HEVC Selective Encryption using Transform Skip Signal and Sign Bin

Yiqi Tew*, Kazuki Minemura* and KokSheik Wong*
* Faculty of Comp. Sci. & Info. Tech., University of Malaya, Malaysia.
E-mail: {yiqi@siswa., kazuki.minemura@siswa., koksheik@}um.edu.my

***Abstract*—A selective video encryption method based on the manipulation of *transform skip signal* and *sign bin* is proposed for the HEVC standard. The basic performance of the proposed selective video encryption method is evaluated in terms of perceptual inspection, outline detection and sketch attack using various classes of test video sequences. Preliminary results show that the proposed method provides quality degradation up to -0.22 in SSIM score when compared to the conventional method [5]. In addition, the edge difference ratio is greater than 0.73, which is closed to the perfect dissimilarity with respect to the original video. Functional comparison between the proposed method and the conventional selective encryption methods is then presented.**

***Index Terms*—video encryption, HEVC, transform skip signal, sign bin**

I. Introduction

In recent years, multimedia data including still image and video are extensively transferred over the Internet thanks to the availability of low cost capturing devices and ubiquitous broadband network connection. While the variety of content may be able to serve a broad range of audience, consumers long for better audio / video quality for the same content. The new video coding standard, i.e., HEVC (High Efficiency Video Coding) is published by ITU-T VCEG and ISO/IEC MPEG [1] to achieve better video compression. HEVC's achieves significant improvement in compression performance when compared to the state-of-the-art standard (i.e., H.264/AVC), with at least 50% reduction in bitrate for producing video of similar perceptual quality [2]. While HEVC is gaining popularity, security and confidentiality of multimedia contents become a challenging research topic. The most straightforward method to secure a video content is to encrypt the entire bitstream by using standard encryption algorithms, e.g., AES (Advanced Encryption Standard). These methods are labeled as NE (Naive Encryption), which treat the video bitstream as binary data without considering the structure of the compressed video [3].

However, NE suffers from several drawbacks. First, the encryption/decryption process becomes computationally expensive for large scale-data, especially for video of high resolution (e.g., 4K and 8K) and high bitrate [4]. Therefore, NE is not suitable for real time video transmission applications, which have rigid restrictions on delay and power consumption on mobile devices. Second, NE prevents untrusted middle-box in the network to perform post-processing operations on the encrypted video bitstream such as transcoding and watermarking. In other words, NE produces a non-format-compliant encrypted video when it is applied directly to the compressed video.

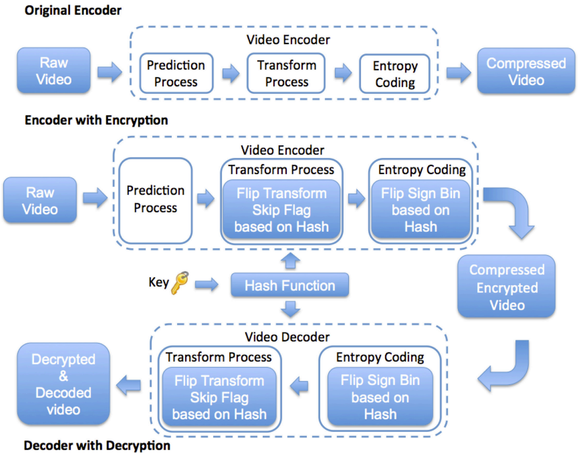


Fig. 1. Original encoding process and encryption/decryption process

As such, selective video encryption emerges as an effective alternative to NE [5], [6]. It considers the coding structure of the video compression standard in question and encrypts only the most sensitive information in the video bitstream. Shahid et al. propose a selective encryption method for HEVC compressed video based on CABAC (Context Adaptive Binary Arithmetic Coding) binstrings in a format compliant manner by utilizing truncated rice code [6]. They put forward an algorithm to convert the encryption space from non-dyadic to dyadic, which can be concatenated to form the plaintext for AES-Cipher Feedback mode. Hofbauer et al. propose another selective encryption scheme for HEVC compressed video which is applicable to a wide range of quantization parameters [5]. Their approach focuses on the AC coefficient signs because the signs are not entropy coded and hence they can be altered directly in the bitstream. This approach enables fast encryption and decryption while maintaining full format-compliance and length-preservation (i.e., identical bitstream size).

II. Proposed Encryption Method

We propose a selective encryption method by utilizing the *transform skip signal* and *sign bin* in the HEVC coding structure. For the selectively encrypted transform skip signal and sign bin, the context of truncated rice code (for binarization of future syntax elements) is left unchanged. Hence the encrypted bitstream is format-compliant and achieves almost the same

TABLE I
AVERAGE PSNR AND SSIM DIFFERENCE IN BETWEEN [5] AND OUR PROPOSED METHOD.

Class	Video Sequences	AI		RA		LP		LB	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
A	PeopleOnStreet	-0.2310	-0.0181	-0.7472	-0.1035	-0.6968	-0.1736	-0.6941	-0.1823
B	Tennis	-2.2663	-0.0694	-0.5394	-0.0982	0.0784	-0.0677	0.3409	-0.0978
C	PartyScene	-0.6635	-0.0879	-1.3805	-0.1404	-0.8230	-0.2040	-1.1859	-0.2225
D	BasketballPass	-1.0648	-0.0906	-1.2700	-0.1372	-0.5271	-0.0875	-0.3777	-0.1237
E	FourPeople	-2.9384	-0.0289	-2.9219	-0.0338	-2.8943	-0.0559	-3.2434	-0.0653
F	ChinaSpeed	-0.2827	-0.0820	0.0508	-0.0910	-0.2262	-0.0897	-0.5254	-0.1110

bit-rate with respect to its original (i.e., plaintext) counterpart. The proposed method requires very little processing power and is ideal for playback on hand held devices. The original video encoder process is shown in the upper part of Fig. 1, while the main idea of the proposed selective encryption and decryption during the encoding and decoding processes are shown in the lower part of Fig. 1.

Here, a secret key is utilized as an input to the cryptography hash function to generate the hash value. These values are utilized to decide the transform skip signal during the transformation process and sign bin during the entropy coding process. To further enhance the encryption effectiveness, we bundle our method with the method proposed by Hofbauer et al. [5] by manipulating the sign bin of AC coefficients and MVD (Motion Vector Displacements). Here, MVD represents the horizontal and vertical movement from the CU (coding unit) being encoded to the matching area in the reference frame and only the direction of MVD (i.e., sign bin of MVD) is manipulated. The resulting method complies to the HEVC format and further distort the video quality of (i.e., produces more encrypted video than) the previous method [5].

A. *Transform Skip Signal*

In HEVC encoder, the option to transform skip signal is configured in the picture parameter set configuration. If activated, a transform skip flag is signaled for each transform block of size 4×4 separately for each color component. The quantizer scaling operation for the coded transform coefficient levels is performed independently of transform skip application. If transform skip is indicated for a transform block, the inverse transform operations are omitted [7], [8].

If a CU has size of 4×4, the encoder has an option to enable transform skip signal. This signal allows encoder to bypass the transform process on that CU and only spatial residual information of CU is encoded. Apart from this, the ringing and blurring artifacts among CUs (i.e., introduced during the reconstruction of the transformed residual signal in the decoder's inverse transform operation) are potentially suppressed when transform skip signal is enabled.

Technically, we toggle the transform skip flag array (i.e., *m_puhTransformSkip*) based on the hash values during the encoding process. The RDO (Rate Distortion Optimizer) in HEVC encoder determines the appropriate coding unit structure by considering the modified transform skip flags. The outcome for toggling the transform skip flag will cause the RDO to pursue a coding unit structure that differs from the originally encoded coding unit structure. This will lead to a slight degradation in video quality, as discussed in Sec. III-A.

B. *Sign Bin*

HEVC stores the sign for coefficients and MVD as they are (i.e., raw and uncompressed) in the bit stream. This makes it straightforward to manipulate the signs directly without impacting the format compliance requirement, while keeping the parsing overhead low. For coefficient sign, a complete sign encryption (i.e., all signs are randomized) is fairly distorted, but even partial sign encryption can introduce sufficient distortions [6]. Therefore, our encryption method toggles the sign of AC coefficients (i.e., *coeffSigns*) of each block while keeping the parsing overhead minimal. Furthermore, our proposed method only toggles sign bits in the luminance channel since the distortion introduced by toggling chrominance channels results in chromatic aberration, which makes the outline more noticeable by the human visual system.

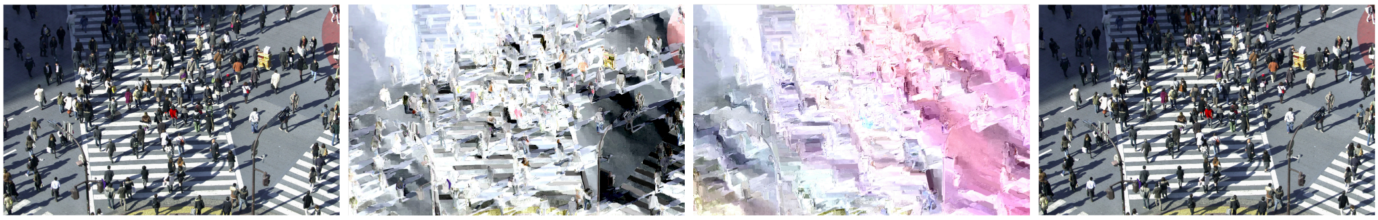
In addition to the transform skip flag manipulation and sign of AC coefficient in the luminance channel, we also toggle the sign of the MVD (i.e., *m_iHor*, *m_iVer*). The DC coefficient sign is left unaltered to avoid extreme drift. Thus, our proposed approach is faster as it minimizes parsing overhead and does not require any modification during the decoding process, i.e., the ciphertext video is format-compliance.

III. RESULT & EVALUATION

HM16.0 reference software [10] is modified to implement the proposed selective encryption method. The performance of the proposed method is verified by using different classes of test video sequence, including PeopleOnStreet (Class A), Tennis (Class B), PartyScene (Class C), BasketballPass (Class D), FourPeople (Class E), and ChinaSpeed (Class F). Each video sequence is encoded and encrypted using Hofbauer's method [5] and our proposed method under four video profiles, namely AI (All Intra), RA (Random Access), LP (Low Delay P) and LB (Low Delay B).

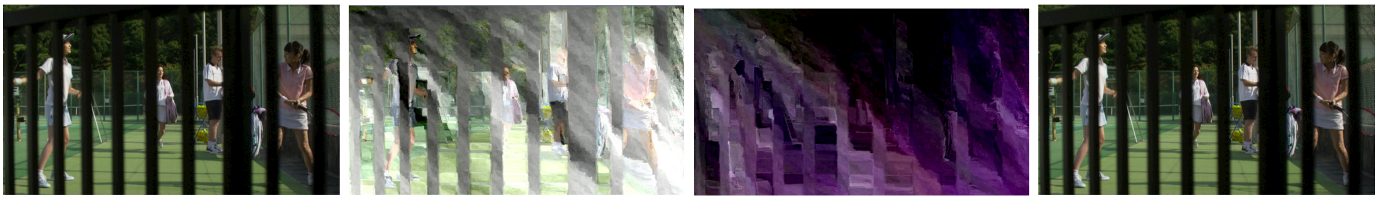
A. *Visual Inspection*

Table I shows the average PSNR and SSIM score [11] differences between Hofbauer's method and our proposed method by using [9], i.e., the difference between "[5] & Plaintext" and "Proposed & Plaintext". Results suggest that most of the average PSNR and SSIM differences are below zero. These values indicate that our proposed method distorts the video quality more to a greater extend when compared to distortion achieved by Hofbauer's method. Particularly, most of the PSNR values drop in the range of 0.07dB to 3.24dB, and the SSIM values drop in the range of 0.01 to 0.22. Note that in RA and LB (i.e., video profiles with B-slices), the PSNR values increase slightly (i.e., 0.05 in Class F and 0.34



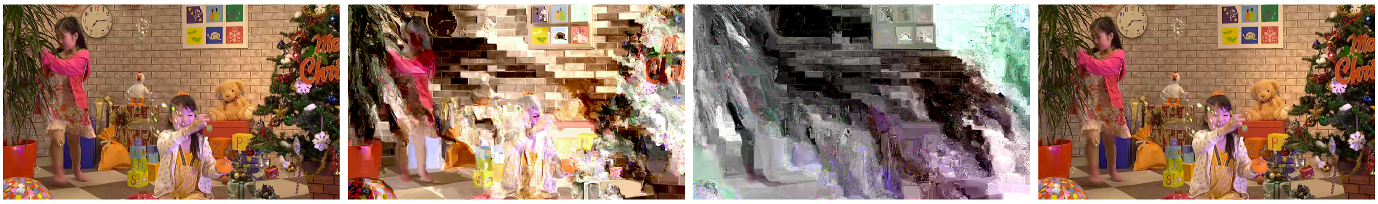
(a) Class A original video (b) Class A with Hofbauer's method (c) Class A with our proposed method (d) Class A decrypted video

Fig. 2. Original and encrypted Class A video by using Hofbauer's and our proposed method



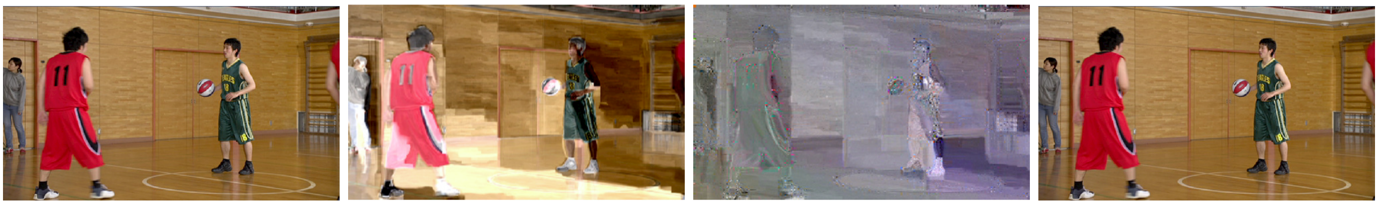
(a) Class B original video (b) Class B with Hofbauer's method (c) Class B with our proposed method (d) Class B decrypted video

Fig. 3. Original and encypted Class B video by using Hofbauer's and our proposed method



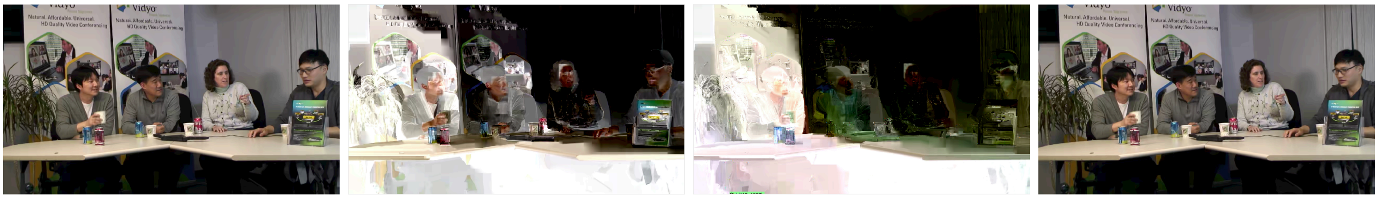
(a) Class C original video (b) Class C with Hofbauer's method (c) Class C with our proposed method (d) Class C decrypted video

Fig. 4. Original and encrypted Class C video by using Hofbauer's and our proposed method



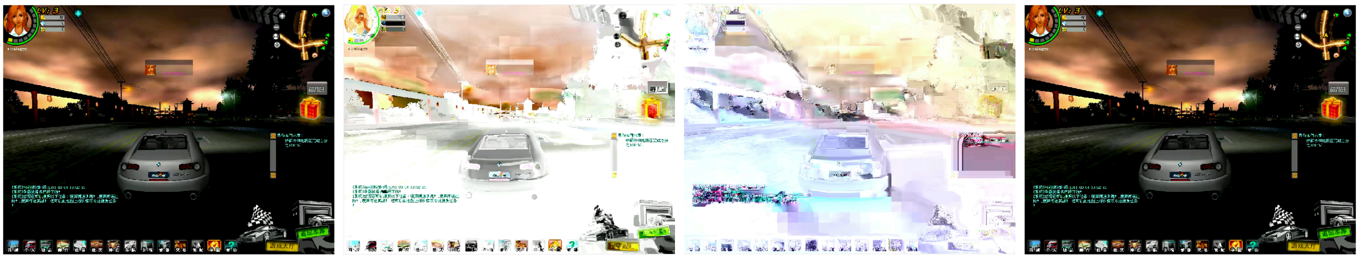
(a) Class D original video (b) Class D with Hofbauer's method (c) Class D with our proposed method (d) Class D decrypted video

Fig. 5. Original and encrypted Class D video by using Hofbauer's and our proposed method



(a) Class E original video (b) Class with E Hofbauer's method (c) Class with E our proposed method (d) Class E decrypted video

Fig. 6. Original and encrypted Class E video by using Hofbauer's and our proposed method



(a) Class F original video (b) Class with F Hofbauer's method (c) Class with F our proposed method (d) Class F decrypted video

Fig. 7. Original and encrypted Class F video by using Hofbauer's and our proposed method

TABLE II
EDGE DIFFERENCE RATIO, $\Re$

Video	CAN		SOB	
(Class)	Hofbauer's	Proposed	Hofbauer's	Proposed
A	0.7618	0.8325	0.7788	0.8822
B	0.7602	0.8519	0.7978	0.9054
C	0.7196	0.7396	0.6482	0.7560
D	0.7227	0.8099	0.6457	0.8303
E	0.7342	0.7738	0.6797	0.8575
F	0.6254	0.8127	0.4752	0.8020

in Class B respectively), probably due to smaller difference between the original plaintext video and the ciphertext video generated by the proposed encryption method (as opposed that generated by [5]). However, the perceptual quality evaluation (measured by SSIM score) is consistently degraded for all video classes and profiles in general. To further illustrate the results, Fig. 2 - 7 show the original, the two encrypted and the decrypted video sequences in Class A, B, C, D, E and F by using Hofbauer's method and our proposed method.

B. Outline Detection

We analyze our encryption method by considering edges (i.e., outline) throughout the encrypted video sequences. Two commonly considered edge detection methods, namely, Canny (CAN) and Sobel (SOB), are chosen to analyze the encrypted video sequences generated by Hofbauer's and our proposed method. Figure 8 - 13 show the detected outline of Class A - F video sequences by using CAN and SOB edge detectors, respectively. These figures consist of detected (i.e., recognizable) edge from the original video (i.e., Fig. (a) and (d) in Fig. 8 - 13), which show a clear outline of object (e.g., basketball players and court lines in Fig. 11(a) and 11(d)). Noted that Fig. 8, 9 and 12 show only a part of the Class A, B and E video slice for closer observation on edge detection. Based on the differences between Fig. (b), (c), (e) and (f) in Fig. 8 - 13, it is evident that the contour lines of the object (e.g., wall and basketball players in Fig. 11(b), 11(c), 11(e) and 11(f)) are increased in our proposed method. That is, the video encrypted by the proposed method produces more complex outline when compared to the encrypted video generated by Hofbauer's method.

The degradation of quality in encrypted video sequences is further evaluated by measuring the edge differential ratio, denoted by $\Re$, between the original and encrypted videos [6]. $\Re$ is computed as follows:

$$\Re = \frac{\sum_{i,j=1}^{N} |P(i,j) - \bar{P}(i,j)|}{\sum_{i,j=1}^{N} |P(i,j) + \bar{P}(i,j)|}, \tag{1}$$

where $P(i,j)$ and $\bar{P}(i,j)$ denote the detected binary pixel values in the original and encrypted video slices, respectively, (i,j) denotes the position of the binary pixel, and N denotes total number of pixels in a video slice. The value of $\Re$ ranges from 0 to 1, where higher value indicates better masking of the structural information of a video slice while lower value indicates higher similarity between the original plaintext and encrypted video frames. Table II shows the average $\Re$ for the encrypted video sequences from various classes generated with Hofbauer's and our proposed methods. It is observed that the $\Re$ value for our proposed method is consistently higher than that of Hofbauer's method (i.e., > 0.73). In other words, our proposed method is able to mask the perceptual meaning of the video more effectively when compared to Hofbauer's method.

TABLE III
COMPARISON WITH OTHER ENCRYPTION METHOD

Encryption method	Functionality			
	Domain	F	C	T
AES Encryption [13]	Bitstream		✓	
NAL unit encryption [14]	Bitstream		✓	✓
Header data encryption [15]	Transform		✓	
Syntax encryption [16]	Bitstream	✓	✓	
Sign encryption [5]	Bitstream	✓		✓
Our proposed method	Trans. & Bits.	✓	✓	✓

F = Format compliant, C = Compression dependent, T = Low computational time

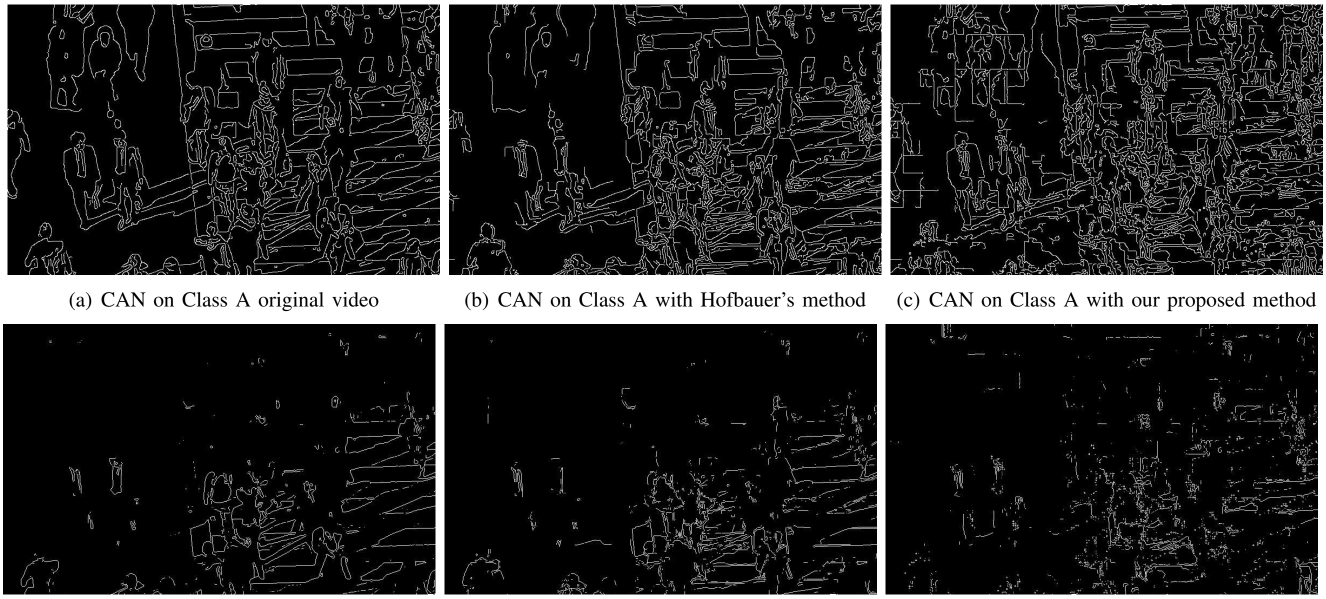
C. Sketch Attack

We apply sketch attack [12] to analyze the encrypted video sequences generated by Hofbauer's and our proposed methods. Results in Fig. 14 and 15 show the sketched images of the original and the encrypted videos in Class C and D. Note that the shape of object can be hardly recognized from the original video (e.g., basketball player in Fig. 15(a)), while the object failed be sketched for both encrypted video (e.g., basketball players in Fig. 15(b) and 15(c)). In addition, the sketched image for our proposed method results is of lower dynamic range (hence the sketch is of lower quality) when compared to that of Hofbauer's method.

D. Functional Comparison

Last but not least, we compare our proposed method with five other encryption methods (i.e., AES Encryption [13], NAL (Network Abstraction Layer) unit encryption [14], Coding Block Header data encryption [15], Syntax encryption [16] and Sign encryption [5]). Table III summarizes the functional comparisons among the video encryption methods considered. Encryption method is indicated as format compliant if it is applicable to the latest HEVC video standard, and able to be decoded while being in the encrypted form (i.e., without decryption prior to decoding). It is found that most methods that manipulate the video content cannot be decoded by using the original decoder except [5], [16] and our proposed method.

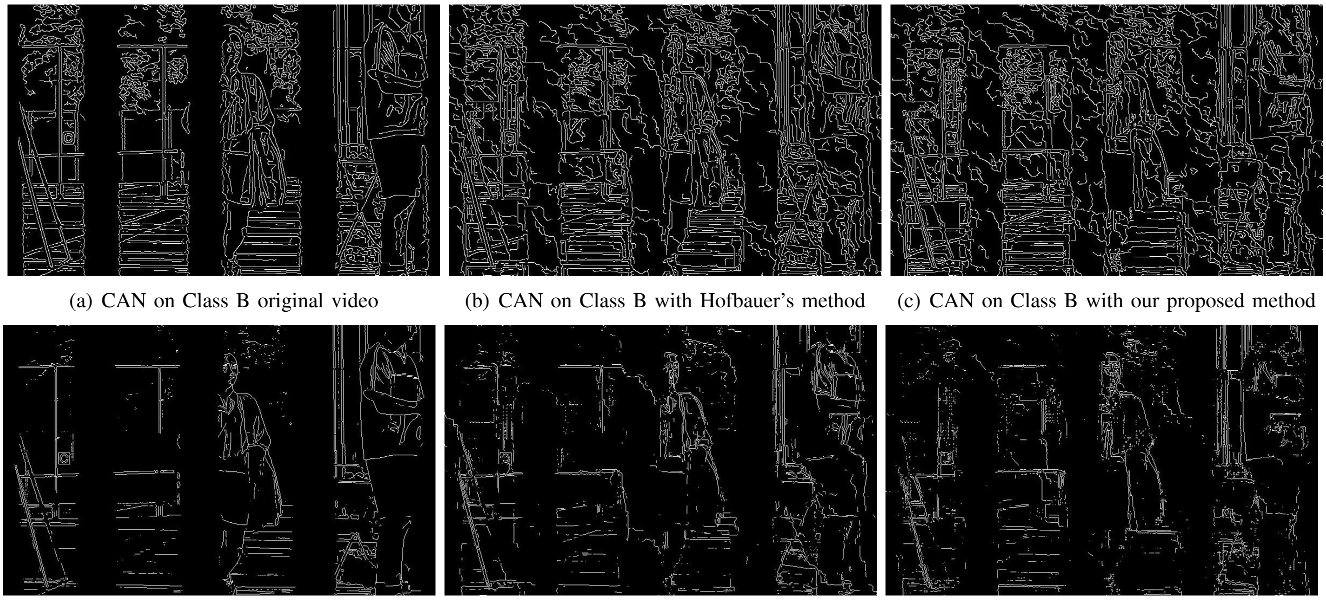
For those methods that modify the video content with respect to the RDO decision, it is indicated as compression dependent. Sign encryption is the only method which does not affect the RDO decision after the encryption process. Hence, our proposed method includes the sign encryption to exploit this advantage. Computational cost for applying encryption method depends on the complexity of the encryption algorithm. Methods [13], [15] and [16] involve high cost operations (e.g., permutation) and long execution time to perform the encryption operation(s) on the particular video components (e.g., coding block header, motion vector displacement). On



(a) CAN on Class A original video (b) CAN on Class A with Hofbauer's method (c) CAN on Class A with our proposed method

(d) SOB on Class A original video (e) SOB on Class A with Hofbauer's method (f) SOB on Class A with our proposed method

Fig. 8. Detected outline for Class A video by CAN and SOB edge detector



(a) CAN on Class B original video (b) CAN on Class B with Hofbauer's method (c) CAN on Class B with our proposed method

(d) SOB on Class B original video (e) SOB on Class B with Hofbauer's method (f) SOB on Class B with our proposed method

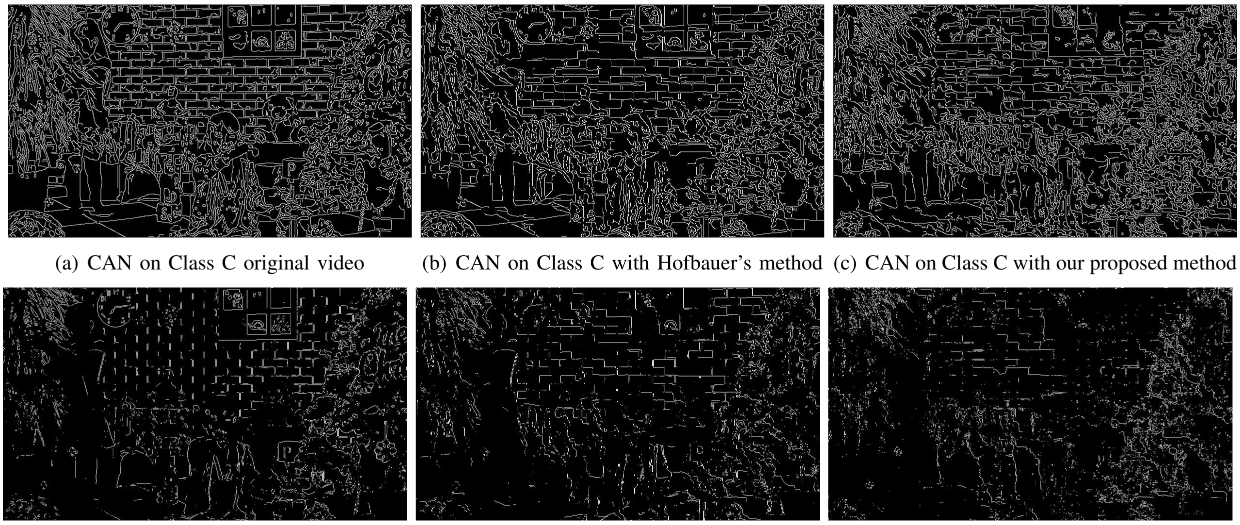
Fig. 9. Detected outline for Class B video by CAN and SOB edge detector

the other hand, NAL unit encryption, sign encryption and our proposed method encrypt a video stream by manipulating particular syntax elements (e.g., *nalUnitType*, *coeffSigns*, *m_puhTransformSkip*) in the HM16.0 encoder during the encoding process. Therefore, these manipulation require minor computational cost, when compared to those that manipulate the video component(s).

IV. Conclusions

A selective video encryption method is proposed by utilizing the transform skip signal and sign bin in the HEVC standard. The coding unit transform skip signal, sign bit of non-zero coefficient and motion vector displacement are manipulated to generate a HEVC format-compliant encrypted video stream. Initial results suggest that the visual quality (e.g., by visual inspection as well as SSIM score) drops significantly when compared to the conventional method [5]. Edge detection results also suggest that our proposed method produces complex edges.

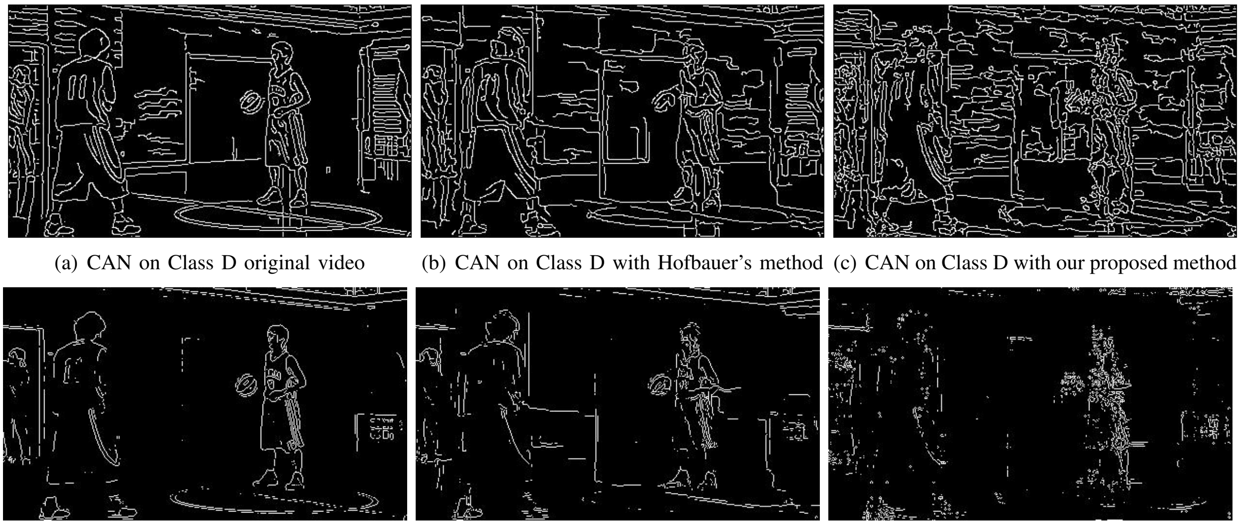
For future work, we shall apply authentication scheme in the encrypted domain for HEVC compressed video by combining the conventional authentication scheme and the proposed encryption methods.



(a) CAN on Class C original video (b) CAN on Class C with Hofbauer's method (c) CAN on Class C with our proposed method

(d) SOB on Class C original video (e) SOB on Class C with Hofbauer's method (f) SOB on Class C with our proposed method

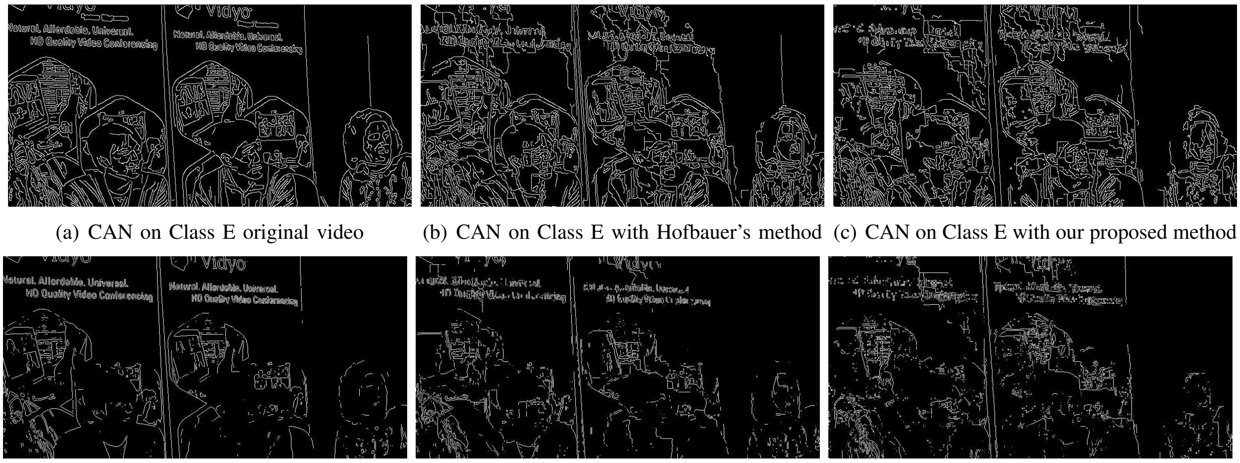
Fig. 10. Detected outline for Class C video by CAN and SOB edge detector



(a) CAN on Class D original video (b) CAN on Class D with Hofbauer's method (c) CAN on Class D with our proposed method

(d) SOB on Class D original video (e) SOB on Class D with Hofbauer's method (f) SOB on Class D with our proposed method

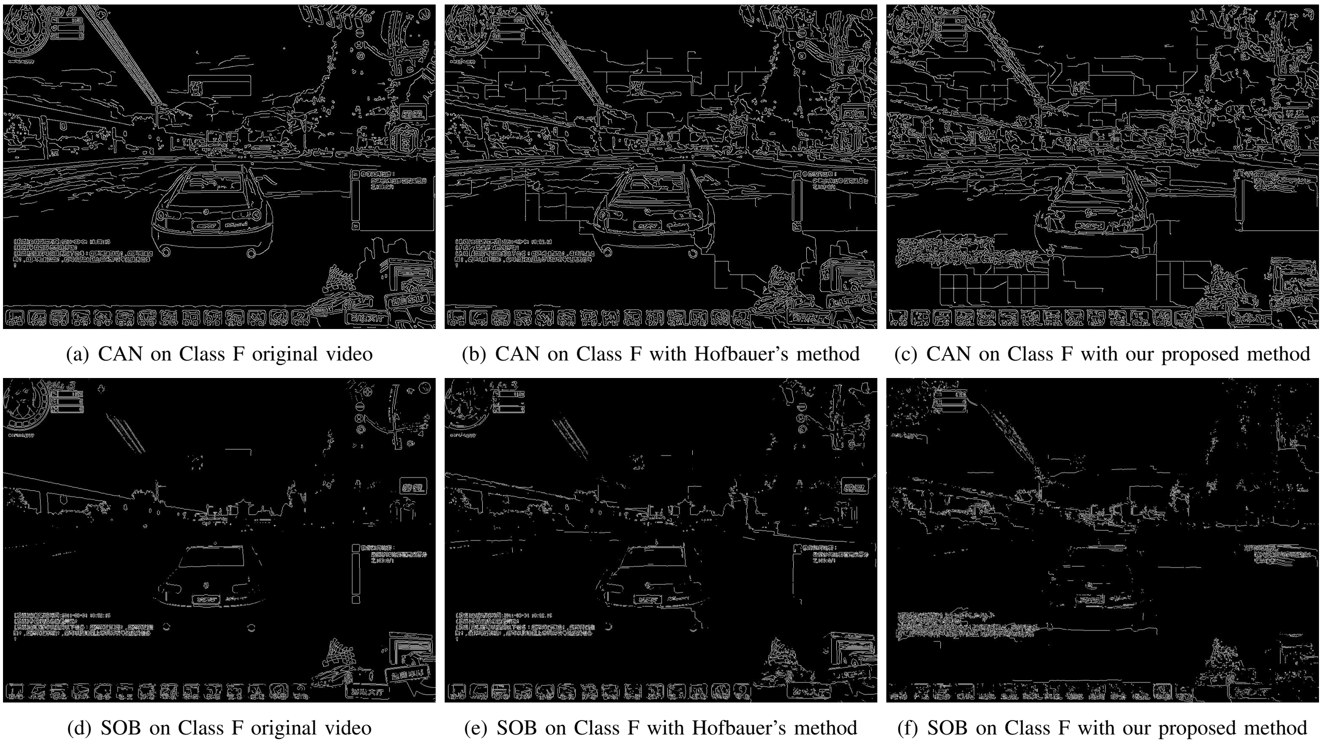
Fig. 11. Detected outline for Class D video by CAN and SOB edge detector


Vidyo
Natural. Affordable. Universal.
HD Quality Video Conferencing

(a) CAN on Class E original video (b) CAN on Class E with Hofbauer's method (c) CAN on Class E with our proposed method

(d) SOB on Class E original video (e) SOB on Class E with Hofbauer's method (f) SOB on Class E with our proposed method

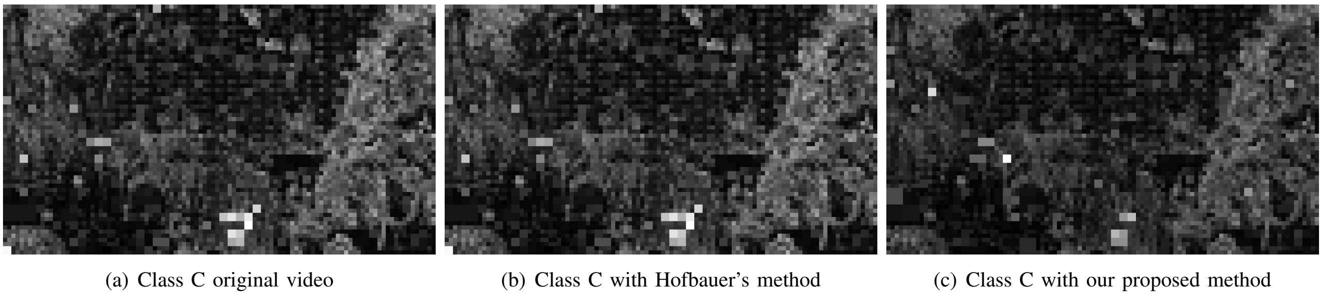
Fig. 12. Detected outline for Class E video by CAN and SOB edge detector



(a) CAN on Class F original video (b) CAN on Class F with Hofbauer's method (c) CAN on Class F with our proposed method

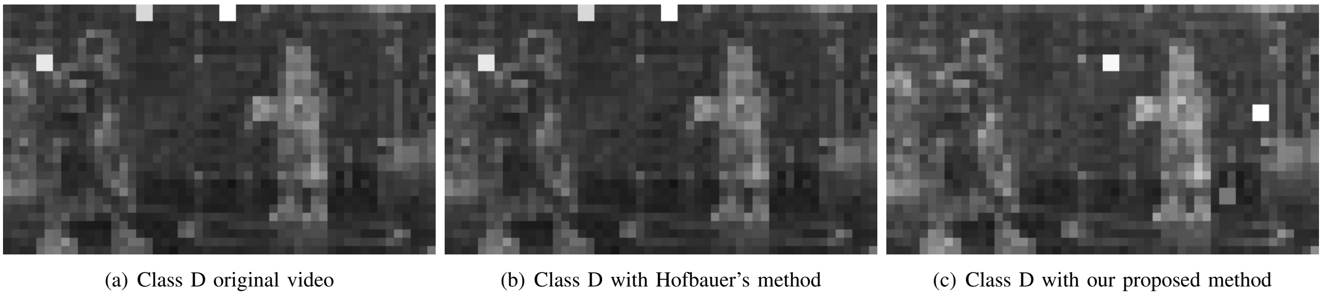
(d) SOB on Class F original video (e) SOB on Class F with Hofbauer's method (f) SOB on Class F with our proposed method

Fig. 13. Detected outline for Class F video by CAN and SOB edge detector



(a) Class C original video (b) Class C with Hofbauer's method (c) Class C with our proposed method

Fig. 14. Sketch Attack Analysis on Class C video



(a) Class D original video (b) Class D with Hofbauer's method (c) Class D with our proposed method

Fig. 15. Sketch Attack Analysis on Class D video

References

[1] ISO, "Information technology - High efficiency coding and media delivery in heterogeneous environments - part 2: High efficiency video coding," *ISO/IEC 23008-2:2015*, Int. Organization for Standardization, Geneva, Switzerland, 2015.

[2] J. -R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards - including High Efficiency Video Coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669-1684, Dec. 2012.

[3] M. Abomhara, O. Zakaria, and O. O. Khalifa, "An Overview of Video Encryption Techniques," *Int. J. of Comput. Theory and Eng.*, pp. 103-110, Feb. 2010.

[4] J. Shah, and V. Saxena, "Video Encryption: A Survey," *IJCSI Int. J. of Comput. Science Issues*, vol. 8, issue. 2, Mar. 2011.

[5] H. Hofbauer, A. Uhl, and A. Unterweger, "Transparent encryption for HEVC using bit-stream-based selective coefficient sign encryption," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, pp. 1986-1990, May 2014.

[6] Z. Shahid, and W. Puech, "Visual protection of HEVC video by selective encryption of CABAC binstrings," *IEEE Trans. Multimedia*, vol. 16, pp. 24-36, Jan. 2014.

[7] M. Wien, *High Efficieny Video Coding: Coding Tools and Specification*, 1st ed., Springer, pp.224, 2014.

[8] V. Sze, M. Budagavi, and G. J. Sullivan, *High Efficieny Video Coding (HEVC): Algorithms and Architectures*, 1st ed., Springer, pp.1-375, 2014.

[9] G. Bjontegaard, "VCEG-M33: Calculation of Average PSNR Differences between RD curves," Video Coding Experts Group (VCEG), Apr. 2001.

[10] Fraunhofer Heinrich Hertz Institute. (2015), *High Efficiency Video Coding: HEVC software repository* [Online]. Available: https://hevc.hhi.fraunhofer.de.

[11] K. Gu; G. Zhai; X. Yang; W. Zhang; M. Liu, "Structural similarity weighting for image quality assessment," *IEEE Int. Conf. on Multimedia and Expo Workshops*, pp.1-6, Jul. 2013

[12] K. Minemura, and K. Wong, "Sketch attacks: A note on designing video encryption method in H.264/AVC," *Asia-Pacific Signal and Inf. Process. Association*, pp. 1-7, Dec. 2014.

[13] D. M. Dumbere, and ; N. J. Janwe, "Video encryption using AES algorithm," *Int. Con. Current Trends in Eng. and Tech. (ICCTET)*, pp. 332-337, Jul. 2014.

[14] C. Li, X. Zhou, and Y. Zong, "NAL level encryption for scalable video coding", *Proc. PCM*, no. 5353, pp. 496505, 2008

[15] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Commutative encryption and watermarking in video compression", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 6, pp. 774-778, Jun. 2007.

[16] X. Wang, N. Zheng, and L. Tian, "Hash key-based video encryption scheme for H.264/AVC," *Signal Process.:Image Comm.*, vol. 15, pp. 427-437, Mar. 2010.