC, the syntax elements of HEVC are represented as a bit string (codeword). The number of the bits in the codeword is proportional to the syntax element probability occurrence. Thus, syntax elements of high frequent occurrence are represented by short codeword (few bits), while those of the low frequent occurrence are represented by long codeword (more bits) [14], [11].

In HEVC, the CABAC was a part of the first test model HM1.0 (HM is the reference software for HEVC) along with the CAVLC [11] that performs a low complexity entropy coding. Lastly, after the HEVC standardization improvements, the last versions of HMs include only CABAC. The basic design of CABAC combines three main phases, which are binarization, context modelling, and binary arithmetic coding. In binarization phase, the non-binary syntax elements are binarized into binary symbols (bins). In context modelling, the probabilities of each bin (regular coded bin) are estimated based on some specific context. These bins are compressed to bits according to the estimated probability in the binary arithmetic coding stage [10], [13], [15]. The binarization stage is explained here, because the proposed encryption is performed based on this stage.

After the prediction and transformation process, some of the HEVC syntax elements are in non-binary form, for instance the transform coefficients and the motion vector differences. Hence, the binarization of those syntax elements are required prior to the context modeling and arithmetic coding process [16].

The main idea of binarization methods is how to represent the non-binary value *N* efficiently in less number of bits [17]. The binarization process in HEVC is performed by five different methods: Unary, Truncated Unary (TrU), *kth* order Exp-Golomb code (EGk), *kth* order Truncated Rice (TRk) and Fixed Length (FL). Each of those binarizations is briefly explained below.

(a) Unary coding represents the non-binary value *N* into *N + 1* bin string length, where the first *N* bins are 1s and the last bin is 0. In the decoder part, searching for a zero (0) is used to determine the end of a syntax element.

(b) Truncated Unary (TrU) represents the non-binary value N less than cMax by the Unary binarization, while if N value equals cMax the bin string is a bit string of length cMax with all bins being equal to 1 [18].

(c) kth order Truncated Rice (TRk) is a concatenation of a prefix and a suffix number, which depends on the k parameter. If the value of N>>k, the prefix is taken as a truncated unary string, where the largest possible value of the prefix is cMax. The suffix is taken as a fixed length binary of the least significant bins of N. However, the number of the least significant bins is specified by k. For k = 0, the binarization process is treated as the truncated unary binarization.

(d) Kth order Exp-Golomb (EGk) code is used as an efficient way of representing syntax elements using the concatenation of prefix and suffix numbers. The prefix part of the EGk bin string for a given unsigned integer value N is generated by a unary code representation of length L(N) = [Log2(N/2k+1)]. The suffix part of the EGk for the same value N is generated as the binary representation of N+2k(1-2L(N)) using k + L(N) significant bits. Therefore, the overall binarized codeword length of EGK is 2L(N)+(k+1).

(e) Fixed length (FL) code represents the unsigned integer value N with a fixed length bin string codeword of length [log2(cMax + 1)]. In FL, the most significant bins are signaled before the least significant bins [11].

The binarization in the HEVC can be performed according to the type of syntax element. The non-binary syntax element *N* can be binarized with a single or by more than one binarization method; for example, the *cu_qp_delta_abs* syntax is binarized by both the TrU and EG0 methods [11].

### State of the Art in Selective Encryption Methods of HEVC

Since the size of digital video is huge, using full video data encryption consumes enormous amount of processing power and time, making full multimedia data encryption is an unacceptable option, especially in real-time applications. One possible alternative is the selective encryption of a limited number of syntax elements, which can provide a sufficient security level with low computational complexity [19], [20], [21]. Different types of encryption algorithms can be used to perform selective encryption in video data, such as AES and Data Encryption Standard (DES). The state of the art in this topic is very limited and that there are only very few works on the same direction.

Hofbauer *et al*. [7] proposed a scheme on HEVC standard in 2014, based on encrypting the sign bits in the luminance channel only. The limitations of this method are concluded from [22], which states that the encryption of the sign bits cannot provide a high security level.

Shahid and Puech [8] proposed an approach where some of the syntax elements are selected and encrypted using AES Cipher Feedback (CFB) mode. Those syntax elements are: the sign bit of the quantized transform coefficient value, the TRp suffix part, the EG0 suffix, the sign of motion vector difference and the suffix of EG1 code. This method provides a high efficient security, on the other hand, the percentage of the encrypted data is high, which leads to an increase in the computational overhead. In A. Saleh [23] scheme, *abs_mvd_minus2* was selected which only encrypts the moving objects information not all video.

## 2.0 METHODOLOGY

In this section, the new proposed approach to secure HEVC video stream is explained. Since the HEVC standard supports only CABAC in each coding configurations, the encryption is selectively performed on syntax elements of CABAC applying the robust AES standard algorithm. Encryption process of the proposed approach maintains the coding efficiency of HEVC standard by generating a completely compliant video bitstream with low computational power.

### Encryption Process

HEVC combines 35 intra prediction modes, where every prediction unit (PU) has vertical and horizontal angular direction modes. The intra prediction unit (spatial unit) contains frame details, and can be further divided into other prediction units wherein every unit has its prediction direction modes. Since the percentage of horizontal intra prediction modes is low compared to the total video data, and due to its sensitivity, it has been selected to secure the whole video information. Furthermore to maintain format compliance, low delay and bit rate in HEVC, horizontal angular direction modes of intra prediction have been encrypted. This choice gives high secured bitstream and fulfills the low resource device requirements. Among the AES encryption modes (CBC, ECB, CTR, OCB, and CFB), the suitable mode for streaming encryption is CFB [24]. Therefore, we have designed our method to encrypt a stream of syntax elements in the horizontal intra prediction modes by CFB. The encryption process is performed as follows:

$$G_i = Ek(C_{i-1}) \quad (1)$$

$$C_i = P_i \oplus G_i \quad (2)$$

where $G_i$ are the generated key streams, the $\oplus$ symbol is the XOR operator, $P_i$ is the plaintext input to the AES, and $C_i$ is the cipher text (output of the encryption algorithm).

In HEVC, the current prediction unit is predicted from the neighboring units according to the angular intra prediction modes [16]. As shown in Figure 1, 33 angular prediction modes in addition to planar and

DC prediction modes for luma intra prediction are used for all the PU sizes in the vertical and the horizontal directions. Here the syntax elements of intra angular prediction modes in the horizontal direction are selected as plaintext. Figure 2 shows the block diagram for the selection process of encrypted syntax elements.

### Decryption Process

In the decoder end, after the arithmetic decoding process, the original plaintext $P_i$ (resulted from the intra prediction direction modes) is retrieved (decrypted) from the ciphertext $C_i$ using AES and the same encryption key. In other words, the original syntax elements are generated back using the same encryption key *Ek* used to generate the keys stream $G_i$ and XORing it by the ciphertext $C_i$.

## 3.0 RESULTS AND DISCUSSION

Using the HEVC, the proposed scheme has been simulated on a system with the specs described in Table 1. All experiments were performed on several types of benchmark video sequences. Table 2 describes each class of the implemented videos. The experiments have been started by analyzing: the data size and bit rate of encrypted video data compared to original video data; percentage of the selected data to the total encoded data size for the different types of video sequences. All of these analyses are described in Table 3 for quantization parameters 32 to 36 and low delay HEVC coding configuration.

The visual distortion of the encrypted videos using the proposed method is clearly observed in Figure 3 for *BasketBallPass* video sequence.

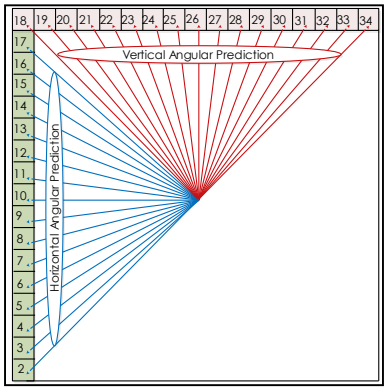


**Figure 1** Angular Intra Prediction Modes in HEVC

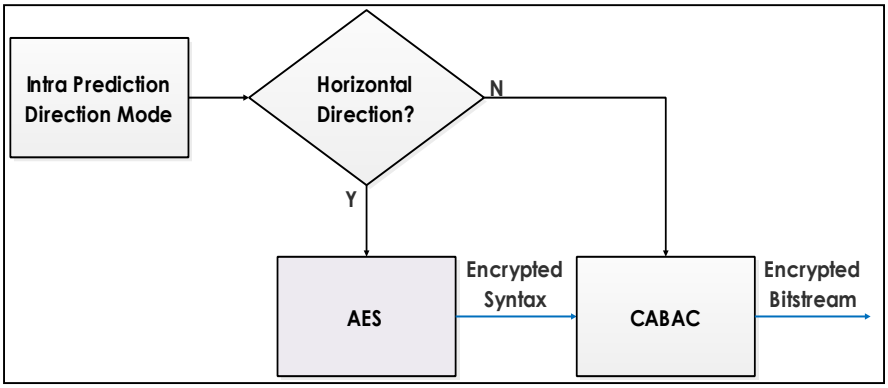


**Figure 2** Selection for the Horizontal Intra Prediction Modes

**Table 1** Specifications of the used PC and HM

| Experimental setup | |
|---|---|
| Processor | Intel(R) core(TM) i5,CPU 3.00GHZ |
| RAM | 8.00GB |
| Number of frames encoded | 100 |
| HEVC Test Model | HM10 |
| Frame Rate | Varied according to the video sequence |
| Coding configurations | Low delay and random access |
| Quantization parameter (QP) | 32 - 36 |

**Table 2** The set of benchmark video sequences used to simulate the video encryption method on HEVC standard

| Class | Sequence | Resolution | Frame Rate |
|---|---|---|---|
| A | Traffic | 2560×1600 | 30 |
| | PeopleOnStreet | 2560×1600 | 30 |
| B | Akiyo | 352×288 | 20 |
| | ParkScene | 1920×1080 | 24 |
| | Kimono | 1920×1080 | 24 |
| B1 | BQterrace | 1920×1080 | 60 |
| C | BasketBallDrill | 832×480 | 50 |
| | BQMall | 832×480 | 60 |
| | PartyScene | 832×480 | 50 |
| | RaceHorseC | 832×480 | 30 |
| D | BasketBallPass | 416×240 | 50 |
| | BQSquare | 416×240 | 60 |

| Class | Sequence | Resolution | Frame Rate |
|---|---|---|---|
| | BlowingBubbles | 416×240 | 50 |
| | RaceHorses | 416×240 | 30 |
| | Vidyo1 | 1280×720 | 60 |
| E | Vidyo3 | 1280×720 | 60 |
| | Vidyo4 | 1280×720 | 60 |

**Table 3** Bit rate and data size for non-encrypted and encrypted video sequences with percentage of encrypted data

| Sequence | Bit rate(kbps) | | Total size | | Encrypted % |
|---|---|---|---|---|---|
| | Original | Encrypted | Original | Encrypted | |
| Traffic | 5019.456 | 5019.456 | 209144 | 209144 | 10.40 |
| PeopleOnStreet | 11112.792 | 11112.792 | 463033 | 463033 | 9.41 |
| ParkScene | 2493.888 | 2493.888 | 129890 | 129890 | 7.9 |
| Kimono | 2111.174 | 2111.174 | 109957 | 109957 | 2.8 |
| BasketBallDrill | 1128.200 | 1128.200 | 28205 | 28205 | 12.67 |
| BQMall | 1738.176 | 1738.176 | 36212 | 36212 | 13.34 |
| PartyScene | 3783.760 | 3783.760 | 94594 | 94594 | 10.99 |
| RaceHorseC | 1647.672 | 1647.672 | 68653 | 68653 | 8.84 |
| BasketBallPass | 343.480 | 343.480 | 8587 | 8587 | 11.82 |
| BQSquare | 828.192 | 828.192 | 17254 | 17254 | 11.05 |
| BlowingBubbles | 573.680 | 573.680 | 14342 | 14342 | 10.95 |
| RaceHorses | 513.144 | 513.144 | 21381 | 21381 | 9.28 |
| Vidyo1 | 1048.368 | 1048.368 | 21841 | 21841 | 14.88 |
| Vidyo3 | 1237.584 | 1237.584 | 25783 | 25783 | 11.33 |
| Vidyo4 | 900.624 | 900.624 | 18763 | 18763 | 13.52 |
| **Average** | | | | | **10.61** |

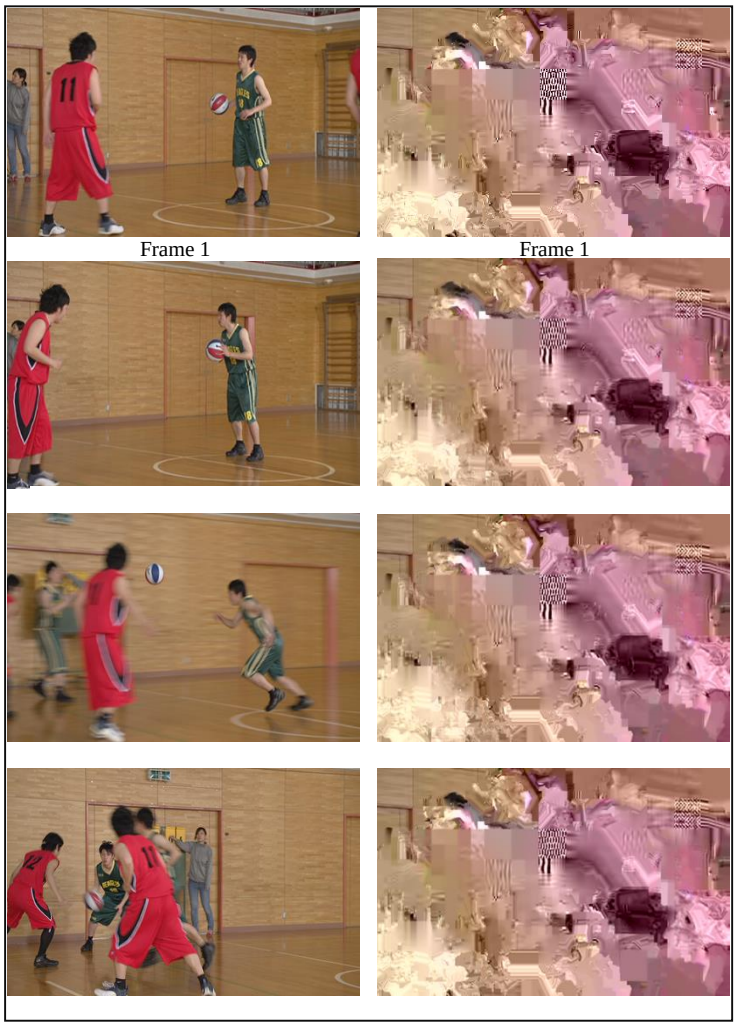


**Figure 3** Encrypted frames from video sequence of *BasketBallPass* using low delay coding configuration

## Video Quality Analysis

The video quality has been tested to evaluate the encryption effects of the proposed method. PSNR [25] and SSIM [26] are common metrics for video quality analysis by comparing the quality of the original video with the encrypted video. Both these metrics have been used for the evaluation on different types of video sequences for the low delay main and random access coding configurations of HEVC. Figure 4 and 5 describe the PSNR and SSIM of the original and encrypted videos using the proposed selective encryption algorithm respectively. As shown in Figure 3 and 6, the effect of the encryption process on the video quality is high, so the proposed approach does distort (and hence to some extent secure) the video visual information.

Furthermore, Table 4 compares the PSNR metric for encrypted and non-encrypted benchmark video sequences on the low delay main HEVC coding configuration. The average values of PSNR(Y), PSNR(U), and PSNR(V) for all original encoded benchmark video sequences are 34.17, 39.33 and 39.72 dB respectively, while the average values of PSNR(Y), PSNR(U), and PSNR(V) for all encrypted benchmark video sequences are 8.06, 13.06 and 12.39 dB respectively. To evaluate the quality difference between original and encrypted benchmark video sequences using SSIM, the results in Table 5 were obtained. In the SSIM analysis, the average of SSIM(Y) of the original video is 0.92dB while the average of SSIM(Y) of encrypted video is 0.28dB. Consequently, it can be concluded that the proposed method is capable of securing the visual information of every class of video sequences adequately. According to the PSNR and SSIM results of different video sequence resolutions, it is clearly that this method is suitable for securing significant visual information for videos in different resolutions.

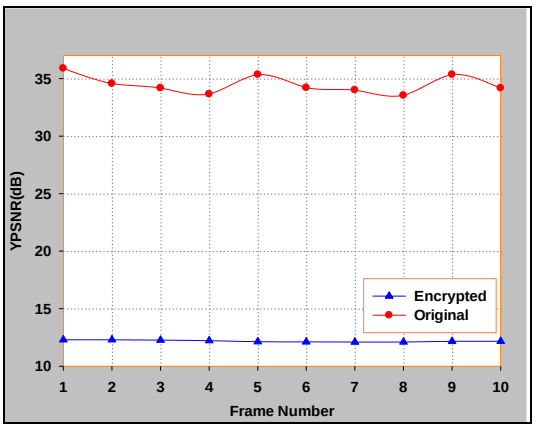


**Figure 4** YPSNR of Original and Encrypted *BasketBallPass* Video sequence

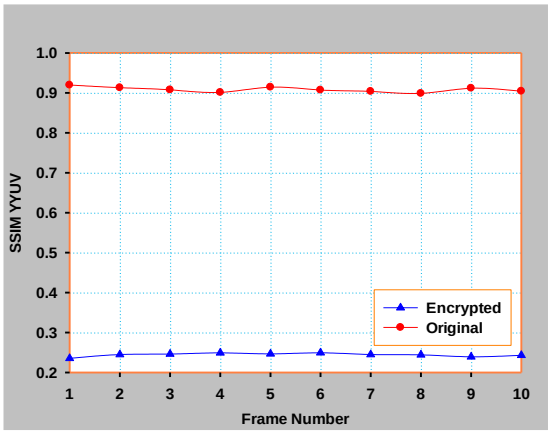


**Figure 5** SSIM of Original and Encrypted *BasketBallPass* Video sequence

**Table 4** PSNR comparison for original and encrypted video sequences

| Sequence | PSNR (Y) dB | | PSNR (U) dB | | PSNR (V) dB | |
|---|---|---|---|---|---|---|
| | Original | Encrypted | Original | Encrypted | Original | Encrypted |
| Traffic | 36.45 | 6.54 | 40.37 | 8.24 | 38.18 | 9.00 |
| PeopleOnStreet | 34.54 | 7.62 | 41.00 | 9.70 | 41.67 | 10.28 |
| ParkScene | 34.75 | 7.42 | 38.56 | 9.92 | 40.09 | 9.56 |
| BasketBallDrill | 34.80 | 8.87 | 38.78 | 12.34 | 39.09 | 8.08 |
| BQMall | 34.17 | 7.35 | 39.05 | 9.77 | 39.91 | 8.83 |
| PartyScene | 31.12 | 9.53 | 36.15 | 12.94 | 36.41 | 12.53 |
| RaceHorseC | 32.41 | 6.42 | 36.78 | 8.70 | 37.98 | 8.58 |
| BasketBallPass | 34.50 | 10.64 | 39.15 | 23.57 | 38.63 | 23.03 |
| BQSquare | 31.59 | 6.99 | 39.64 | 23.53 | 40.08 | 25.99 |
| BlowingBubbles | 33.15 | 8.33 | 38.91 | 13.88 | 41.09 | 10.31 |
| RaceHorses | 32.07 | 10.30 | 36.44 | 19.13 | 38.06 | 17.40 |
| Vidyo1 | 31.92 | 7.96 | 36.42 | 12.74 | 36.79 | 11.88 |
| Vidyo3 | 38.90 | 7.86 | 43.93 | 8.59 | 44.37 | 8.43 |
| Vidyo4 | 38.06 | 7.04 | 45.49 | 9.79 | 43.75 | 9.50 |
| Average | 34.17 | 8.06 | 39.33 | 13.06 | 39.72 | 12.39 |

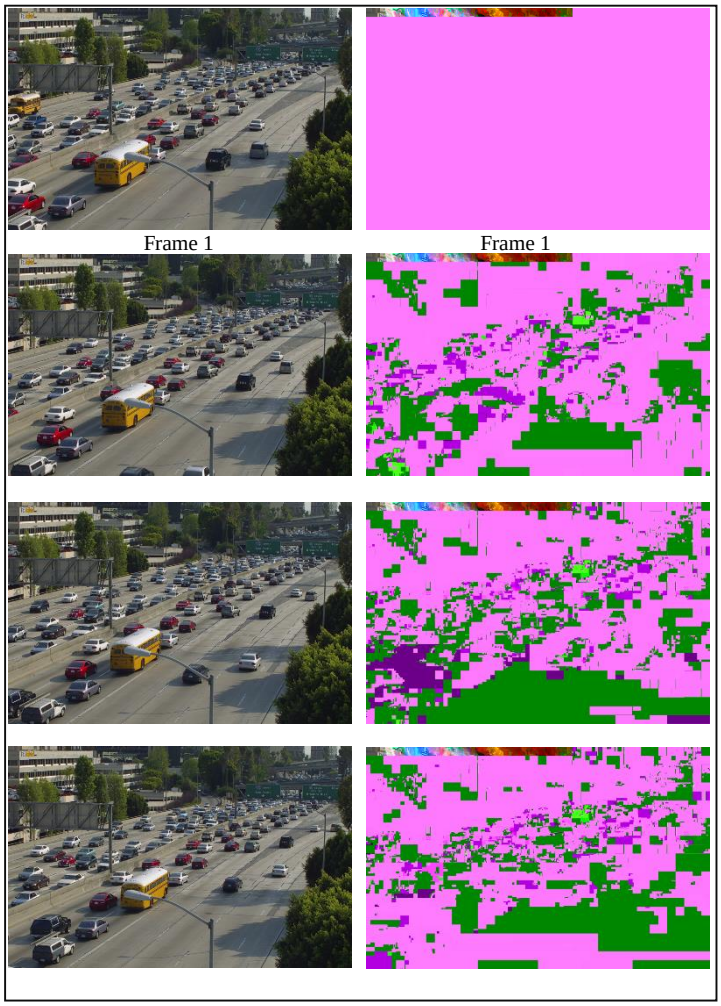


**Figure 6** Encryption for *Traffic* video sequences using random access coding configuration

**Table 5** SSIM comparison for original and encrypted video sequences

| Sequence | SSIM (Y)dB | | SSIM (U) dB | | SSIM (V) dB | |
|---|---|---|---|---|---|---|
| | Original | Encrypted | Original | Encrypted | Original | Encrypted |
| Traffic | 0.94 | 0.29 | 0.91 | 0.59 | 0.94 | 0.63 |
| PeopleOnStreet | 0.92 | 0.36 | 0.95 | 0.73 | 0.95 | 0.74 |
| ParkScene | 0.90 | 0.23 | 0.90 | 0.40 | 0.90 | 0.46 |
| BasketBallDrill | 0.89 | 0.42 | 0.95 | 0.53 | 0.95 | 0.63 |
| BQMall | 0.94 | 0.30 | 0.93 | 0.56 | 0.94 | 0.60 |
| PartyScene | 0.92 | 0.08 | 0.90 | 0.19 | 0.91 | 0.18 |
| RaceHorseC | 0.91 | 0.25 | 0.91 | 0.44 | 0.93 | 0.47 |
| BasketBallPass | 0.91 | 0.24 | 0.94 | 0.73 | 0.95 | 0.75 |
| BQSquare | 0.90 | 0.12 | 0.92 | 0.84 | 0.94 | 0.81 |
| BlowingBubbles | 0.90 | 0.24 | 0.90 | 0.52 | 0.93 | 0.66 |
| RaceHorses | 0.91 | 0.17 | 0.92 | 0.37 | 0.93 | 0.38 |
| Vidyo1 | 0.95 | 0.40 | 0.97 | 0.71 | 0.98 | 0.73 |
| Vidyo3 | 0.95 | 0.40 | 0.98 | 0.55 | 0.97 | 0.55 |
| Vidyo4 | 0.95 | 0.46 | 0.98 | 0.75 | 0.98 | 0.74 |
| Average | 0.92 | 0.28 | 0.93 | 0.57 | 0.94 | 0.60 |

## Computational Analysis

Table 6 shows the results of the encoding and decoding times of different benchmark video sequences with and without encryption and decryption processes. From the obtained results, it can be deduced that the difference in time is negligible with respect to the standards required for video real-

time streaming [27][28]. Due to the low encryption data percentage as shown in Table 3, the computational complexity of the encryption process is also low. In other words, the percentage of encrypted data is directly proportional to the computational complexity and time delay, where the encrypted data are low compared to the total video data.

The impact of the encryption process on the computational complexity of a specific video sequence (the BasketBallPass) is depicted in Figure 7. The differences in the time requirements of encoding and decoding for the encrypted and non-encrypted frames are very small, indicating that the delay effect can be assumed to be negligible.

## Security Analysis of the Proposed Scheme

In order to check whether the encryption approach is robust and valid, it should be analyzed based on some criteria. This section presents the security analysis for the proposed scheme in terms of the frame entropy, local standard deviation, the video correlation, the encryption key effectiveness and the security level of the approach in the face of known plaintext as well as brute force attacks.

## Analysis for Entropy and Local Standard Deviation

The frame data contents have been analyzed by calculating the frame entropy and local standard deviation of the original and the encrypted frames. According to [29], the original frame has higher entropy and lower standard deviation value than the encrypted frame. The frame entropy is defined as in equation (3). By exploiting the pixels of the encrypted frame, the local standard deviation is calculated as in equation (4). $\overline{p(j)}$ is the local mean of the neighbor pixels, $a_i$ is the frame gray levels, $p(a_i)$ is the probability of gray level, $k$ is the number of bit per pixels and $m$ is the pixel block size used to calculate the local mean and standard deviation.

$$E(X)=-\sum_{i=0}^{2^k} p(\alpha_i)\log 2(\alpha_i). \quad (3)$$

$$\sigma(j)=\sqrt{\frac{1}{m}\sum_{i=1}^{m} p(i)-\overline{p(j)}} \quad (4)$$

By testing the entropy of the original frame of the Traffic 2560×1600 video sequence, we obtained E(X) = 7.5447 bits/pixel, while the entropy of the encrypted frame is E(X) = 0.1171 bit/pixel. As for the local standard deviation σ(j), the values of all pixel in the original and encrypted frames of the Traffic 2560×1600 video sequence have been analyzed to evaluate their variation. The obtained value of the mean local standard deviation for the original frame is 86.0712 gray levels, while the mean local standard deviation of the encrypted frame is equal to 194.6586 gray levels. From those results, we concluded that the encryption approach is a robust protection against the statistical attacks.

**Table 6** Processing time of encoding and decoding with and without encryption for all benchmark video sequences

| Sequences | Time (s) | | | |
|---|---|---|---|---|
| | Original | | Encrypted | |
| | Encode | Decode | Encode | Decode |
| Traffic | 5,003.95 | 18.09 | 5,104.50 | 18.57 |
| PeopleOnStreet | 8,470.46 | 32.02 | 8,723.59 | 34.20 |
| ParkScene | 2,738.56 | 10.40 | 2,752.60 | 10.64 |
| Kimono | 3,954.82 | 14.43 | 3,964.03 | 14.48 |
| BasketBallDrill | 656.45 | 2.65 | 666.01 | 2.68 |
| BQMall | 548.29 | 2.20 | 548.32 | 2.29 |
| PartyScene | 743.13 | 3.63 | 748.23 | 3.60 |
| RaceHorseC | 962.97 | 3.84 | 999.04 | 3.91 |
| BasketBallPass | 432.59 | 1.35 | 434.97 | 1.38 |
| BQSquare | 137.04 | 0.72 | 137.60 | 0.76 |
| BlowingBubbles | 155.39 | 1.10 | 160.60 | 0.77 |
| RaceHorses | 244.38 | 1.06 | 243.02 | 1.09 |
| Vidyo1 | 950.64 | 3.47 | 956.95 | 3.44 |
| Vidyo3 | 1,019.74 | 3.57 | 1,004.45 | 3.62 |
| Vidyo4 | 970.18 | 3.53 | 974.02 | 3.46 |

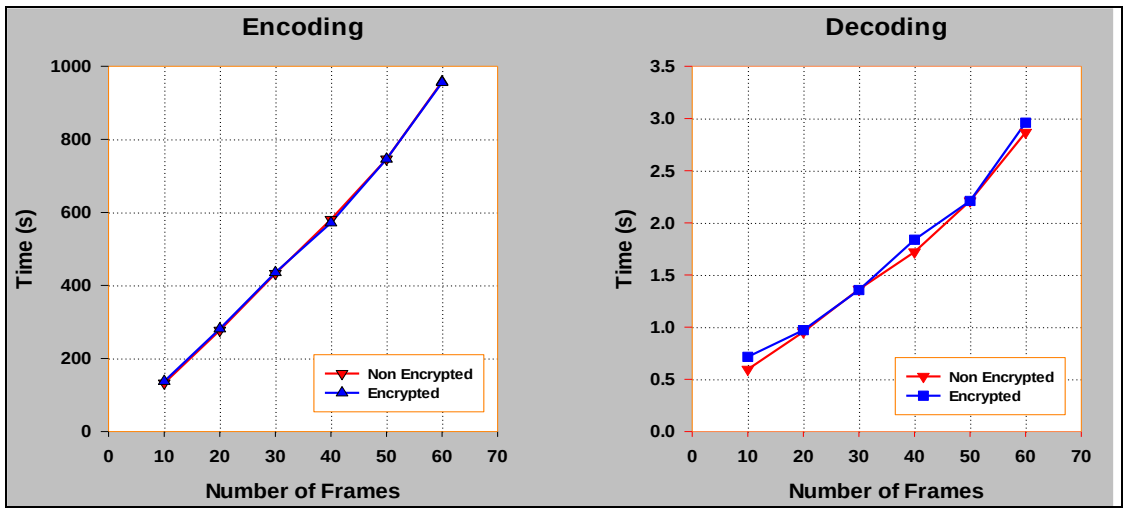


**Figure 7** Time taken by encryption of *BasketBallPass* video sequence

## Correlation between Pixels

In the normal frame, the correlation between pixels is high. Thus, if the encrypted frame has still high correlation among its pixels or even the same value of correlation as the original frame, then that frame can be detected, i.e. the encrypted information contents of a frame can easily reveal the relation to the original image. The encryption process is considered as highly secured if the correlation between the neighboring pixels is low after encryption. Our proposed approach produces low pixels correlation within the encrypted frame. The correlation of a pixel with its neighboring pixel is specified by equation (5).

$$corr(x,y)=\frac{1}{n-1}\sum_{0}^{n}\left(\frac{x_i-\overline{x_i}}{\sigma_x}\right)\left(\frac{y_i-\overline{y_i}}{\sigma_y}\right) \quad (5)$$

where $n$ is the number of pixels, $\overline{x_i}$ and $\overline{y_i}$ are the horizontal and vertical local means respectively, $\sigma_x$ and $\sigma_y$ represent the standard deviations of pixels x and y respectively. Using equation (5), the correlation between adjacent pixels in the original frames of BasketBallPass video is 0.964, while it is equal to 0.0664 in encrypted frames. These values clearly show the big difference between pixel correlation of the original and encrypted frames.

## Sensitivity to the Encryption Key

The sensitivity of the encryption scheme to the key is one of the main factors to guarantee the scheme's security. The cryptosystem is not robust if it is not sensitive to small changes in the key. We applied the proposed method with one bit change in the decryption key, and listed the quality analysis of encryption sequence in Table 7, while the retrieved visual result is shown in Figure 8.

**Table 7** Impact of one-bit change in the decryption key of the *BasketBallPass* video sequences

| Frame number | Original Key | | 1bit change | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| 1 | 36.7515 | 0.9252 | 13.6282 | 0.3347 |
| 2 | 36.1781 | 0.9223 | 13.6791 | 0.3389 |
| 3 | 35.866 | 0.9198 | 13.7103 | 0.3408 |
| 4 | 35.5908 | 0.9172 | 13.7255 | 0.3425 |
| 5 | 35.7281 | 0.9179 | 13.7244 | 0.343 |
| 6 | 35.6563 | 0.9173 | 13.7315 | 0.3438 |
| 7 | 35.5738 | 0.9165 | 13.7352 | 0.3438 |
| 8 | 35.4582 | 0.9154 | 13.739 | 0.3439 |
| 9 | 35.5524 | 0.9158 | 13.737 | 0.3434 |
| 10 | 35.5257 | 0.9154 | 13.7473 | 0.3435 |

**Figure 8** Key sensitivity for encrypted frame #1 of *BasketBallPass* video sequence, encrypted with1-bit different key

### Known Plaintext and Brute Force Attack

The concept of Known Plaintext Attack (KPA) refers to the prediction of encrypted data based on the knowledge of non-encrypted data. In case the encrypted data is a single bit, it can be easily retrieved using the brute force attack. In the proposed approach, the encrypted data length is eight bits. In addition, every intra prediction direction is based on the reference coding units, thus it gives a robust encryption against the brute force attack. Since AES algorithm is used, the ciphertext is not vulnerable to known plaintext attacks as in [30]. Furthermore, according to the work in [31], deriving the key using known plaintext or a brute force attack from data that are encrypted by AES is difficult.

### Comparative Analysis

In this section, the proposed scheme is compared with the predecessor schemes. The efficiency of encryption schemes based on the encrypted syntax element, the percentage of selected data; the encryption algorithm; effective of the encryption on visual information of video and compression efficiency. Thus, the encrypted element and its percentage concurrent with encryption algorithm have to be selected carefully. Some of the recent work produce high level security but with high percentage of the encryption data while the others select low percentage of encryption data but produce low security level. For instant in Shahid [8] the high security level was provided but the average of percentage of the encrypted data ([8] Table II) is 18.45 while in our scheme the high security level was provided while and the percentage of encrypted data is 10.61. On the other hand in Hofbauer [7], the percentage of selected data is low but the security level is low as well [22]. In A. Saleh [23] the proposed scheme was dedicated on the encryption for the moving information only.

Table 8 summarizes the comparison between our proposed scheme and recent work in terms of: encrypted elements; processing time; computational complexity; bitrate overhead; encrypting algorithm; encrypted information; security level.

**Table 8** Comparative summarizing of proposed scheme and the previous works

| Scheme | C1 | C2 | C3 | C4 | C5 | C6 | C7 |
|---|---|---|---|---|---|---|---|
| Hofbauer [7] | Sign bits for luminance coefficients | Yes | Na | Na | AES | Visual | Low |
| Shahid [8] | Suffix of TRk, Suffix of EG0, Sign of MVDs and Suffix of EG | Yes | Yes | High | AES | Visual | High |
| A. Saleh [23] | abs_mvd_minus2 | Zero | Zero | Low | AES | Moving | High |
| Our scheme | Horizontal Intra Prediction modes | Zero | Zero | Zero | AES | Visual | High |

C1 Encrypted element
C2 Increase in time
C3 Increase in computational complexity
C4 Increase in bitrate
C5 Encrypting algorithm
C6 Encrypted information
C7 Security level

## 4.0 CONCLUSION

In this paper, a new method has been developed to provide lightweight and robust encryption for video bitstreams in the context of the recent video coding standard HEVC. We started by analyzing the HEVC entropy coding and reviewing the main previous

proposed approaches of HEVC encryption alongside their limitations, before describing the proposed method. The proposed algorithm adopts the selective approach using the AES standard to encrypt syntax elements the horizontal intra prediction modes of CABAC. In addition to security level, the computational complexity of the proposed scheme for different video sequences was analyzed under the Low delay and random access profiles. The experimental results showed that the new method generates reliable perceptual security, protection against key plaintext attacks as well as brute force attacks, while maintaining compliance to the HEVC standard's bitstream format with low computational complexity. Consequently, we conclude that the proposed encryption method is suitable for real-time video applications and systems with limited resources.

## Acknowledgement

The authors would like to thank the Ministry of Higher Education (MOHE) Malaysia for providing the grant 600-RMI/NRGS 5/3(5/2013), and Research Management Institute of Universiti Teknologi Mara for supporting this research work.

## References

[1] H. K. Arachchi, X. Perramon, S. Dogan, and A. M. Kondoz. 2009. Adaptation–aware Encryption of Scalable H.264/AVC Video for Content Security. *Signal Process. Image Commun.* 24(6): 468-483.

[2] J. Jegajeevanram and M. Sangeetha. 2014. Concealing Information In H.264/AVC Compressed Video Using DWT. *International Journal of Advanced Technology in Engineering and Science.* 2(11): 384-389.

[3] Cisco. 2015. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014–2019. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html. [Accessed: 29-Jul-2015].

[4] B. Girod, E. Steinbach, N. Farber, and I. Farber. 1995. Comparison of the H.263 and H.261 Video Compression Standards. Standards and Common Interfaces for Video Information Systems (SPIE). Philadelphia, USA. 25-26 October, 1995. 233-251.

[5] D. Austerberry. 2005. *The Technology of Video and Audio Streaming*. 2nd edi. USA: Elsevier.

[6] M. A. Saleh, H. Hashim, N. Tahir, and E. Hisham. 2014. Review for High Efficiency Video Coding (HEVC). *IEEE Conf. Syst. Process Control* (ICSPC). Kuala Lumpur, Malaysia. 12-14 Dec. 2014. 141-146.

[7] H. Hofbauer, A. Uhl, and A. Unterweger. 2014. Transparent Encryption for HEVC Using Bit-Stream-Based Selective Coefficient Sign Encryption. *IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP).* 4-9 May, 2014. 1986-1990.

[8] Z. Shahid and W. Puech. 2014. Visual Protection of HEVC Video by Selective Encryption of CABAC Binstrings. *IEEE Trans. Multimed.* 16(1): 24-36.

[9] M. P. Nirmala. 2014. Commutative Protection System for High Secure Video Coding. *Trans. Eng. Science.* 2(1): 6-10.

[10] T. Davies and A. Fuldseth. 2011. JCTVC-F162: Entropy Coding Performance Simulations. *Jt. Collab. Team Video Coding*.

[11] G. J. S. Vivienne Sze, Madhukar Budagavi. 2014. *High Efficiency Video Coding (HEVC) Algorithms and Architectures*. 1st edi. London. Springer.

[12] Iain E. Richardson. 2010. *The H.264 Advanced Video Compression Standard*. 2nd edi. UK. Wiley.

[13] D. Marpe, H. Schwarz, and T. Wiegand. 2003. Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard. *IEEE Trans. Circuits Syst. Video Technol.* 13(7): 620-636.

[14] Sharman, K. James, Gamei, and J. Alexander. 2015. Data Encoding And Decoding. WO 167299 Al.

[15] B. Peng, D. Ding, X. Zhu, and L. Yu. 2013. A Hardware CABAC Encoder for HEVC. *IEEE Symposium on Circuits and Systems (ISCAS).* Beijing, China. 19-23 May, 2013. 1372-1375.

[16] M. Wien. 2015. *High Efficiency Video Coding Coding Tools and Specification*. 1st edit. London. Springer.

[17] J. Lou and L. Wang. 2013. Method of Determining Binary Codewords for Transform Coefficients. US 0202029 A1.

[18] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien. 2007. Joint Draft ITU-T Rec. H.264 | ISO/IEC 14496-10 / Amd.3 Scalable Video Coding. *Jt. Video Team JVT-X201*. 108(563).

[19] T. Lookabaugh and D. C. Sicker. 2004. Selective Encryption for Consumer Applications. *IEEE Conference Consumer Communications and Networking(CCNC).* 5-8 Jan. 2004. 124-129.

[20] A. Pande, P. Mohapatra, and J. Zambreno. 2013. Securing Multimedia Content using Joint Compression and Encryption. *IEEE Multimed.* 20(4): 50-61.

[21] S. Gupta, L. Kishor, and D. Goyal. 2014. Comparative Analysis of Encrypted Video Streaming in Cloud Network. *Int. J. Comput. Sci. Inf. Technol.* 5(4): 5470-5476.

[22] Y. Wang, S. Member, O. Neill, and S. Member. 2013. A Tunable Encryption Scheme and Analysis of Fast Selective Encryption for CAVLC and CABAC in H.264/AVC. *IEEE Transic. Circuits Syst. Video Technol.* 23(9): 1476-1490.

[23] M. A. Saleh, N. M. Tahir, and H. Hashim. 2016. Moving Objects Encryption of High Efficiency Video Coding (HEVC) using AES Algorithm. *J. Telecommun. Electron. Comput. Eng.* 8(3): 31-36.

[24] M. A. Saleh, N. Tahir, and H. Hashim. 2015. An Analysis and Comparison for Popular Video Encryption Algorithms. *IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE).* Langkawi, Malaysia. 12-14 April, 2015. 90-94.

[25] A. Tanchenko. 2014. Visual-PSNR Measure of Image Quality. *J. Vis. Commun. Image Represent.* 25(5): 874-878.

[26] Y. A. Y. Al-najjar and D. C. Soong. 2012. Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI. *Int. J. Sci. Eng. Res.* 3(8): 1-5.

[27] M. N. A. and M. Ghanbari. 2013. An Efficient Security System for CABAC Bin-Strings of H.264/SVC. *IEEE Trans. Circuits Syst. Video Technol.* 23(3): 425-437.

[28] G. Raja and S. Rehman. 2014. Performance Evaluation of HEVC over Broadband Networks. *IJCSI Int. J. Comput. Sci. Issues*. 11(4): 68-74.

[29] C. E. Shannon. 1949. Communication Theory of Secrecy Systems. *Bell Syst. Tech. J.* 28(4): 656-715.

[30] A. Unterweger. 2014. Post-Compression Multimedia Security. Ph.D. Thesis. Dept. Comp. Scin., University of Salzburg, Salzburg, Austria.

[31] A. Bogdanov, D. Khovratovich, and C. Rechberger. 2011. Biclique Cryptanalysis of the Full AES. *Conference on the Theory and Application of Cryptology and Information Security*. Springer Berlin Heidelberg. 4-8 Dec, 2011. 344-371.