Moving Objects Encryption of High Efficiency Video Coding (HEVC) using AES Algorithm

Mohammed A. Saleh, Nooritawati Md. Tahir , Habibah Hashim
Faculty of Electrical Engineering,
Universiti Teknologi MARA (UiTM),
40450, Shah Alam, Selangor, Malaysia.
moh3ahm@gmail.com

***Abstract*— In recent time, the security of multimedia information has become a topic of great interest to researchers worldwide. One of the main concerns of multimedia security is content protection techniques which primarily involves encryption. In this paper, we discuss a new technique in encrypting moving objects in High Efficiency Video Coding (HEVC) media. Due to high computational complexity requirements of video encryption, selective encryption for the moving objects in the contents of the video has been encrypted. Vertical data of Motion Vector Difference (MVD) has been selected to be encrypted using the AES algorithm. The result has shown that the scheme provides an adequate security level for the moving objects information while giving consideration to the trade-off between the computational complexity, the encryption reliability and video coding efficiency for a real-time application.**

***Index Terms*— AES algorithm; HEVC; Moving object encryption.**

I. INTRODUCTION

Video coding (compression) standards are developed to minimize the video data size in order to maintain the bandwidth consuming [1]. The first video coding standard was in 1960 as an analog videophone system [2]. After sequences of improvement for the video coding, the current video coding standard is High Efficiency Video Coding (HEVC), where in earlier 2013 the first edition of HEVC standard was completed [3].

HEVC was developed by joining Video Coding Experts Group ITU-T with Moving Picture Experts Group ISO/IEC. Whereas, the main objective of developing the HEVC standard is to support a significant improvement in compression performance compared with the existing video coding standards and give a range of 50% reducing in bitrate of H.264/AVC standard with the same perceptual quality of video [4]. The idea of the video coding (compressing) is exploiting the data redundancy in the video to minimize video data size. In addition to the removing the spatial redundancy in frames using Intra prediction and block transformation, the Inter prediction is used as one of the main methods to remove the temporal redundancy in the sequence of video frames. Since the Inter-frame exploits the difference between the frames sequence, the motion parameters are used to generate the prediction sample of Inter-frame prediction unit PU [3].

The main stages of the video compression in modern video coding are frame partition, prediction, transform and entropy coding. All of these techniques are used to represent the video data in a small size. The entropy coding stage is suitable to perform the encryption process. Generally, there are two types of the entropy coding, Context Adaptive Variable Length Coding (CAVLC) and Context-Based Adaptive Binary Arithmetic Coding (CABAC) [5]. The last versions of HEVC include only the CABAC that firstly introduced in H.264/AVC standard [6], [7], [8].

The CABAC in HEVC combines three main parts, binarization, context modeling, and arithmetic coding. Non-binary syntax elements are converted into binary (bins) in binarization stage; the probabilities of the produced bins are estimated by context modeling. According to that estimated probability, the bins compresses to bits by the binary arithmetic coding [5], [6], [9].

In the HEVC, some of the syntax elements that generated after the prediction and transformation stages are in non-binary form. In order to pass the syntax elements to the context modeling and arithmetic coding, non-binary elements are converted into binary form [10]. The goal of the binarization is to represent the non-binary syntax elements efficiently in less number of bits. Five different methods are used in the binarization process of HEVC: Unary, Truncated Unary (TrU), kth order Truncated Rice (TRk), kth order Exp-Golomb code (EGk) and Fixed Length (FL) coding [7].

The encryption for all video data is not possible even after the compression process. Thus, the selective encryption has been considered to maintain video streaming requirements, format compliance, compression efficiency, and resources. Therefore, different encryption methods have been proposed for securing the video data.

Kwok et al [11] proposed "simultaneous arithmetic coding and encryption scheme utilizing chaotic maps" method on H.264/AVC standard. Whereas the line segments position and direction in the piecewise linear chaotic map are controlled using a secret key.

Hofbauer et al. [12], proposed an encryption approach on HEVC standard by encrypting the sign bits of the luminance data only. However according to [13] the encryption for the sign bits is not provide a higher security level.

Shahid [14] proposed an approach to secure HEVC standard. He has encrypted the sign bit of quantized transform coefficient value, the TRp suffix, the EG0 suffix, the sign of motion vector difference and the suffix of EG1 code using AES-CFB. Since this approach provides a high security level,

the percentage of the encryption data is high that cause the increasing in the computational complexity of HEVC coding standard.

In this paper, we have aimed to develop a relevant method to secure a moving objects information in the video with taking into account, the real-time video streaming, bitrate and maintain the video quality and computational overhead. Here the motion information in the video is secured by encrypting the sensitive syntax elements of HEVC. The selected syntax element has been encrypted by AES algorithm.

The rest of this work has been organized as: The proposed approach has been explained in section 2. In sections 3, the implementation experiment and the results of the proposed method have been illustrated. Finally, in section 4, the conclusion of this paper has been presented.

II. Proposed Approach

The encryption for video is suitable for the CABAC stage in order to select the low value that can give high effect for an encryption process on the selected syntax element. The main goal of this approach is to serve the real-time encryption for moving information of video streaming by generating the same encrypted bit rate compared to non encrypted bit rate with low computational overhead. In the sake of achieving the video motion encryption while avoiding the: computational complexity, delay and fulfill the encryption requirements we have selected specific syntax elements of MVD.

The motion vector of the prediction unit in the HEVC is used to indicate the offset to a prediction reference in a previously encoded frame. The current frame motion vectors encoding can be predicted by utilizing the motion vectors those already encoded in the reference frames. The difference between the reference motion vector and the current motion vector called Motion Vector Difference (MVD), the nonzero values of MVD are encoded and transmitted. The prediction unit is transmitted as a series of syntax elements, including a prediction unit headers and motion vector difference (X, Y).

Furthermore, each moving PU will be represented by four syntax elements as follow:

i. abs_mvd_greater0_flag: Specifies whether the absolute value of a motion vector component difference is available (greater than 0).
ii. abs_mvd_greater1_flag. Specifies whether the absolute value of a motion vector component difference is greater than one.
iii. abs_mvd_minus2: Specifies the reminder of the absolute value of a motion vector component difference.
iv. mvd_sign_flag: Specifies the sign of a motion vector component difference [21].

The absolute value of a motion vector difference is assigned horizontally and vertically. In this method, the vertical of *abs_mvd_minus2* syntax elements are selected to encrypt as the input of the proposed algorithm. That element is encoded with bypass mode, which denoted as the lighter mode in term of computational complexity for encoding.

A. Encryption and Decryption Processes

The selected syntax element of HEVC is *abs_mvd_minus2* that can be encrypted by AES algorithm. Whereas this method gives high secure bit stream and can be performed on the low resource device. The AES-CFB is used to encrypt the stream of *abs_mvd_minus2* values. To encrypt a plaintext Pi and generate Ciphertext using AES-CFB mode, the secret key (Ek) and initialization vector (IV) are required. The encryption process is depicted in Equation (1) and (2).

$$Z_i = Ek(\mathrm{C}_{i-1}) \quad (1)$$

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

For this method, the initialization vector IV is used for the first iteration, Z_i is generated as the key streams by AES-CFB, $\oplus$ is XOR operator, P_i is the plaintext input, C_i is the chipper text. The Ciphertext length is specified to be same as the *abs_mvd_minus2* length, that maintain the bit rate of the encrypted syntax as length as the original syntax. After encrypting the selective syntax, the plaintext $P_i(s)$ was substituted by the ciphertext C_i. In the decoder end, the original *abs_mvd_minus2* value series are retrieved (decrypted) from the ciphertext C_i using the proposed algorithm, rely on AES-CFB encryption mode and same encryption key Ek.

III. Experimental Implementation And Results

In this section, the implementation and the result of the encryption scheme are presented as follow. The test model HM10 of the HEVC has been used to apply our method. The used system properties have been described in Table 1. Whereas the proposed encryption was implemented on different types of video sequences. In the HEVC, the percentage of MVD data is low compared to the total video data. Here *abs_mvd_minus2* is selected to encrypt the moving information of moving objects in the video. The encrypted data compared to total encoded data for the various types of benchmark video sequences in addition to the bit rate and data size of non-encrypted and encrypted video sequences are shown in Table 2. The encryption process is applied on the low delay HEVC coding configuration (i.e. this configuration is used for real-time application) with quantization parameters 32 to 36.

Table 1
Experiment's Pc Properties

Experimental Setup	
Processor	Intel(R) core(TM) i5,CPU 3.00GHZ
RAM	8.00GB
HEVC Test Model	HM10
Coding Configurations	Low Delay
Quantization parameter (Qp)	32 - 36

Table 2
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	5019.46	5043.67	209144.0	210153.00	9.87
PeopleOnStreat	11112.79	11427.50	463033.0	476146.00	21.52
ParkScene	2493.89	2508.38	129890.0	130645.00	10.49
Kimono	2111.17	2140.24	109957.0	111471.00	9.90
BasketBallDrill	1128.20	1143.16	28205.00	28579.00	11.87
BQMall	1738.18	1747.78	36212.00	36412.00	9.31
PartyScene	3783.76	3817.80	94594.00	95445.00	9.52
RaceHorseC	1647.67	1689.62	68653.00	70401.00	14.66
BasketBallPass	343.48	345.12	8587.00	8628.00	13.47
BQSquare	828.19	835.49	17254.00	17406.00	8.70
BlowingBubbles	573.68	576.12	14342.00	14403.00	11.56
RaceHorses	513.14	528.46	21381.00	22019.00	17.57
Vidyo1	1048.37	1055.18	21841.00	21983.00	8.39
Vidyo3	1237.58	1230.00	25783.00	25625.00	10.56
Vidyo4	900.62	904.08	18763.00	18835.00	8.30

A. Video Quality Analysis

The video quality evaluation gives the impact of the encryption method on the video in terms of visual quality. In addition to the visual distortion, the common metric to evaluate the video quality are Peak Signal-to-Noise Ratio (PSNR) [15] and Structural similarity index metric (SSIM) [16]. Here we used PSNR and SSIM metrics together to analyze the system of human vision that particularly extracts the viewing field structural information and to utilize the structural distortion measurement to give high accurate analysis. By using of PSNR and SSIM, we compared the quality of original video with encrypted video.

Figure 1 and Figure 2 describe the PSNR's and SSIM values of frames sequence for each of the original and encrypted RaceHorseC video using the proposed selective encryption algorithm of motion vector difference. Since the moving objects in the video are detected starting from the frame number two (i.e. the first frame denoted as a still frame). Thus, the encryption does not effect on the first frame (I-Frame) or any fixed objects.

The quality analysis results of PSNR and SSIM metrics are presented in Table 3, and giving the PSNR and SSIM of the original and encrypted video is performed for all classes of video sequences. The results differ according to the percentage of the moving objects in the video sequences. From the simulation results, it is clearly concluded that the proposed MVD encrypted approach is only secure the moving objects in video sequences and skips the still objects. As the quality analysis of the encrypted video using MVD encryption approach and based on the visual information, the PSNR and SSIM results of moving objects in different resolution video sequence, we observed that this method is suitable to secure any resolutions type of video.

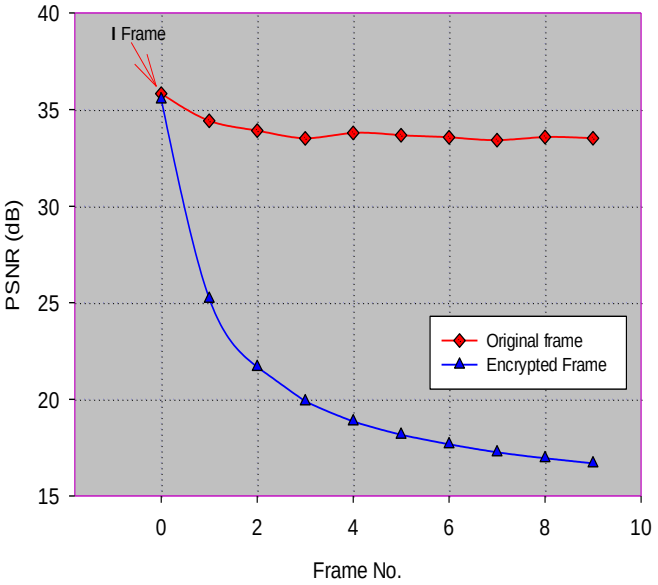

Figure 1: PSNR of non-encrypted and encrypted RaceHorseC video sequence

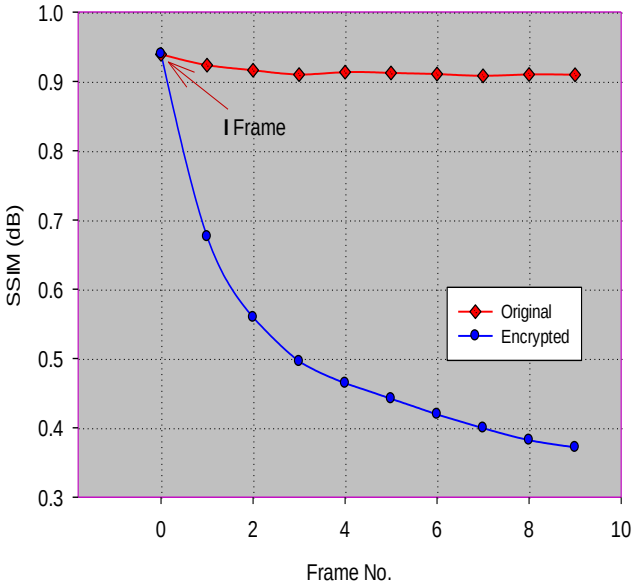

Figure 2: SSIM of non-encrypted and encrypted RaceHorseC video sequence

In Figure 3, the visual distortion for the encrypted videos using the encryption approach is clearly observed for BasketBallPass video sequences.

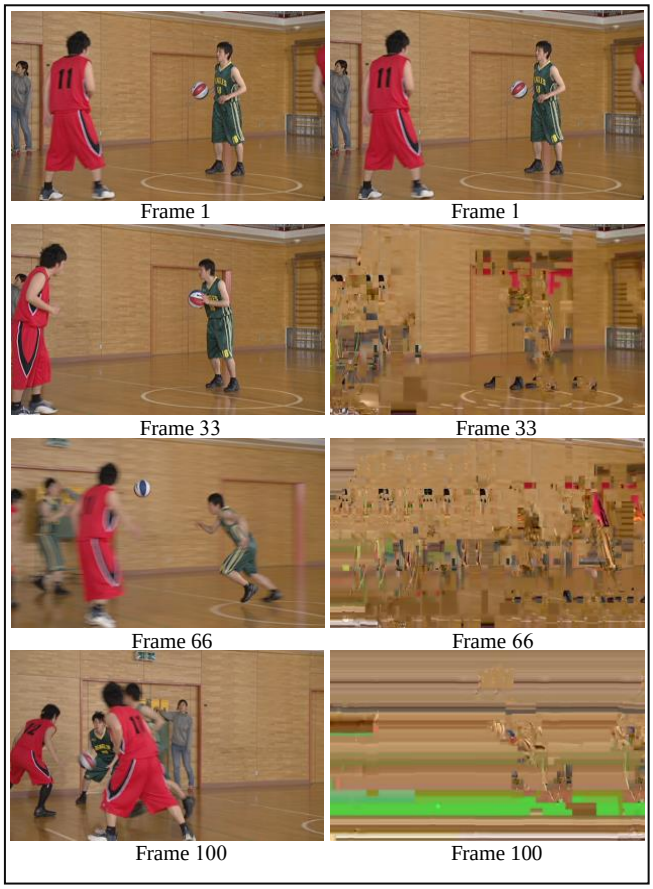

Figure 3: Encrypted frames of BasketBallPass video sequence

Table 3
PSNR and SSIM for original and encrypted Videos sequences

Sequence	PSNR (Y) dB		SSIM (Y) dB	
	Original	Encrypted	Original	Encrypted
Traffic	36.45	21.77	0.94	0.72
PeopleOnStreat	34.54	18.64	0.92	0.58
ParkScene	34.75	20.76	0.90	0.47
BasketBallDrill	34.80	28.02	0.89	0.82
BQMall	34.17	24.44	0.94	0.84
PartyScene	31.12	26.90	0.92	0.79
RaceHorseC	32.41	19.21	0.91	0.47
BQSquare	31.59	23.12	0.90	0.83
BlowingBubbles	33.15	21.42	0.90	0.64
Vidyo1	31.92	24.26	0.95	0.88
Vidyo3	38.90	25.46	0.95	0.85
Vidyo4	38.06	25.03	0.95	0.87
Average	34.17	22.90	0.92	0.73

B. Computational Analysis

Here the cost of our encryption approach in term of encoding and decoding time and CPU load has been analyzed. Since the percentage of encrypted data is directly proportion to the computational complexity and time delay, the percentage of selected data compared to the total encoding data is low as shown in Table 2. The average of that encrypted data is 11.71% for the horizontal and vertical MVD that means, the computational complexity of the encryption process for the vertical *abs_mvd_minus2* is low. Figure 4 describes the difference in the time taken in encoding and decoding for encryption and non encryption process. The results of encryption time show that the difference in encoding time of the encrypted video is small compared to the encoding time of non encrypted video sequences. Thus, the produced delay is acceptable which lead to use this approach for encrypting the videos in real time streaming.

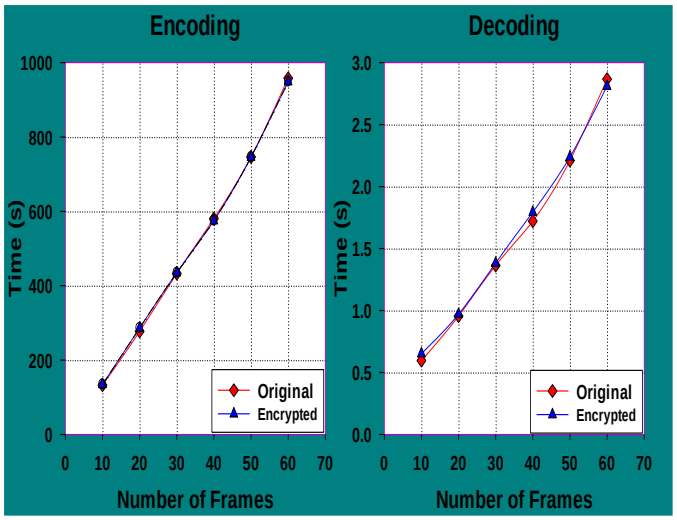

Figure 4: Time taken by encrypting MVD of BasketBallPass video sequence

Table 4 shows the processing CPU weight of the encrypted and non-encrypted videos. In this test, we used seven types' video sequences. The percentage of CPU weight of encrypted and non-encrypted process is likely same, therefore, the different between those percentages is equivalent. Consequently, the impact of the MVD encryption can be denoted as neglected delay.

Table 4
Analysis of CPU Processing Power for HEVC Encoder and Decoder with Encryption and Without Encryption on Low Delay Coding Configuration

Sequence	Encoding CPU% Usage			Decoding CPU% Usage		
	Original	Encrypted	Difference	Original	Encrypted	Difference
BasketBall Pass	19.38	19.25	-0.13	0.09	0.09	0.00
BQSquare	21.74	18.80	-2.94	0.11	0.14	0.03
Blowing Bubbles	15.21	17.88	2.67	0.15	0.09	-0.06
Race Horses	19.93	19.70	-0.23	0.12	0.10	-0.02
Vidyo1	23.56	19.08	-4.48	0.18	0.15	-0.03
Vidyo3	14.48	18.97	4.49	0.12	0.29	0.17
Vidyo4	23.23	22.34	-0.89	0.15	0.19	0.04

C. Security Analysis for MVD Encryption

In this section, the security analysis of the MVD encryption on several types of video class has been presented. The analysis has been accomplished on the encrypted video for; entropy and local standard deviation; the correlation; and Known Plaintext and brute force attack.

i. Analysis for Entropy and Local Standard Deviation

The entropy H(X) of the frame is used for determining the quality of original and encrypted frames, according to Shannon [17], the original frame has higher entropy value and lower redundancy value than the encrypted frame. In addition to the entropy of frames, the local standard deviation $\sigma(j)$ for

the encrypted frames have been analyzed. The entropy and local standard deviation are calculated by Equations (3) and (4) respectively. $p(\alpha_i)$ is the pixels probability of gray level, $\overline{p(j)}$ is the local mean of the neighbor pixel, α_i is the frame gray levels, k is the number of bit per pixels (i.e. in the used version of HEVC, k equal to 8) and m is the pixel block size that used to get the local mean and standard deviation. Using equation (3), the entropy H(X) of original frames is 7.2771 bits/pixel.

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

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

while the entropy H(X) of the encrypted frames is equal to 0.5747 bit/pixel. By analyzing the value of each pixels in the non-encrypted and encrypted frame of RaceHorseC video we got the results as; in the Equation (4), the value of the mean local standard deviation for the non-encrypted frame is equal to 56.4464 gray levels, while the mean local standard deviation of the encrypted frame is equal to 153.1377gray levels. From here, it can be deduced that the MVD encryption method is able to secure the video against statistical attacks.

ii. Correlation between Pixels

The correlation between pixels in any normal image is high because the neighbourhood pixels have rational similarity. Therefore, the correspondence of pixels values in horizontal, vertical and diagonal directions highly occur. On the other hand, if the encrypted image/frame contains high correlation pixels or has similarity in pixels values that mean it can be retrieved, i.e. the encrypted pixels data contents of a frame can give the relation to the original image easily. While the encryption considered as a high secured if the correlation between the neighbouring pixels in the same frame is low. By defining the correlation in horizontal and vertical directions between the adjacent pixels is gotten using the Equation (5).

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

In MVD encryption approach, the correlation between pixels within the encrypted frame is low compared to the correlation between pixels within the original frame. Since the correlation of the original sequence is ≅ 1. By applying equation (5), the correlation result of pixels in original frames of Racehorses sequences is 0.9934, while the pixels correlation in the encrypted frame of Racehorses sequences is equal to 0.1076. Therefore, from the large variance between the pixels correlation of non-encrypted and encrypted frame vertically and horizontally, it can be concluded that the MVD encryption method is strong enough to secure the moving objects in video data.

iii. Known Plaintext and brute force Attack

The known plaintext attack (KPA) is identified as the mechanism for recovering the encrypted data using the non_encrypted data. Since the AES has been used, the ciphertext is not vulnerable to known plaintext attack as in [18]. On the other hand, if the encrypted data is small (one or two bits) for example the flag bits for the Motion Vector Difference (MVD) or sign of transform coefficient, the encrypted data can be easily attacked by the brute force. However, in this encryption approach, the length of selected data is more than two bits, thus, it cannot be effected by brute force attack. Furthermore, according to the [19], the driving key using Known Plaintext Attack or a brute force attack from the encrypted data that encrypted by AES is difficult.

IV. CONCLUSION

This encryption approach was designed by selecting the horizontal vector of motion vector difference syntaxes in HEVC (HM10) to secure all moving object information in a video stream utilizing the AES algorithm. The selected data have been chosen carefully to take into the consideration all of the follow; security level of moving objects in the video; compression efficiency; bit rate increasing; and bit stream formatting compliance. All of the simulation results show that the MVD encryption scheme fulfills the low-resource device in terms of computational overhead, time delay, and bit rate. The proposed method performance gave good results after analyzing the level of security video, quality of encrypted video, statistical analysis of non-encrypted and encrypted video sequences, and the computational cost of the encryption method on the CPU. The simulation results clearly show that the proposed encryption methods had no negative impact on; the efficiency of video compression, video encryption security, data format compliance between encoder and decoder sides, and computational overhead.

This method is denoted as the first method to encrypt the moving object information in HEVC standard. Due to the bit rate maintaining, lack of computational overhead and time delay, this encryption method can be utilized in real-time applications for low resource devices. This method is suitable for most of the video class types with different resolution, frame rate, and it is suitable for all of the HEVC coding configurations.

ACKNOWLEDGMENT

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

REFERENCES

[1] B. Girod, E. Steinbach et al., "Comparison of the H.263 and H.261 Video Compression Standards," in Standards and Common Interfaces for Video Information Systems, Critical, 1995, vol. 60, pp. 233 –251.

[2] D. Austerberry, The Technology of Video and Audio Streaming, Second Edi. 2005.

[3] M. A. Saleh, H. Hashim et al., "Review for High Efficiency Video Coding (HEVC)," IEEE Conf. Syst. Process Control, pp. 141–146, Dec. 2014.

[4] G. J. Sullivan, J. Ohm et al., "Overview of the High Efficiency Video Coding (HEVC) Standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1649–1668, Dec. 2012.
[5] T. Davies and A. Fuldseth, "JCTVC-F162: Entropy coding performance simulations," Jt. Collab. Team Video Coding, 2011.
[6] D. Marpe, H. Schwarz et al., "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 620–636, Jul. 2003.
[7] G. J. S. Vivienne Sze, Madhukar Budagavi, High Efficiency Video Coding (HEVC) Algorithms and Architectures. Springer, 2014.
[8] Iain E. Richardson, The H.264 advanced video compression standard, 2nd Editio. JohnWiley & Sons, 2010.
[9] B. Peng, D. Ding et al., "A Hardware CABAC Encoder for HEVC," in IEEE Symposium on Circuits and Systems (ISCAS), 19-23 May, 2013, pp. 1372–1375.
[10] M. Wien, High Efficiency Video Coding Coding Tools and Specification. Springer, 2014.
[11] K.-W. W. K.-W. Wong, Q. L. Q. Lin et al., "Simultaneous Arithmetic Coding and Encryption Using Chaotic Maps," IEEE Trans. Circuits Syst. II Express Briefs, vol. 57, no. 2, 2010.
[12] H. Hofbauer, A. Uhl et al., "Transparent Encryption for HEVC Using Bit-Stream-Based Selective Coefficient Sign Encryption," in IEEE Conference on Acoustics, Speech and Signal Processing , 4-9 May, 2014, pp. 1986–1990.
[13] Y. Wang, S. Member et al., "A Tunable Encryption Scheme and Analysis of Fast Selective Encryption for CAVLC and CABAC in H.264/AVC," IEEE Transic. Circuits Syst. Video Technol., vol. 23, no. 9, pp. 1476–1490, Sep. 2013.
[14] Z. Shahid and W. Puech, "Visual Protection of HEVC Video by Selective Encryption of CABAC Binstrings," IEEE Trans. Multimed., vol. 16, no. 1, pp. 24–36, Jan. 2014.
[15] A. Tanchenko, "Visual-PSNR measure of image quality," J. Vis. Commun. Image Represent., vol. 25, no. 5, pp. 874–878, Jan. 2014.
[16] Y. A. Y. Al-najjar and D. C. Soong, "Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI," Int. J. Sci. Eng. Res., vol. 3, no. 8, pp. 1–5, Aug. 2012.
[17] C. E. Shannon, "Communication Theory of Secrecy Systems," Bell Syst. Tech. J., vol. 28, no. 4, pp. 656–715, Oct. 1949.
[18] A. Unterweger, "Post-Compression Multimedia Security," Ph.D. thesis, Dept. Comp. Scin., University of Salzburg, Salzburg, Austria, 2014.
[19] A. Bogdanov, D. Khovratovich et al., "Biclique cryptanalysis of the full AES," in Conference on the Theory and Application of Cryptology and Information Security, 4-8 Dec, 2011, pp. 344–371.