Journal of Multimedia Information System VOL. 5, NO. 3, September 2018 (pp. 201-208)
http://dx.doi.org/10.9717/JMIS.2018.5.3.201

Multi-Sever based Distributed Coding based on HEVC/H.265 for Studio Quality Video Editing

Jongho Kim[1,*], Sung-Chang Lim[1], Se-Yoon Jeong[1], Hui-Yong Kim[1]

Abstract

High Efficiency Video Coding range extensions (HEVC RExt) is a kind of extension model of HEVC. HEVC RExt was specially designed for dealing the high quality images. HEVC RExt is very essential for studio editing which handle the very high quality and various type of images. There are some problems to dealing these massive data in studio editing. One of the most important procedure is re-encoding and decoding procedure during the editing. Various codecs are widely used for studio data editing. But most of the codecs have common problems to dealing the massive data in studio editing. First, the re-encoding and decoding processes are frequently occurred during the studio data editing and it brings enormous time-consuming and video quality loss. This paper, we suggest new video coding structure for the efficient studio video editing. The coding structure which is called "ultra-low delay (ULD)". It has the very simple and low-delayed referencing structure. To simplify the referencing structure, we can minimize the number of the frames which need decoding and re-encoding process. It also prevents the quality degradation caused by the frequent re-encoding. Various fast coding algorithms are also proposed for efficient editing such as tool-level optimization, multi-serve based distributed coding and SIMD (Single instruction, multiple data) based parallel processing. It can reduce the enormous computational complexity during the editing procedure. The proposed method shows 9500 times faster coding speed with negligible loss of quality. The proposed method also shows better coding gain compare to "intra only" structure. We can confirm that the proposed method can solve the existing problems of the studio video editing efficiently.

Key Words: HEVC, Distributed coding, Parallel coding, Studio editing.

I. INTRODUCTION

HEVC is the most recent video coding standard which was established in January 2013[1]. It was developed by Joint Collaborative Team on Video Coding (JCT-VC) formed by the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Pictures Experts Group (MPEG). Many proposals to improve the coding efficiency were submitted and various coding tools based on proposal were adopted. HEVC RExt called second version of HEVC was published in July, 2014. The main purpose of HEVC RExt standardization was making the video coding standard to dealing the high quality video data. By this reason, it can support the various color sampling formats and bit-depths compare to HEVC version 1. HEVC RExt is suitable for dealing the high quality and massive data such as medical images or studio images.

In this paper, efficient studio video data editing method using HEVC RExt is proposed. To adopt the HEVC RExt in the studio video editing, the following points should be considered [2].

1) *Low-delay coding and decoding (1 frame or less)*
2) *Low-loss compression providing visually near-perfect reproduction.*
3) *Multi-generation compression adds negligible concatenation errors and additional loss.*
4) *A symmetrical encode/decode algorithm that should be relatively easy to implement both in hardware and software*
5) *Support for full range of images (image sampling, resolution, frame-rate, bit depth and color gamut)*
6) *Compression range is from 2 to 20 times.*
7) *Similarly, such a codec can also be used to decrease file sizes to improve storage efficiency and download times during production.*

The clause 2) to 7) are the basic function of the video codec to dealing high quality data. HEVC RExt was designed for satisfying these requirements in the standardization stage. The clause 1) is the main feature which can reveal the characteristics of the studio codec. If the coding delay is increased in the studio editing, the efficiency of the editing process might be decreased. Most of video codec support low-delay coding structure and one of the typical structure is the "intra only" structure. In case of "intra only" coding structure, it supports frame level editing without any decoding process. Because referencing frames are not necessary in "intra only" coding structure.

Manuscript received August 20, 2018; September 10, 2018; Accepted September 13, 2018. (ID No. JMIS-2018-0041)

Corresponding Author (*): Jongho Kim, 218 Gajeong-ro, Yuseong-gu, Daejeon, 34129, KOREA, pooney@etri.re.kr

[1] Realistic AV research group, Electronics and Telecommunications Research Institute, Daejeon, Republic of Korea,

As is well known, some tradeoffs are existed between coding efficiency and delay. When the number of reference frame for current frame is increasing, the coding efficiency is also increasing in most cases. In that case, the delay for editing current frame is also increasing due to reference frames decoding. It is shown in Fig. 1[3].

Fig. 1 is the example of the general low-delay (LD) coding structure in HEVC common test condition. Although the reference frames for picture #6 are two, the all pictures before picture #6 should be decoded for editing. It is against the above clause 1) and it also brings quite big inconvenience during studio editing work.

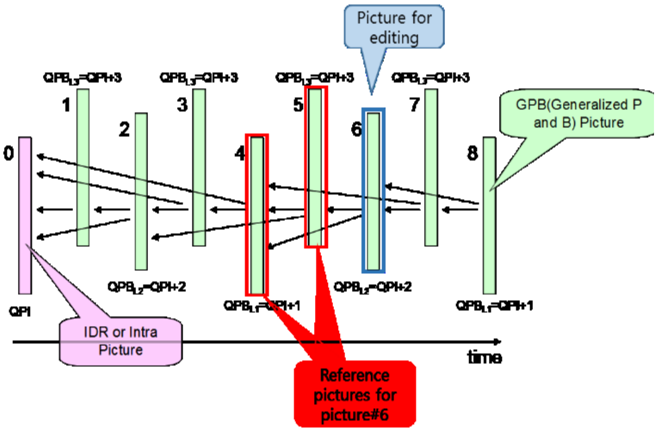

Fig. 1. Presentation of low-delay coding structure.

One of the main motivation of this paper is making the new coding structure which minimizing the unnecessary referencing structure to reduce the editing delay and re-encoding. Even though the reference structure is simplified, the proposed method would maintain the coding gain of the reference structure. In this paper, we propose new low-delay coding structure which includes referencing structure without any violation of mentioned clauses. The proposed coding structure satisfy the clause 1) and it also shows better coding performance compare to "intra only" coding structure.

The detail descriptions of the proposed coding structure are described in section 2-A.

Many coding tools were adopted during the standardization. The computational complexity was also increased. Other main motivation of the proposed method is reducing the computational complexity of decoding and re-encoding process during the editing. To reduce the computational complexity, many efficient reports were published [4-8]. Kim et al, propose fast intra prediction method based on SATD (sum of absolute transformed difference) cost. They determine the available prediction modes by the SATD cost of some pre-defined prediction mode. Morta et al, also propose the intra mode decision algorithm. The proposed method decides the candidate prediction modes using the direction information of neighboring blocks. The inter prediction complexity reduction method is proposed by Kim et al. They determine the weather conduct the bi-prediction or not by the SAD (sum of absolute difference) cost of the largest block. The SIMD based fast coding method is proposed by Jeon et al. The main coding functions such as transform, intra prediction, motion estimation/compensation are implemented by SIMD. Most of recent encoding optimization methods are based on fast algorithms. The fast coding approaches based on parallel coding methods using multi-core and multi-server are very few.

In this paper, we also consider many fast coding methods to reduce the computation complexity. First, we remove the coding tools which have heavy complexity and little coding gain. Second, various parallel processing methods are adopted such as frame/tile-level parallel processing and the parallel processing using SIMD. Last, the distributed coding is applied based on multi-severs.

The remainder of this paper is organized as follows: In section II, we explain the proposed ultra-low delay coding structure. The proposed fast coding approaches are described in section III. Finally, experimental results and some conclusions are given in sections IV and V.

II. PROPOSED VIDEO CODING STRUCTURE FOR EFFICIENT EDITING

In this section, we explain the new coding structure which is called "ultra-low delay (ULD)" in this paper. As we mentioned it in the previous section. The main problem of the studio video editing work is the editing delay caused by complex referencing structure. It also brings frequent decoding and re-encoding. In proposed structure, using the IDR picture as the only reference picture, the only one frame delay is need for editing in one GOP (group of pictures). We also bring the coding gain by maintaining the temporal referencing structure compare to "intra only" structure.

A. Propose ultra-low delay (ULD) coding structure

The basic purposed of the proposed coding structure is both satisfying the clause 1) and accommodating the advantage of referencing structure.

The proposed coding structure is basically following the LD coding structure of HEVC common test condition. The main differences compare to low-delay structure are described as follow.

1) *While the LD structure is consisting of four hierarchies, the ULD structure supports just two hierarchies.*
2) *Only previous coded intra picture is used for reference picture of other following pictures within the same intra period.*

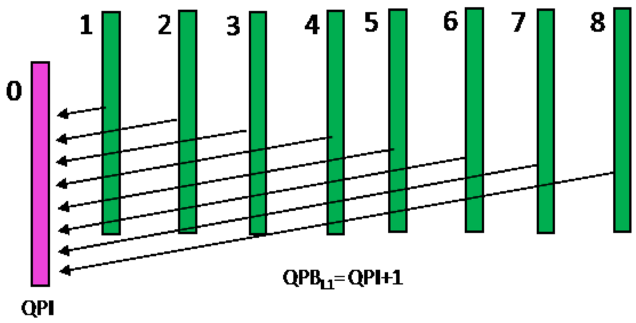

Fig. 2. Presentation of ultra-low delay coding structure.

The quantization parameter (QP) of LD structure is charged by corresponding hierarchy of layer. The base QP is the QP of intra picture (QPI). When hierarchy of layer is increased, the QP is also increased form QPI+1 to QPI+3. ULD structure support only two hierarchy. Then the QP of non-intra pictures is determined by QPI +1. The non-intra pictures of ULD are non-referencing picture and it refers just previous intra picture in same group of pictures (GOP). The advantage of ULD structure is from above mentioned referencing structure. Decoding just one or less picture is need for the accessing the picture that user want to edit. The coding efficiency of ULD is also better than "intra only" which is based on non-referencing structure.

Generally, the editing result is emitted after all editing procedures are finished. If the coding structure of the result is based on the referencing structure (such as LD), the re-encoding procedure is required. (Shown in Fig. 3)

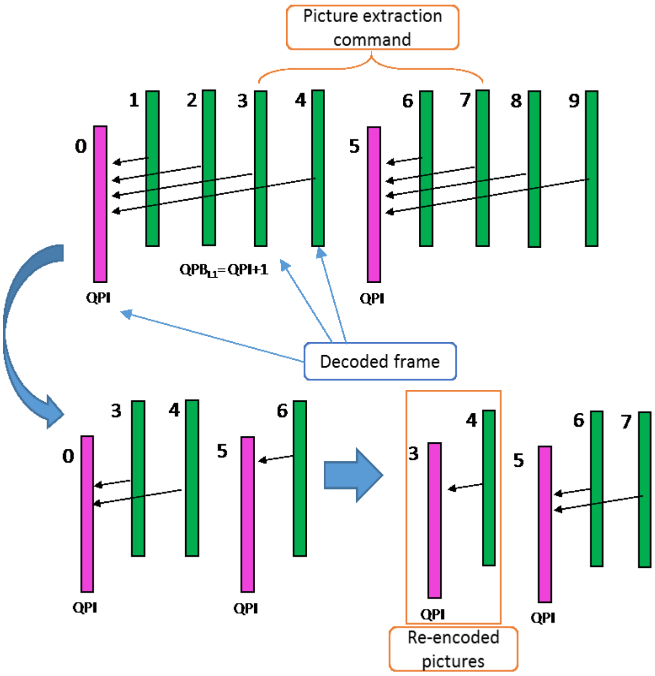

Fig. 3. Example of decoding and re-encoding procedure during the picture extraction.

In case of Fig. 3, the picture extraction is occurred between picture #2 and picture #8. In ULD case, three picture are decoded and only two pictures are re-encoded. In LD case, all pictures before picture #8 should be decoded and re-encoded. Reducing the number of pictures which need decoding and re-encoding procedure is the another main advantage of the ULD.

III. PROPOSED FAST CODING METHODS

In this section, the fast coding methods are proposed to reduce the computation complexity of re-encoding process during the editing. The main idea of proposed method is the parallel processing using the multi-core and distributed coding based on multi-server platform. By adopting the various fast coding techniques, 4K-UHD video can be coded in real time. It brings many conveniences in studio quality video editing.

Table 1. Overall complexity of main tools (YUV4:2:2, 10bit).

Tools	QP			
	22	27	32	37
Intra prediction	6.94	5.23	4.47	4.06
Inter prediction	58.2	68.01	74.82	77.69
Transform	3.98	3.53	3.31	3.10
Quantization	20.81	16.81	12.94	11.47
Entropy coding	7.90	4.40	2.91	2.03
Loop filter	0.17	0.22	0.23	0.25
Etc.	2.00	1.80	1.32	1.40

Table 2. Overall complexity of main tools (YUV4:4:4, 10bit).

Tools	QP			
	22	27	32	37
Intra prediction	6.94	5.23	4.47	4.06
Inter prediction	58.2	68.01	74.82	77.69
Transform	3.98	3.53	3.31	3.10
Quantization	20.81	16.81	12.94	11.47
Entropy coding	7.90	4.40	2.91	2.03
Loop filter	0.17	0.22	0.23	0.25
Etc.	2.00	1.80	1.32	1.40

A. Complexity analysis of ULD coding structure

As we mentioned above section, the low-delay coding structure is the essential for editing the studio video. The random access is also essential requirement. So the IDR frame is inserted every 0.1 second in this paper. The complexity analysis of HEVC RExt is conducted under JCT-VC low delay common test condition. The complexity of main tools is described in Table 1 & 2

The complexity of inter prediction which includes motion estimation/compensation and interpolation is the biggest. The second is quantization procedure. Main complexity of quantization is caused by rate-distortion optimized quantization.

B. Tool-level optimization

In this paper, the tool-level optimization is adopted. Tool-level optimization is determination procedure that which tools are used or not by the tradeoff between complexity and coding gain. We calculated the complexity and coding gain of each tools. Then on or off decision of each tools is determined and it is described in Table 3.

Table 3. Tool-level On/Off decision of each tools.

Tools	On/Off
Tile	16 tiles per one frame
In-loop Filter Across Tile Boundary	On
WPP	Off
Multi-slice	Off
Transform Skip	Off
AMP	Off
RDO-Q	Off
Deblocking Filter	On
SAO	On
PCM	On
Scaling List	Off (Flat)
Lossless Coding	Off
Sign Bit Hiding	Off
Extended Precision Processing	Off
Intra Block Copy	Off
Implicit(Intra) Residual DPCM	Off
Explicit(Inter) Residual DPCM	Off
Residual Rotation	Off
Large-block Transform Skip	Off
Single Sig. Map Context (for Transform Skip)	Off

C. Parallel coding structure based on ULD

In this paper, the parallel encoding method is proposed to accelerate re-encoding process during the editing work. The proposed parallel processing is consisted of two levels. First is the tile-level parallel processing. The picture is divided into several blocks which is called "tiles" then each tiles are coded at the same time. The other one is the picture level parallel processing. The non-intra pictures which refer same intra picture can be encoded in parallel. Because in ULD, there is no referencing relation between non-intra pictures. The details are shown in Fig. 4. After intra picture is coded, the following non-intra pictures which refer previous coded intra picture are encoded in parallel. Non-intra pictures in same group of picture are coded at the same time. Also, the tiles in the picture are also coded in parallel. The 16 tiles per picture are coded in parallel. These parallel coding concepts also can be adopted in decoding procedure as well as re-encoding procedure.

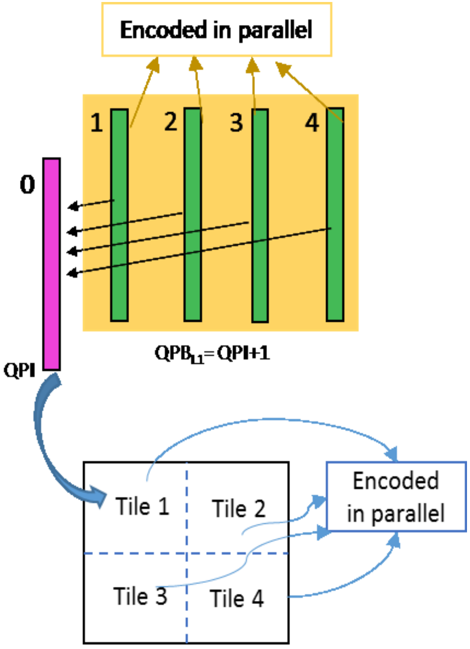

Fig. 4. Picture and tile level parallel processing.

Table 4. Overall performance of the proposed method (YUV4:2:2).

Color Sampling Format	Bit Depth	Sequence (Resolution)	BD-Rate (%)	Time Saving (%)
4:2:2	10 bit	Traffic (2560x1600)	-36.28	55.86
		Kimono1 (1920x1080)	6.49	45.57
		[EBU] Horse (1920x1080)	-33.96	66.42
		[EBU] Graphics (1920x1080)	-75.55	80.16
		[EBU] WaterRocksClose (1920x1080)	2.35	57.21
		[EBU] KidsSoccer (1920x1080)	-25.84	56.40
		Seeking (1920x1080)	-8.98	56.14
Average			-24.54	59.68

The overall performance of the proposed coding structure and parallel processing is described in Table 4 & 5. The test is conducted on HEVC test model (HM) version

15.0+RExt-8.1, which is set as the anchor for comparison test. To evaluate the coding performance of the proposed coding structure is compared with “intra only” configuration which is one of the common test condition in JCT-VC.

The bit depth of test sequences are 10bit and test sequences are categorized into 2 category by color sampling format. In case of 4:2:2, the average BD-rate and time saving were 24.54% and 59.68%, respectably. The average decrement in BD-rate and time was 18.33% and 56.14% in 4:4:4 sequences. Even though the BD-rate of some sequences were increased, it shows meaningful time reduction for these sequences.

Table 5. Overall performance of the proposed method (YUV4:4:4).

Color Sampling Format	Bit Depth	Sequence (Resolution)	BD-Rate (%)	Time Saving (%)
4:4:4	10 bit	Traffic (2560x1600)	-28.00	49.87
		Kimono1 (1920x1080)	7.63	36.84
		[EBU] LupoCandlelight (1920x1080)	-28.55	69.74
		[EBU] RainFruits (1920x1080)	-44.97	70.62
		VenueVu (1920x1080)	-17.43	45.27
		BirdsInCage (1920x1080)	-10.36	67.85
		CrowdRun (1920x1080)	-6.69	50.38
Average			-18.33	56.14

D. Parallel coding based on SIMD

Single instruction, multiple data (SIMD) is a class of parallel computers in Flynn's taxonomy. It describes computers with multiple processing elements that perform the same operation on multiple data points simultaneously. Such machines exploit data level parallelism, but not concurrency. Easily, it is a kind of parallel processor which can perform the multiple calculations with one instruction Generally, SIMD operator can deal the 128 bits data operation in once. So if the data type is ‘short’, SIMD operator can handle the 8 operation at a time. The best advantage of SIMD operation is that it can reduce the computational complexity effectively without any loss of quality. The implementation history is shown in Table 6.

Table 6. The implementation history of SIMD.

Tools	Main function in HM	Implementation history
Quantization	✓ xQuant	✓ Quantization level calculation ✓ SIMD operation level: 32bit
De-quantization	✓ xDeQuant	✓ Inverse quantization coefficient Clipping ✓ SIMD operation level: 16bit
Hadamard Transform	✓ xCalcHADs4x4 ✓ xCalcHADs8x8	✓ 4x4/8x8 Hadamard transform ✓ SIMD operation level: 16bit
Interpolation	✓ filterVer ✓ filterHor ✓ filterCopy ✓ filter	✓ Block level luma/chroma interpolation ✓ SIMD operation level: 16bit
Intra Prediction	✓ xPredIntraAng ✓ predIntraGetPredValDC ✓ xDCPredFiltering	✓ Angular/DC prediction ✓ SIMD operation level: 16bit
	✓ xPredIntraPlanar	✓ Planar prediction ✓ SIMD operation level: 32bit
Intra Reconstruction	✓ xIntraCodingTUBlock	✓ prediction signal Clipping ✓ SIMD operation level: 16bit
Intra Residual Coding	✓ xIntraCodingTUBlock	✓ Residual signal calculation ✓ SIMD operation level: 16bit
Remove High Frequency	✓ removeHighFreq	✓ SIMD operation level: 16bit
SAD (Sum of Absolute Differences)	✓ xGetSAD8 ✓ xGetSAD16 ✓ xGetSAD32 ✓ xGetSAD64	✓ SAD calculation ✓ SIMD operation level: 16bit
SSE (Sum of Squared Errors)	✓ xGetSSE4 ✓ xGetSSE8 ✓ xGetSSE16 ✓ xGetSSE32 ✓ xGetSSE64	✓ SSE calculation ✓ SIMD operation level: 16bit
Transform	✓ fastForwardDst ✓ fastInverseDst ✓ partialButterfly4/partialButterflyInverse4 ✓ partialButterfly8/partialButterflyInverse8	✓ 4x4 DST, 4x4/8x8 DCT ✓ SIMD operation level: 16bit
	✓ partialButterfly16/partialButterflyInverse16 ✓ partialButterfly32/partialButterflyInverse32	✓ 8x8/16x16/32x32 DCT ✓ SIMD operation level: 32bit
Add/Subtract /Average function	✓ addClip ✓ Subtract ✓ addAvg	✓ Add/Subtract/Average/Clipping related function ✓ SIMD operation level: 16bit
Entropy Coding	✓ countNonZeroCoeffs ✓ codeCoeffNxN	✓ Non-zero coefficient scanning and counting ✓ SIMD operation level: 16bit
Signal Input	✓ readPlane	✓ Reading the input signal ✓ SIMD operation level: 32bit

The overall performance of SIMD operation is shown in Table 7 & 8. The test condition is same as that previously mentioned.

In case of YUV 4:2:2 10 bit sequences, 31.36% of overall encoding time is reduced in average without any loss of quality.

Table 7. Overall result of SIMD implementation (YUV4:2:2, 10bit).

Sequence (4:2:2, 10bit)	QP	Time Saving
Traffic (2560x1600)	22	34.96%
	27	39.12%
	32	41.68%
	37	43.19%
Kimono1 (1920x1080)	22	35.04%
	27	40.27%
	32	42.74%
	37	38.00%
[EBU] Horse (1920x1080)	22	30.08%
	27	34.64%
	32	37.05%
	37	38.04%
[EBU] Graphics (1920x1080)	22	26.99%
	27	30.28%
	32	34.93%
	37	38.17%
[EBU] WaterRocksClose (1920x1080)	22	21.47%
	27	31.21%
	32	35.95%
	37	37.77%
[EBU] KidsSoccer (1920x1080)	22	16.02%
	27	19.96%
	32	23.94%
	37	24.79%
Seeking (1920x1080)	22	12.37%
	27	18.84%
	32	24.79%
	37	25.83%
Average		31.36%

In case of YUV 4:4:4 10 bit sequences, 38.81% of overall encoding time is reduced in average without any loss of quality. The complexity reduction tendency according to QP variation is similar to each test set.

Table 8. Overall result of SIMD implementation (YUV4:4:4, 10bit).

Sequence (4:4:4, 10bit)	QP	Time Saving
Traffic (2560x1600)	22	34.16%
	27	38.51%
	32	42.37%
	37	42.32%
Kimono1 (1920x1080)	22	31.50%
	27	38.41%
	32	42.15%
	37	42.02%
[EBU] Horse (1920x1080)	22	37.62%
	27	40.97%
	32	43.26%
	37	44.39%
[EBU] Graphics (1920x1080)	22	33.10%
	27	39.28%
	32	41.73%
	37	41.81%
[EBU] WaterRocksClose (1920x1080)	22	36.32%
	27	37.34%
	32	40.28%
	37	41.57%
[EBU] KidsSoccer (1920x1080)	22	31.82%
	27	43.43%
	32	42.58%
	37	42.55%
Seeking (1920x1080)	22	28.16%
	27	31.67%
	32	37.24%
	37	40.09%
Average		38.81

E. Multi-Server based Distributed Coding using MPI

In this paper, we propose the multi-server based distributed coding using MPI (Message Passing Interface) protocol. 16 servers are used for distributed encoding. Input sequence is divided into random access unit. 10 random access units are coded in one server in parallel. The overall parallel structure of proposed method is shown in Fig. 5.

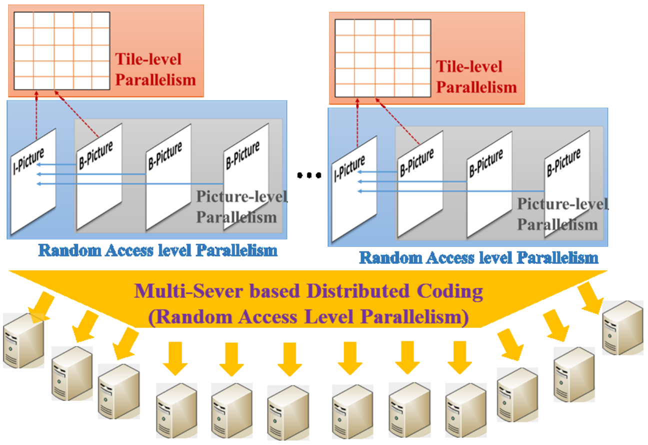
Tile-level Parallelism
I-Picture
B-Picture
B-Picture
B-Picture
Picture-level Parallelism
Random Access level Parallelism
…
Tile-level Parallelism
I-Picture
B-Picture
B-Picture
B-Picture
Picture-level Parallelism
Random Access level Parallelism
Multi-Sever based Distributed Coding
(Random Access Level Parallelism)

Figure 5. Overall Parallel Structure

IV. EXPERIMETAL RESULTS

The proposed coding structure is adopted and tested on HEVC test model (HM) version 15.0+RExt-8.1, which is set as the anchor for comparison test.

To evaluate the coding performance of the proposed coding structure is compared with "intra only" configuration which is one of the common test condition in JCT-VC [9]. The quantization parameter (QP) range which is defined in main tier is used for the experiment. The Bjøntegaard delta bit rate (BD-rate) [10] and time saving was used for the performance comparison measure. The proposed algorithm fully guarantee real-time encoding in Full-HD sequences. So, the UHD (Ultra High Definition) sequences are used in final evaluation. The overall results are shown in Table 8.

Table 8. Overall result of proposed method (YUV4:2:2, 10bit)

Sequence (Resolution)	BD-Rate (%)	Intra Only		Proposed ULD	
		Bitrate (Mbps)	Encoding speed (fps)	Bitrate (Mbps)	Encoding speed (fps)
[EBU] Studio_dancer (3840x2160)	-7.47	258	0.0087	190	90.09
[EBU] FountainLady (3840x2160)	-7.40	170	0.0093	226	80.97
[EBU] LupoConfetti (3840x2160)	-8.20	234	0.0086	184	85.96
[EBU] RainFruits (3840x2160)	-7.70	229	0.0091	201	82.99
Average	-7.69	222	0.0089	200	85

General real-time broadcasting encoder cover under 50Mbps data rate for encoding the 4K-UHD video. The coded bit rate of test sequences is average 200Mbps and PSNR is approximately from 42dB to 50dB. It is quite high quality studio quality data compare to broadcasting data. Even though the date rates are quite big, the average encoding speed of proposed method is 85fps. It means that the proposed method can encode the UHD sequence as faster than real time. The proposed method shows approximately 9500 times faster than compare to intra only structure. The BD-rate of the proposed methods is average -7.69% compare to "intra only" structure. It means that proposed method shows better compression performance than "intra only" structure with a similar video quality.

V. CONCLUSION

The new coding structure was proposed for efficient studio data editing. The proposed ULD structure can minimize the number of re-encoding picture. The coding time also can be reduced greatly by picture/tile level parallel processing and multi-server based distributed coding. In view of coding efficiency, the proposed method can reduce average 7.69% in BD-rate with 9500 times faster coding speed compare to "intra only" coding structure.

Acknowledgement

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00072, Development of Audio/Video Coding and Light Field Media Fundamental Technologies for Ultra Realistic Tera-media)

REFERENCES

[1] ISO/IEC JTC 1 SC29 WG11, "Joint Call for Proposals on Video Compression Technology," Doc. N11113, Jan. 2010.

[2] R.T. Russell, "Mezzanine Compression for HDTV", BBC R&D White Paper, WHP119, Sep. 2005.

[3] Ken et al, "High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Improved Encoder Description", JCTVC-S1002, Oct. 2014.

[4] Y Kim, DS Jun, S Jung, JS Choi, J Kim "A Fast Intra-Prediction Method in HEVC Using Rate-Distortion Estimation Based on Hadamard Transform, " ETRI Journal, vol 35, no 2, pp270-280 April., 2013.

[5] Y Kim, DS Jun, S Jung, JS Choi, J Kim "A Fast Intra-Prediction Method in HEVC Using Rate-Distortion Estimation Based on Hadamard Transform, " ETRI Journal, vol 35, no 2, pp270-280 April., 2013.

[6] Motra, A.S. Gupta, A. ; Shukla, M., Bansal, P., Bansal, V, "Fast intra mode decision for HEVC video encoder," International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Sept. 2012

[7] J. Kim, D. Jun, S. Jeong, S. Cho, J. S. Choi, J. Kim, and C. Ahn, "An SAD-Based Selective Bi-prediction Method for Fast Motion Estimation in High Efficiency Video Coding", ETRI Journal, vol. 34, no. 5, Oct. 2012, pp. 753-758.

[8] DS Jun et al., "Development of an ultra-HD HEVC encoder using SIMD implementation and fast encoding schemes for smart surveillance system", Journal of Supercomputing, July, 2016.

[9] C. Rosewarne, K. Sharman, M. Naccari, G. J. Sullivan, HEVC Range Extensions Test Model 7 Encoder Description JCTVC-Q1013, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), April. 2014.

[10] G. Bjøntgaard, "Calculation of Average PSNR Differences between RD-curves," ITU-T SG16 Q.6 VCEG, Doc. VCEG-M33, 2001.

Authors

Jongho Kim received BS degree from Control and Computer Engineering Department, Korea Maritime University in 2005 and he also received MS degree from University of Science and Technology (UST) in 2007. He has been a senior researcher at Electronics and Telecommunications Research Institute (ETRI) since 2016 in Daejeon, Korea. His research interests include video processing, video coding and machine learning based compression.

Sung-Chang Lim received his BS (with highest honor) and MS degrees in computer engineering from Sejong University, Korea, in 2006 and 2008, respectively. Since 2008, he has been a senior researcher in Broadcasting and Media Research Laboratory of Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea. His research interests include video coding, image processing.

.

Seyoon Jeong received the BS and MS in electronics engineering from Inha University, Korea, in 1995 and 1997, respectively. Since1996, he has been a senior member of research staff with ETRI, Korea, and he is also working toward the PhD in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Korea. His current research interests include video coding, video transmission, and UHDTV.

Hui Yong Kim received his BS and MS degrees and his PhD from Korea Advanced Institute of Science and Technology in 1994, 1998, and 2004, respectively. From 2003 to 2005, he was the leader of the Multimedia Research Team of AddPac Technology Co. Ltd. In 2005, he joined the Broadcasting and Telecommunications Media Research Laboratory of ETRI, and currently serves as the director of the Visual Media Research Section. From 2006 to 2010, he was also an affiliate professor at UST. From 2013 to 2014, he was a visiting scholar at the Media Communications Lab of USC. He made has many contributions to the development of international standards such as MPEG Multimedia Application Format (MAF) and JCT-VC High Efficiency Video Coding (HEVC), as an active technology contributor, editor, and ad-hoc group co-chair in many areas. His current research interests are in image and video signal processing and compression for realistic media applications such as UHD, 3D, VR, and HDR.