A Pilot Exploration of Industrial Video Scene Data Embedding using Real-Time MV-HEVC

Yik Siang Pang* and Yiqi Tew*
* Faculty of Computing & Information Technology, Tunku Abdul Rahman University College, Malaysia.
E-mail: {pangys-wt12@student., yiqi@}tarc.edu.my

***Abstract*—Video sequences can be coded under Multi-view High Efficiency Video Coding (MV-HEVC) standard efficiently by using simultaneous captured multiple camera views to a single video stream to achieve a real time stereoscopic view. As a successor of Multi-view Video Coding in Advanced Video Coding, MV HEVC performs better compression with bitrate saving up to 50% for multi-view video compared to multiple independent views of earlier coding standard. In this work, we explore the entities of multi-view video compression and decompression processes, investigating the prediction mode manipulation for serving data embedding purpose, and tested on Linux-based Operating System to achieve a near real-time video streaming with data embedding capability. The performance of MV-HEVC with three views streaming are evaluated based on industrial machines views, as a part of Industry 4.0 implementation and compared to the benchmark data sets. The experimental results indicate a minor change in the Bjøntegaard-Bitrate and peak signal-to-noise ratio scales, at minimum 0.0039% and 0.0013dB respectively. Besides, the proposed method also compared to multi-view simulcast HEVC coding in term of execution time, with the average difference ranging from 0.1995 to 3.8839% of additional time required for MV video stream with data embedding application.**

***Index Terms*—Multi-view, MV-HEVC, Data Embedding, Real-time Video Streaming, Industry 4.0**

I. INTRODUCTION

Watching video becomes one of the most consumed multimedia technology, with the existence of various online video sharing platform (e.g., YouTube, Vimeo, TikTok, etc) that relies heavily on Internet traffic. This becoming one of the biggest challenges for future Internet, especially the application of Ultra-High-Definition (UHD), multiple views and 3 Dimensional (3D) video content streaming. According to CISCO global Internet Protocol (IP) traffic forecast, by 2023, it is estimated that 66 % of connected flat-panel TV sets will be UHD [1]. Besides that, the main challenging of future Internet is to deliver multiple view and 3D video. On the other hand, 5G cellular network have colossal improvement on the Internet transmission speed compared to previous version cellular network; it may help to resolve the issue of the higher definition video transmission delay. The new revolution of industrial involves a large amount of data, and it strictly relies on the speed of the data transmission processes, reducing the receiving time delay to achieve near real-time.

One of the common used video compression standard, High Efficiency Video Coding (HEVC) has included an extension to support multiple camera views (multi-view) and serve better visual content to the broad spectrum of user application requirement. Nevertheless, this efficient video compression standard may not serving authenticating genuine video, labeling video content with appropriate tags and protecting video copyright with watermarking. Data embedding into a video bit stream (i.e., compressed video stream) without having a trade off on visual quality for transmission has been focused by various researchers. Various state-of-the-art data embedding method by using intra-prediction [2], discrete cosine transform coefficients [3], syntax elements [4] applied on H.264/AVC [5] and HEVC [6], but yet to be tested in its extension, i.e., multi-view application.

In this paper, we extend our interest to explore the possibility on applying data embedding on HEVC multi-view, with real time streaming in Industry 4.0 application. We have proposed a prototype to perform near real-time multi-view video streaming by implementing data embedding in MV-HEVC video coding. We utilize an adaptive manipulation on block size prediction and preserve the quality of video content. The proposed technique has great potential on handling actual application, including video content authentication from industry 4.0 implementation. Two and three camera views set up are utilized as a reference for our case study. However, the proposed method is also applicable to more views of the camera without changing the code. The camera arrangement studies are not into consideration in this paper, because it will not affect the objective of this work.

II. OVERVIEW OF HEVC AND MULTI-VIEW HEVC

The latest common video compression standard, i.e., HEVC [7] appears in most of the close circuit television monitoring system and it plays an essential role in the video processing industry. This standard reduces the streaming bitrate with identical quality compare to previous video compression standard, i.e., MPEG4/Advanced Video Coding (AVC) for Video On Demand (VOD) and online video streaming applications. The extension of HEVC standard was established in 2014 on 3D video compression format to support stereoscopic, multiple views display with high compression capability and scalability on spatial, quality and color gamut contents. These extensions namely MV-HEVC, 3D-HEVC and SHVC, which was completed in 2014 by (JCT-VC) [8]. With the vast number of camera installed in most of the manufacturing industry that intentionally enrolled in digital transformation for industry 4.0 realization, the market interest and demand of using MV-HEVC can be rapidly increased. The concept of Multi-

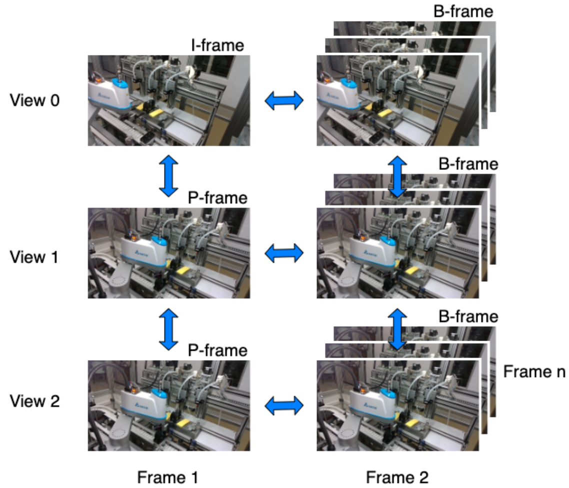

Fig. 1: Inter-layer prediction structure for MV-HEVC

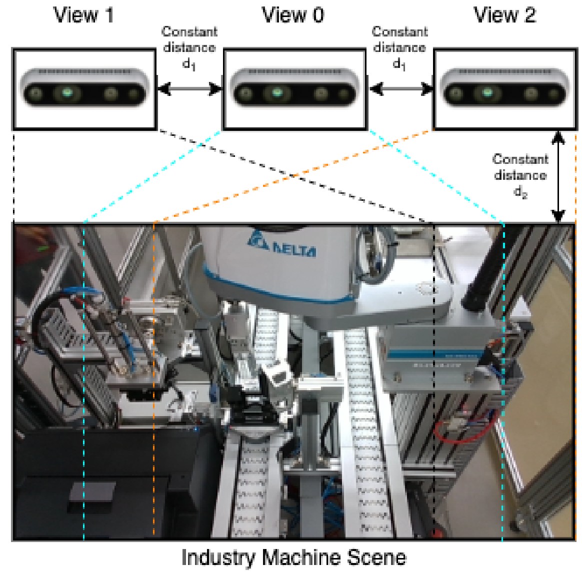

Fig. 2: Multi-view cameras set up in industrial site

view Video Coding (MVC) extension was initially developing in 2009 for H.264/AVC 2009 in 3D-Blu-Ray technology. It supports stereo and multi-view capture by without changing the low-level syntax in a single video stream. MV-HEVC is also a format required to support many promising 3D video applications, such as Free viewpoint TV, 3D TV, immersive teleconferencing and holographic displays by rendering the stereoscopic video in the simplest case, or multi-view auto-stereoscopic video in more advanced scenarios.

MV-HEVC extension is improving the high-level syntax with the feature of disparity compensated prediction (DCP) to reducing inter-view redundancies created higher coding efficiency and compression compared with simulcast [8]. The extension without modifying the base layer (also known as the base view or View 0 in MV-HEVC) which remains to encode by HEVC standard has no inter-layer dependency, while enhancement layers/views (View 1 and more) depend on the base layer. Fig 1 shows the inter-layer prediction structure for MV-HEVC by using one of the manufacturing industry machine video scene. The base layer and base view in MV-HEVC are the same, whereas in 3D-HEVC, it has texture and depth components. The first frame of base view in a Group of Picture (GOP) is under full intra-prediction (I-frame), while the first frame of enhancement view (P-frame) is inter-view predicted from I-frame. Besides, the temporal inter-picture prediction is happening between every single picture/frame in a GOP of the same view and component. A picture under inter-picture dependency namely B-frames. Other than that, the combination of functionalities between temporal inter-picture prediction and inter-view prediction is another prediction model in MVC.

A. *RELATED WORK*

As we know, MV-HEVC is an upgraded version of MVC in H.264/AVC and various researchers has exploited the coding capabilities of MV-HEVC to accelerate the compression and achieve better Rate Distortion Optimization (RDO) performance. Diaz et al. has proposed a hybrid based codec mixed with AVC and HEVC standard to encode multi-view videos by accelerate the Coding Tree Unit (CTU) splitting decision based on statistical Näive-Bayes (NB) classifier [9]. Besides, Jiang et al. and Mallik et al. also developed a optimization multi-view codec based on MV-HEVC standard to reduce coding complexity and required lesser time during encoding process [10], [11]. Along with the exploration on MV-HEVC RDO and coding complexity, we have included data embedding features on MV-HEVC by extending our previous exploration on data embedding in HEVC CTU structure [12].

Similar to HEVC, MV-HEVC consists of CTU structure with some number of Coding Block (CB) in the size from 64×64 down to 4×4 pixels. The sizes variation allows the MV-HEVC encoder to decide the best encoding code representation based on the spatial activity of the video frame. For instance, a small CB size to literally describe the intensity values of each pixel variation of a water waves, and a big CB size to describe a smooth plain wall region in a frame. CB is further depicted as Prediction Block (PB) and Transform Block (TB) in prediction and transformation processes respectively. Our work focus on manipulating PB decision in I-frame, based on the proposed data embedding mechanism, as shown in Fig. 3.

In conjunction with the MV-HEVC coding exploration based on the reference architecture, i.e., HTM-16.3, Springer et al. has developed a HEVC Analyzer for Rapid Prototyping (HARP) to allows real-time HEVC encoding and decoding process with visualisation of coding processes [13]. HARP built a video coding environment with instant responses on video processing with HEVC standard which able to monitor encoder behavior with RDO analysis. Originally, HARP provides two options for HEVC encoder, i.e., HM reference encoder and x265 encoder with lightweight solution. We enhance the HARP code to support the MV-HEVC encoder and decoder with multiple camera for our real time application experimen-

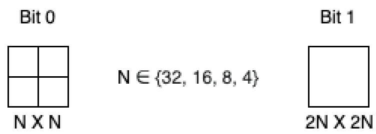

Fig. 3: Data Embedding Method based on PB size decision

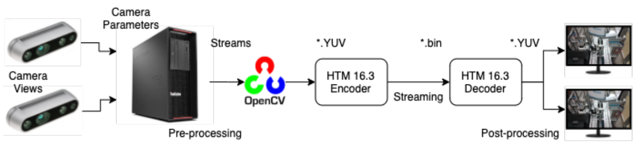

Fig. 4: Live Multi-view Video Streaming Setup

tal work and result collection. This experiment includes the multi-view camera set up at manufacturing industry machines, with the consideration of constant distance in between cameras and object in a scene, as shown in Fig 2.

III. Methodology

A. Apparatus and Prototyping

We consider the original HARP toolkit [13] (mentioned in Section II) as a reference prototyping. HARP includes an open-source single view live streaming of HEVC video codec. Our proposed method has improve the HARP for supporting MV-HEVC and utilized it as part of our project setup. Along with the HARP code improvement, the compression of the multi-view live stream is also associated by HTM-16.3 reference software and parameter setting, as shown in Table I.

The proposed prototype of multi-view real-time video streaming with MV-HEVC extension as shown in Fig 4 and described as follows:

1) Three cameras were employed to capture two raw video streams at frame rate 15fps. The live stream method was validated by the pre-recorded video method as an original result.
2) The proposed prototype is computed by using Ubuntu 14.04.6 LTS (64bit) operating system on eight cores Intel® Xeon(R) CPU E5-1620 v4 @ 3.50GHz Processor with 32GB RAM and without indicated with GPU.
3) The pre-recorded multi-view videos are captured using FFmpeg command-tool [14] and converted to YUV 4:2:0 format. A total of 200 frames are divided into 25 GOP to be encoded. Each GOP consists of 8 frames.
4) In order to imitate the industrial relevant scene, there have two different scenes recorded, which are "Product" and "Filler". The proposed prototype provides instant (real-time) multiple video viewing compared to benchmark video [15] based on recorded (offline) video.

The graphical user interface (GUI) of the proposed work is programmed by Python, associated by a C++ based HTM encoder and decoder with PicklingTools [16], which allowed export of HTM software information as python dictionaries. The OpenCV library helps on capturing the multiple video source from the camera then instantly transforms the video output under YUV format. In addition, there have multiple

Table I: Parameter Setting

Prediction structure	Random Access (IBPBP)
Entropy coding	CAVLC
QP	20,25,30,35
FrameRate	30 fps
Total Frame	200
GOP Size	8
Encoding bit-depth	8-bit

thread arrangement with parallel processing to serve multiple video input with the following consideration:

1) Simulcast HEVC with parallel thread, is able to encode all the video views at the same time by distributing the processing towards number of core processors.
2) MV-HEVC is implemented with parallel video input arrangement. All the views are able to be encoded at the same time and generate only one compressed file compared to simulcast HEVC.
3) Proposed MV-HEVC with data embedding capability, is implemented by modifying the PB decision in all the I-frames, based on two categories: Bit 1 and Bit 0, as shown in Fig 3. Results are collected and analysed based on video quality with respect to the achievable bitrates.

B. Video Scene Capturing

The multi-view videos' test sequences used in this experiment are "Product" and "Filler" scene on industrial site. These two test sequences are newly captured for this project. The reason of not using the existing pre-recorded video by relevant researchers is because the live stream model requires an instant video recording for encoding and decoding on the spot. Both of these test sequences repeat the same process in a loop. For example, "Product" will continuously moved on a line track and "Filler" for the product filling process. We utilized the product movement on a line track as our experimental scene in this project because it is a close loop track in a uniform speed. Therefore, the "Product" sequence is suitable to calculate the average mean of 25 GOPs encoding by live video stream model. Besides, "Filler" is also running a repeating process to fill up a product container, then moved towards a conveyor belt for the next processing section. So, the average mean of instant video capturing and compression method can be validated by the pre-recorded video compression method. We emphasize on the set up to capture the multi-view scene with a constant distance, d_1 and d_2, as shown in Fig 2, to provide a linear camera arrangement on test video sequences.

IV. Results and Discussion

This work aims to build a multi-view HEVC codec with data embedding capability, applied on a live streaming prototype for future MV-HEVC extension work investigation. The proposed prototyping is important to validate on the compression efficiency, in terms of video quality and execution time compare to original results. Therefore, the encoded results of compression efficiency of a two views video live stream model (proposed) is compared with a pre-recorded video compression model (original). Here, we collected results for

Table II: Experimental result for both test and benchmarks [15] video sequences, analysed by respective codecs

Video scenes	QP	Original			Bit 0			Bit 1		
		PSNR	SSIM	Bitrate	PSNR	SSIM	Bitrate	PSNR	SSIM	Bitrate
Product	20	44.8843	0.9873	3727.76	44.8822	0.9873	3726.83	44.8842	0.9874	3729.75
	25	43.3340	0.9811	1672.73	43.3329	0.9812	1672.50	43.3321	0.9815	1675.87
	30	41.3804	0.9694	915.62	41.3764	0.9696	916.30	41.3905	0.9696	916.74
	35	38.9753	0.9537	525.05	38.9778	0.9535	525.88	38.9719	0.9525	525.90
Filler	20	45.6727	0.9728	3477.50	45.6696	0.9727	3483.27	45.6702	0.9726	3482.48
	25	44.0839	0.9653	1617.27	44.0790	0.9652	1617.53	44.0802	0.9652	1620.92
	30	42.0872	0.9531	875.60	42.0825	0.9533	875.84	42.0856	0.9534	878.55
	35	39.7547	0.9335	504.36	39.7409	0.9333	503.61	39.7411	0.9324	504.12
Balloon	20	47.9228	0.9725	1239.84	47.9134	0.9725	1238.48	47.9109	0.9725	1240.05
	25	45.5990	0.9680	615.70	45.5970	0.9679	616.44	45.6049	0.9680	617.58
	30	42.8615	0.9605	320.25	42.8556	0.9607	320.65	42.8550	0.9605	320.52
	35	40.0854	0.9480	171.23	40.0945	0.9479	171.14	40.0913	0.9476	171.47
Kendo	20	43.8706	0.9764	1675.40	43.8685	0.9763	1676.68	43.8673	0.9764	1676.04
	25	42.3367	0.9713	502.06	42.3351	0.9714	502.15	42.3361	0.9714	503.97
	30	40.4395	0.9632	233.00	40.4421	0.9632	232.83	40.4321	0.9635	233.73
	35	38.1334	0.9524	130.74	38.1301	0.9525	130.66	38.1289	0.9520	130.88

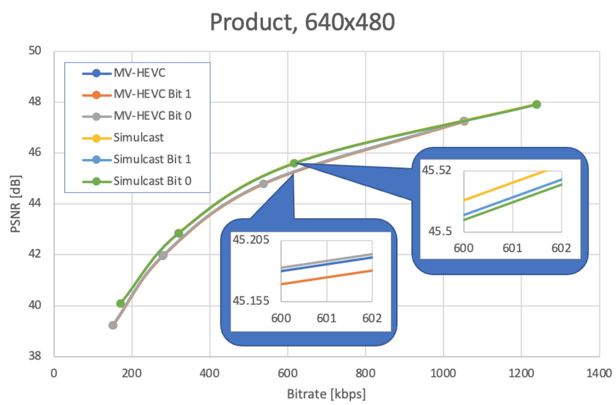

Fig. 5: PSNR vs Bitrate for Product scene

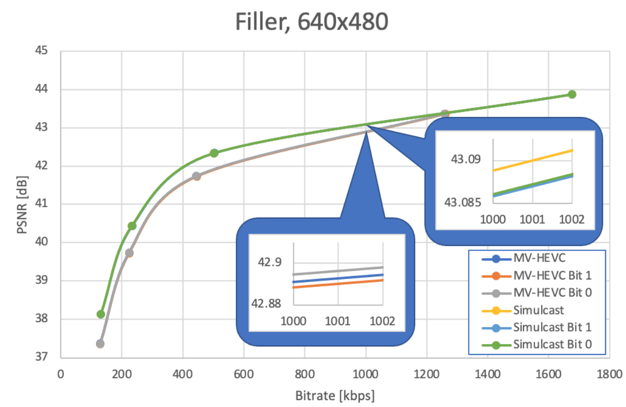

Fig. 6: PSNR vs Bitrate for Filler scene

two industrial video scene, i.e., "Product" and "Filler" and two benchmarks video sequence, i.e., "Balloon" and "Kendo" [15]. Results consists of original and proposed method with Bit 0 and Bit 1 PB decision in I-frame to embed data for the worse case scenario (e.g., assume that all the embedded data are 0 or 1 respectively), in terms of peak signal-to-noise ratio (PSNR) and Structure Similarity Index (SSIM) and achievable bitrate, as shown in Table II. These bitrate and PSNR values are use to further compute and evaluate the performance between proposed method (live stream model) and pre-record video method by using Bjøntegaard-PSNR (BD-PSNR) and Bjøntegaard-Bitrate (BD-rate) calculation [17], as shown in Table III. The Bjontegaard metric is measuring average difference between the two methods of RD-curve to provide relative gain in terms of video quality and bitrate.

From Fig 5 and 6, there are clearly shown that live stream video compression and pre-recorded video compression yielded a similar trend on rate-distortion curve for both multi-view video scenes. An acceptable result validated to original result in Table II based on the RD-curve of live stream video provided. It represents the achieved BD-rate and BD-PSNR for "Product" and "Filler" multi-view videos of the live streaming video model with respect to MV-HEVC pre-recorded video model. The negative value of BD-rate and BD-PSNR showed in this table are indicated that the live streaming model has a bitrate and PSNR differences compare to original model. Table III shows the BD-rate ranging from -0.0105 to 0.0091% and BD-PSNR ranging from -0.2195 to 0.4383dB. This prove that the proposed live video streaming by MV-HEVC video coding could deliver a acceptable video quality and corresponding bitrate, with the proposed data embedding mechanism. Moreover, a simulcast HEVC coding was tested and compared to proposed method, the results show an average BD-rate difference: 0.004 - 0.008% and an average BD-PSNR difference: 0.12 - 0.29dB, this is slight higher gain than the results of (proposed vs MV-HEVC) and it is reasonable because the coding efficiency of MV-HEVC is better than multiple view of HEVC standard. The "Filler" result shows minimal changes (i.e., values close to zero) in RD-rate and BD-PSNR at 0.0013dB and 0.0039% respectively. As we can see that in IV the average change in encoding time (ΔT) of different video sequences have a gain ranging from 0.6-1.5%.

Theoretically, the compression outcome of two methods will be relatively similar to each other. There are few reasons that cause the result have a minor changes between both proposed method and original method:

1) The CPU requires higher computational demand to process live stream framework, while pre-recorded video method is direct compressed by the encoder without going through the live streaming process.
2) The live stream video input could be start up at anytime,

Table III: BD–PSNR and BD–Rate of proposed method against MV-HEVC and simulcast

	Proposed vs MV-HEVC				Proposed vs Simulcast			
	I-Frame (Bit 0)		I-Frame (Bit 1)		I-Frame (Bit 0)		I-Frame (Bit 1)	
	BD-PSNR	BD-rate	BD-PSNR	BD-rate	BD-PSNR	BD-rate	BD-PSNR	BD-rate
Product	-0.2195	0.0091	0.2065	-0.0087	0.1584	-0.0061	0.1960	-0.0075
Filler	0.0668	0.0014	0.3326	-0.0069	-0.0039	-0.0013	0.4383	-0.0081
Balloon	0.1384	-0.0049	0.1759	-0.0056	0.1183	-0.0029	0.1664	-0.0060
Kendo	0.0938	-0.0021	0.3658	-0.0103	0.1920	-0.0056	0.3543	-0.0105
Average	0.0199	0.0009	0.2702	-0.0078	0.1162	-0.0040	0.2888	-0.0080

Table IV: Change in encoding time compared to our proposed method

Video	Average ΔT, %			
	MV-HEVC		Simulcast	
	I-Frame (Bit 0)	I-Frame (Bit 1)	I-Frame (Bit 0)	I-Frame (Bit 1)
Product	0.3008	0.3008	0.2511	0.1995
Filler	0.4203	0.6223	1.5191	1.7792
Balloon	0.8472	2.1122	3.8839	2.8974
Kendo	0.8469	1.0765	0.2439	0.4374
Average	0.6038	1.0279	1.4745	1.3284

thus random GOP by instant video capturing will be encoded in the live streaming model compare to a pre-recorded model. Therefore, the result of intra-prediction of the starting frame "I frame" in a GOP is not same with the first frame predicted on pre-recorded video compression.

The capturing scenes ("Product" and "Filler") are chosen from repeating processes in a loop system to maintain the consistency of captured video content. Therefore, the average mean of instant video capturing and compression method can be validated to the pre-recorded video compression method.

V. CONCLUSION AND FUTURE WORK

This project presented a live multi-view video steaming on industrial machine scene based modified MV-HEVC codec prototype with data embedding capability. Collected results indicate that our proposed approach requires an additional of maximum 0.0105% in bandwidth and 0.4383dB difference in video quality in the BD-rate and BD-PSNR scales validated to pre-recorded video method. In future work, we would explore more advantageous of data embedding application, e.g., watermarking and authentication, as well as exploring the data embedding capacity on MV-HEVC codec by using current prototyping work.

VI. ACKNOWLEDGMENTS

The authors would like to acknowledge the project members: Prof. Dr. Lim Tong Ming, Assoc. Prof. Dr. Lee Wah Pheng, Assoc. Prof. Dr. Wong Kok Sheik, facility providers from the Computational Intelligence Research Laboratory, Faculty of Computing and Information Technology, Tunku Abdul Rahman University College and Ministry of Education Malaysia (MOE) for the financial support of this research. This research is supported MOE under the Fundamental Research Grant Scheme (FRGS) with project code: FRGS/1/2018/ICT01/TARUC/02/1.

REFERENCES

[1] CISCO, "Cisco annual internet report (2018-2023) white paper," https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html, 2020, Last accessed 14 Jul 2021.

[2] Dong-Wook Kim, Young-Geun Choi, Hwa-Sung Kim, Ji-Sang Yoo, Hyun-Jun Choi, and Young-Ho Seo, "The problems in digital watermarking into intra-frames of H.264/AVC," *Image and Vision Computing*, vol. 28, no. 8, pp. 1220–1228, 2010.

[3] Sagar Gujjunoori and B.B. Amberker, "DCT based reversible data embedding for MPEG-4 video using HVS characteristics," *Journal of Information Security and Applications*, vol. 18, no. 4, pp. 157–166, 2013.

[4] Rangding Wang Qian Li, "Watermarking in H.264/AVC compressed domain using CAVLC," *Journal of Computing*, vol. 8, no. 11, pp. 3126–3133, 2013.

[5] Yiqi Tew and KokSheik Wong, "An overview of information hiding in h.264/avc compressed video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 2, pp. 305–319, 2014.

[6] Bo Guan, Dawen Xu, and Qian Li, "An efficient commutative encryption and data hiding scheme for hevc video," *IEEE Access*, vol. 8, pp. 60232–60245, 2020.

[7] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[8] G. Tech, Y. Chen, K. Müller, J. Ohm, A. Vetro, and Y. Wang, "Overview of the multiview and 3d extensions of high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 35–49, Jan 2016.

[9] A. J. Diaz-Honrubia, J. De Praeter, S. Van Leuven, J. De Cock, J. L. Martinez, and P. Cuenca, "Using bayesian classifiers for low complexity multiview h.264/avc and hevc hybrid architecture," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2015, pp. 1–6.

[10] BruhanthFOP Mallik, Akbar Sheikh-Akbari, and Ah-Lian Kor, "Mixed-resolution hevc based multiview video codec for low bitrate transmission," *Multimedia Tools and Applications*, vol. 78, no. 6, pp. 6701–6720, Mar 2019.

[11] C. Jiang and S. Nooshabadi, "Fully pipelined real time hardware solution for high efficiency video coding (hevc) intra prediction," *J Real-Time Image Proc*, vol. 17, pp. 197–213, 2020, Multi-level complexity reduction for HEVC multiview coding.

[12] Yiqi Tew and KokSheik Wong, "Information hiding in hevc standard using adaptive coding block size decision," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 5502–5506.

[13] Dominic Springer, Wolfgang Schnurrer, Andreas Weinlich, Andreas Heindel, Jürgen Seiler, and André Kaup, "Open source hevc analyzer for rapid prototyping (harp)," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 2189–2191.

[14] Suramya Tomar, "Converting video formats with FFmpeg," *Linux Journal*, vol. 2006, no. 146, pp. 10, 2006.

[15] Toyohiro Saito, "Nagoya university multi-view sequences download list," https://www.fujii.nuee.nagoya-u.ac.jp/multiview-data/, 2021, Last accessed 14 Jul 2021.

[16] Richard T. Saunders, "Picklingtools: Cross language tools for communication," http://www.picklingtools.com/home, 2009, Last accessed 14 Jul 2021.

[17] OTTVerse, "Bd-rate and bd-psnr: Calculation and interpretation," https://ottverse.com/what-is-bd-rate-bd-psnr-calculation-interpretation/, 2021, Last accessed 14 Jul 2021.