HEVC-Compliant Screen Content Transcoding Based on Mode Mapping and Fast Termination

Fanyi Duanmu[1], Meng Xu[3], Yao Wang[1] and Zhan Ma[2],
[1]New York University, Tandon School of Engineering, Brooklyn, NY 11201, USA;
[2]Nanjing University, 22 Hankou Road, Nanjing 210093, China;
[3]Ubilinx Technology, Inc., 2870 Zanker Road #110, San Jose, CA 95134, USA.

Abstract - In this paper, a novel screen content (SC) transcoding framework is presented to efficiently bridge High Efficiency Video Coding (HEVC) and its Screen Content Coding (SCC) extension and support the bitstream backward-compatibility over legacy HEVC devices. Based on the side information analysis of SCC bit-stream, fast mode and partition heuristics are designed to accurately determine the mapping between novel SCC modes and conventional HEVC modes. Compared with the trivial SCC-HEVC transcoding solution, the proposed framework achieves a 42% re-encoding complexity reduction over standard JCT-VC screen content sequences with only 0.52% negligible BD-Rate loss under All-Intra (AI) configuration.

Index Terms - High Efficiency Video Coding (HEVC), Screen Content Coding (SCC), Video Transcoding, Fast Mode Decision, Mode Mapping.

I. INTRODUCTION

Screen content (SC) videos have become popular in recent years due to the technological advances in mobile and cloud applications, such as cloud gaming, wireless display, remote desktop collaborations, virtual desktop interfacing, etc. Such emerging applications and market demands create an urgent need for more efficient compression and low-latency delivery of screen content videos. To address these challenges, the "Joint Collaborative Team on Video Coding" (JCT-VC), launched the standardization of SCC extension [1] in 2014 on top of the latest High Efficiency Video Coding (HEVC) standard and finalized this extension in 2016. The SCC Test Model (SCM) software achieves over 50% BD-Rate saving over HEVC Range Extension (RExt) [1] for typical computer-generated contents. This significant gain is achieved from 4 major coding tools beyond HEVC, i.e., "Intra Block Copy" (IBC) [2], "Palette Coding Mode" (PLT) [3], "Adaptive Motion Compensation Precision" (AMCP) [4] and "Adaptive Color Transform" (ACT) [5].

Recognizing the market need, many industrial companies were following HEVC-SCC extension during standardization and most likely may include these new coding technologies into their products. Therefore, how to efficiently bridge the existing HEVC and its SCC extension using video transcoding (VTC) technology becomes interesting and useful, especially during the phase when both bit-streams coexist.

Video transcoding is a mature technology to realize video adaptations. During the transcoding process, many properties may change, such as the video format, bitrate, frame rate, resolution, coding tools, etc. Besides, additional information may be inserted into bit-stream such as video watermarking, error resilience, and so on. From the literature, the conversion within the same video format (e.g., spatial scaling in H.264) is referred as "homogeneous transcoding" while the conversion between different formats (e.g., between H.264 and HEVC) is referred as "heterogeneous transcoding". In practice, a central transcoding server can be used to periodically examine the client's constraints (such as bandwidth, resolution, etc.) and accordingly tailor the suitable bit-streams.

Even though it is possible to use the "trivial" approach, which firstly decodes the source bit-stream and afterward completely re-encodes into target bit-stream, however, such approach proves inefficient from complexity perspective. A reasonable solution should maximally utilize the decoded side information from the source bit-stream to facilitate the re-encoding such that both coding performance is well-preserved while the re-encoding speed is significantly improved.

There are many pioneer works in VTC area. In [6], a video transcoding overview is presented. Spatial/temporal resolution reduction, DCT-domain down-conversion are both introduced. When HEVC is finalized in 2013, many VTC studies were redirected into "H.264-HEVC" transcoding area. For instance, Peixoto, et al. proposed a few machine learning and statistics study based frameworks (e.g., [7] [8], etc.) to improve HEVC re-encoding speed. In their papers, H.264 Macroblocks (MB) are mapped into HEVC coding units (CU) based on motion vector (MV) distribution through online or offline training. Incorporated with statistics-based fast termination criteria, the proposed solution could introduce >3x encoder speedup with only 4% BD-Rate loss compared with trivial transcoder. Diaz-Honrubia, et al. also proposed a few fast VTC schemes (e.g., [9], [10]) to exploit H.264/AVC decoded side information for HEVC CU partition decision based on a Naïve-Bayes (NB) classifier, specifically for CUs with size 32x32 and 64x64, whereas for smaller CUs, the proposed transcoder simply mimics H.264/AVC encoding behaviors. A speed-up of 2.5x is reported with only 5% BD-Rate penalty. In [11], a HEVC fast transcoder is proposed based on merged block prediction homogeneity. Residuals and MV consistencies are populated to represent the homogeneity of target region and decide CU partitions. Another similar work [12] proposed by Zheng, et al. uses residual mean absolute deviations (MAD) and sum of absolute residual (SAR) as the homogeneity indicator to early terminate CU partitions. A 57% complexity reduction is achieved with only 2.2% BD-Rate loss. In [13], another mode merging and mapping solution is presented using motion

978-1-5386-0462-5/17/$31.00 ©2017 IEEE. VCIP 2017, Dec. 10 – 13, 2017, St Petersburg, U.S.A.

vector variance and mode conditional probabilities to predict merge decisions of H.264 blocks. A 50% complexity reduction is reported with negligible BD-Rate loss.

Though there have been substantial prior research efforts in fast video transcoding, our previous work [14] is so far the only algorithm proposed for fast SC transcoding, addressing HEVC-SCC fast forward transcoding for bandwidth reduction consideration. In this work, we focus on the fast transcoding from SCC bit-stream into HEVC bit-stream for the backward compatibility over the legacy HEVC devices, as illustrated in Fig.1. We apply the mode mapping techniques to efficiently decide the HEVC re-encoding modes and partitions based on the decoded SCC bit-stream for Intra-frame coding. To our best knowledge, this is the first work addressing SCC-HEVC transcoding based on official SCM software release (which is most likely to be deployed into next-generation SC products).

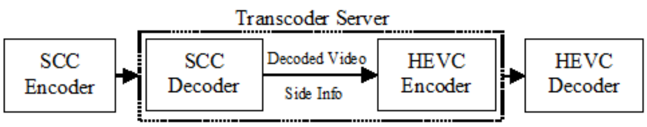

Figure 1. SCC-HEVC Transcoding Framework

The sequel of this paper is structured as follows. Section II briefly reviews SCM software architecture, new coding tools and major technical challenges for SCC-HEVC transcoding. In Section III, our proposed fast transcoding framework is presented. In Section IV, experimental results are provided and evaluated. This paper concludes in Section V with some future work summarized.

II. Screen Content Model (SCM) - Review

Screen Content Model (SCM) is the JCT-VC official test model for SCC extension development. Beyond HEVC, new tools are introduced to improve the coding efficiency.

A. SCM Block Scheme and New Coding Modes

SCM shares the same flexible quadtree block partitioning scheme as HEVC, which enables the use of Coding Units (CU), Prediction Units (PU) and Transform Units (TU) to adapt to diverse picture contents. CU is the basic unit for mode decision and is always in square shape. At encoder, the incoming pictures are divided into non-overlapping Coding Tree Units (CTU) of 64x64 pixels and each CTU can be divided into four equal-sized smaller CUs recursively, until the maximum depth is reached, as shown in Fig.2. At each CU, to determine the optimal encoding parameters (e.g.: partition, mode, etc.), an exhaustive search is employed by examining the Rate-Distortion (RD) cost using different coding modes and comparing the optimal RD cost at the current CU level against the sum of the RD costs of its sub-CUs (each using best mode and partition).

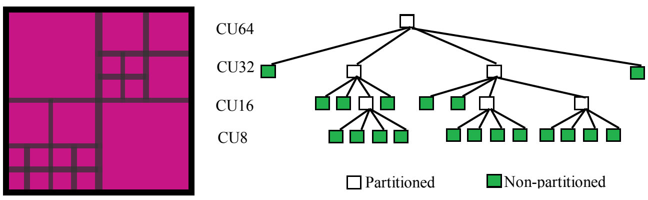

Figure 2. SCM CU Hierachitical Quadtree Partitioning Structure

SCM adopts two major coding tools to improve coding efficiency, known as Intra Block Copy (IBC) [2] and Palette (PLT) [3] mode. IBC mode is an Intra-frame block matching framework. To compress the current CU, the encoder will look back into the previously-coded area and find the best matching block. If chosen, a block vector (BV) is signalled to indicate the spatial offset between the matching block and the current CU. PLT coding mode encodes the current CU as a combination of a color table and a corresponding index map. The color table stores the representative colors. The original pixel block is translated into an index map indicating which color entry in the table is used for each pixel location.

B. SCC-HEVC Fast Transcoding Challenges

Different from HEVC Intra mode, SCC modes are highly dependent on the previous graphical patterns and image colors. Such "historical dependency" makes fast decision and mode mapping much more complicated and challenging. For example, in IBC blocks, depending on whether similar pattern appeared previously, the encoding RD costs of the same CU pattern but at different locations may vary significantly. Similarly, for PLT mode, two color tables are used. One is used for the current CU and the other one (also known as "palette predictor") is used as a dynamic look-up table storing the historical colors previously used. Depending on whether similar colors appeared and how frequent these colors are, the PLT mode RD costs of the same CU pattern but at different locations may also vary significantly. Furthermore, the novel SCC modes allow "inhomogeneous" CUs to be encoded in larger blocks without splitting. As shown in Fig.3, the 16x16 textual CUs in the top row are directly encoded using PLT mode (marked in green) without splitting into smaller 8x8 Intra CUs (marked in purple) in the bottom row. To conclude, due to the unique SC signal characteristics and the designs of PLT and IBC algorithms, existing mode mapping and fast splitting termination algorithms cannot be applied for SC transcoding directly. How to accurately map SCC modes into HEVC modes and efficiently determine the HEVC partition is a challenging problem, even for human judgement.

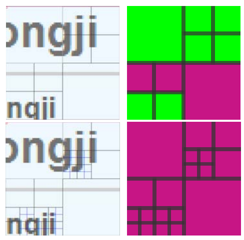

Figure 3. CTU Partition Decision Comparison between SCM and HEVC (Top: Text CTU coded by SCM; Bottom: Text CTU coded by HEVC)

III. Fast SCC-HEVC Transcoding System Using Mode Mapping and Fast Termination

Statistically, we have some important prior knowledge and observations about the HEVC and SCC, which are used to design our transcoding algorithm, as summarized below.

1. Over flat, smooth or directional blocks, HEVC and SCC can both use Intra-mode without further partitions. Therefore, the transcoder may copy the Intra mode from SCC bit-stream and directly apply to HEVC.

2. SCC new modes enable "inhomogeneous" blocks to be encoded in larger CUs, as shown in Fig.3. Compared with Intra-mode, PLT and IBC modes are mostly chosen at a larger CU size. Therefore, an intuitive yet safe transcoding heuristic is that the CU hierarchical depth in HEVC should be greater than the depth in SCC over the same block. For example, in Fig.3, the optimal coding depth for this block is 3 using SCC modes but 4 using HEVC Intra mode.

3. For PLT mode, the decoded index map provides valuable information regarding the CU structure and directionality, as illustrated in Fig.4. When the index map structure is purely flat, horizontal or vertical, during transcoding, the HEVC encoder can directly trigger the corresponding Intra sub-mode and terminate CU splitting. Otherwise, the encoder can bypass the current level Intra mode safely.

Figure 4. PLT Block and Index Map Illustration

4. For IBC mode, the decoded block vectors (BV) may be used to pinpoint the matching block in the previously-coded area. Since the matching block is the same as or similar to the current CU, therefore, our transcoder can copy or predict the mode and partition from the matching block along with its neighborhood, as shown in Fig.5. If the matching block is located precisely on the coding unit grid (i.e., the starting and ending positions of the matching block are multiples of 8, as shown in Fig.5 left), the current CU can copy the mode and the sub-CU partitions from the matching block. Otherwise (i.e., the matching block is across the grid boundaries, as shown in Fig.5 right), the current CU will examine the modes of the overlapping cells (i.e., inside the blue bounding-box). If all the cells share the same Intra-mode, then the current CU can directly inherit this shared Intra-mode and then immediately terminate CU splitting. Otherwise (i.e., the overlapping cells are coded using different Intra-modes), the current CU can bypass the current level Intra mode selection and proceed with the next level CU processing.

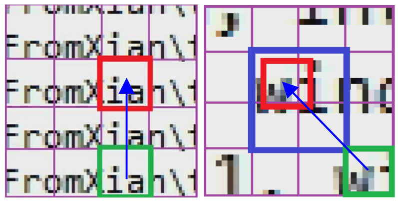

Figure 5. Block Vector Reuse for IBC-Intra Mode Mapping
Green Box: Current CU; Red Box: IBC Matching Block;
Blue Arrow: IBC Block Vector; Purple Lines: Coding Unit Grids

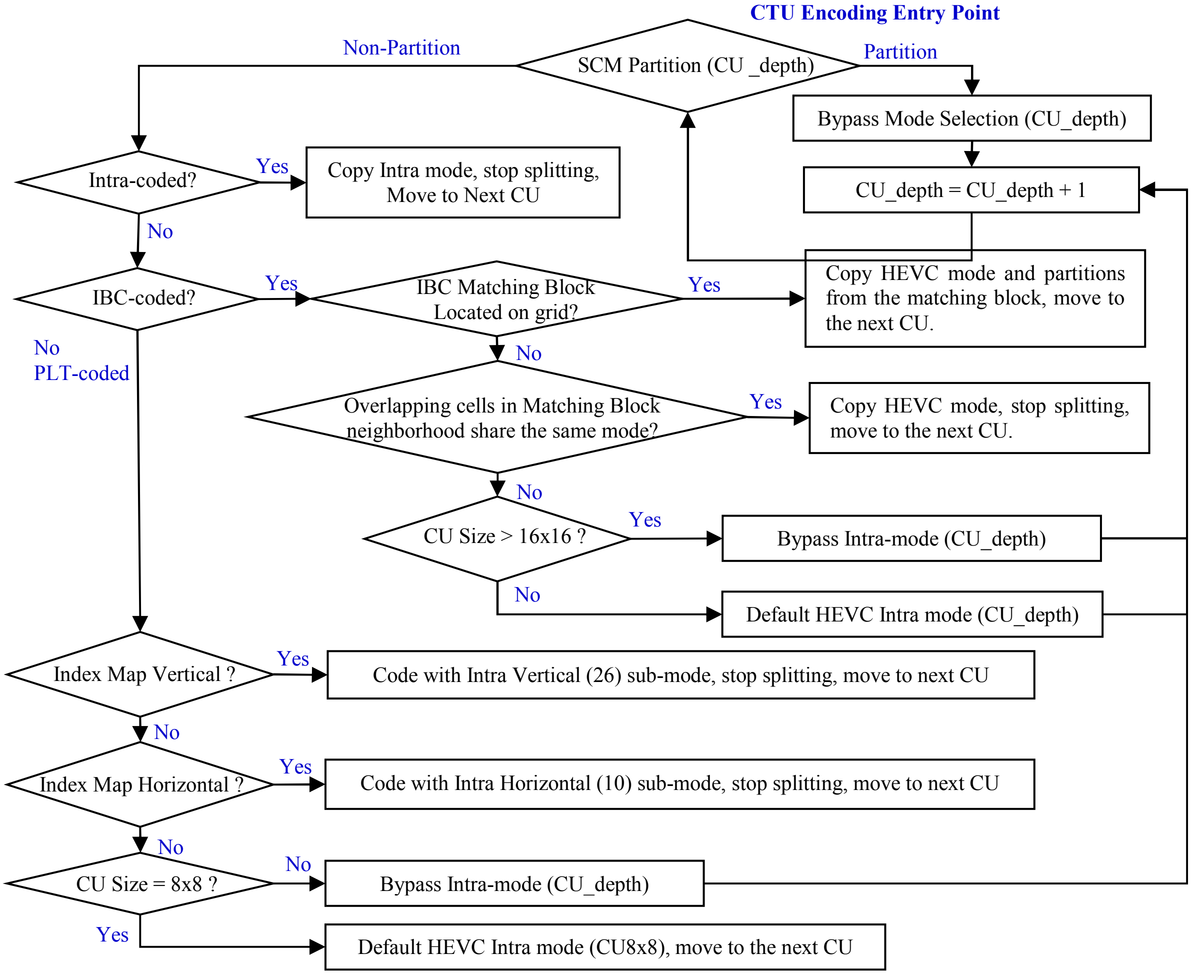

Figure 6. Proposed Fast SCC-HEVC Transcoding Algorithm

Based on such observations, our proposed fast transcoding system is implemented as shown in Fig.6. In this system, depending on the block coding mode, corresponding Intra-mode mapping algorithm is triggered. For Intra-coded blocks, our transcoder directly copies SCC Intra-mode and applies to HEVC. For IBC blocks, depending on the matching block location and its neighboring Intra mode distribution, our transcoder either directly copies HEVC mode and partition or directly bypass the current level Intra-mode processing. From our simulations, the Intra bypass at larger CU sizes (>16x16) introduces unnoticeable BD-Rate loss. For PLT blocks, based on the index map directionality, the transcoder triggers the corresponding Intra sub-mode. For simplicity, in this work, we only consider the two most dominant directions for fast termination (as investigated in our previous work [15] [16]). The typical textural or graphical PLT blocks need to be partitioned into very small Intra CUs to be homogeneous and therefore can be fast-bypassed safely.

IV. Experimental Results

Our proposed SCC-HEVC transcoding system is evaluated as follows: Firstly, 7 standard SCC sequences (with sample frames shown in Fig.7) are encoded using SCM-4.0 according to the common testing conditions (CTC) [17]. Then, SCC bit-stream is decoded with side information retrieved and stored. Finally, our proposed transcoding system will load the side information and re-encode the previously-decoded videos using the same QP values under All-Intra (AI) configuration.

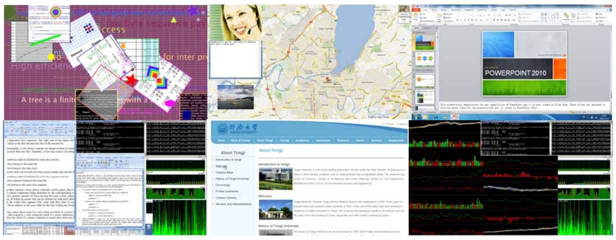

Figure 7. Sample Frames from JCT-VC Standard SCC Sequences

The system performance is evaluated using homogeneous Windows (64-bit) desktops with Intel-i5 CPU (2.67 GHz dual cores) and 4GB RAM. The re-encoding complexity reduction is directly measured using the encoding time. Compared with the trivial transcoding solution, our proposed scheme achieves a 42% re-encoding speed-up with only 0.52% BD-Rate [18] loss. Detailed simulation results are provided in Table I.

TABLE I. Proposed Transcoder Performance Evaluation

Sequence	QP	Anchor HM-16.4			Proposed			Performance	
		Rate	PSNR	Time	Rate	PSNR	Time	Rate	Time
FlyingGraphics 1080p YUV	22	153899	48.44	39	154731	48.40	25	+1.04%	-35%
	27	114004	44.09	36	115111	44.05	23		
	32	79039	39.80	33	79272	39.74	22		
	37	49486	35.75	30	49838	35.71	20		
Desktop 1080p YUV	22	186010	49.83	41	186330	49.82	24	+0.11%	-41%
	27	152153	44.99	39	152381	45.01	23		
	32	123079	40.18	36	123377	40.17	22		
	37	89295	35.25	33	89215	35.29	20		
Console 1080p YUV	22	84389	52.44	34	84507	52.39	22	+0.57%	-37%
	27	69466	47.47	33	69737	47.43	21		
	32	56139	42.78	31	56329	42.72	20		
	37	42810	37.52	29	42893	37.42	18		
WebBrowsing 720p YUV	22	24359	52.14	15	24409	52.05	7	+0.05%	-50%
	27	18636	47.31	14	18661	47.29	7		
	32	13198	43.35	13	13177	43.39	7		
	37	7237	37.78	12	7249	37.75	6		
Programming 72p YUV	22	58414	49.16	16	58454	49.11	9	+0.50%	-45%
	27	40124	45.03	15	40270	44.99	8		
	32	27507	41.13	14	27611	41.13	8		
	37	19391	37.63	13	19407	37.65	7		
SlideShow 720p YUV	22	10337	52.10	14	10388	52.21	7	+0.63%	-52%
	27	7480	48.22	13	7528	48.23	6		
	32	5304	43.99	12	5312	43.95	6		
	37	3499	39.92	11	3534	39.82	5		
MissionControl 1080p YUV	22	171936	49.56	37	172111	49.47	19	+0.77%	-46%
	27	128683	45.04	35	129128	45.03	18		
	32	90528	40.38	32	90903	40.30	17		
	37	57294	36.17	29	57588	36.13	17		

Rate: Average Per-frame coding rate (bits/second); PSNR: average PSNR across YUV-components (dB); Time: average per-frame coding time (second).

V. Conclusion

In this paper, a novel SCC-HEVC transcoding framework is presented based on mode mapping and fast termination. Based on the decoded side information, the proposed solution can efficiently and accurately determine the corresponding HEVC mode and partition. Compared with the trivial SCC-HEVC transcoder, the proposed solution achieves 42% re-encoding complexity reduction with only 0.52% negligible BD-Rate loss on average under All-Intra configuration. Future studies include Inter-frame SCC-HEVC transcoding algorithm design and Single-Input-Multiple-Output transcoding between HEVC and SCC bit-streams.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China projects (Grants #: 61371166, 61422107 and 61571215), in part by the National Science Foundation for Young Scholar of Jiangsu Province, China (Grant # BK20140610). Dr. Z. Ma is the corresponding author.

References

[1] G.-J. Sullivan, J. Boyce, Y. Chen, J.-R. Ohm, A. Segall, and A. Vetro, “Standardized Extensions of High Efficiency Video Coding (HEVC),” IEEE Journal of Selected Topics in Signal Processing, vol. 7, no. 6, pp. 1001 – 1016, Dec. 2013.

[2] C. Pang, J. Sole, L. Guo, M. Karczewicz, and R. Joshi, “Non-RCE3: Intra Motion Compensation with 2D MVs,” Doc. JCTVC-N0256, 2013.

[3] L. Guo, M. Karczewicz, and J. Sole, “RCE3: Results of Test 3.1 on Palette Mode for Screen Content Coding,” Doc. JCTVC-N0247, 2013.

[4] B. Li, J. Xu, G. J. Sullivan, Y. Zhou, B. Lin, “Adaptive motion vector resolution for screen content,” Doc. JCTVC-S0085, 2014.

[5] L. Zhang, J. Chen, J. Sole, M. Karczewicz, X. Xiu, Y. He, Y. Ye, “SCCE5 Test 3.2.1: In-loop Color-Space Transform,” Doc. JCTVC-R0147, 2014.

[6] A. Vetro, C. Christopoulos, and H.Sun, “Video Transcoding Architectures and Techniques: An Overview”, IEEE Signal Processing Magazine, vol. 20, no. 2, pp.18-29. March 2003.

[7] E. Peixoto, B. Macchiavello, R. Queiroz, E.M. Hung, “Fast H.264/AVC to HEVC Transcoding Based on Machine Learning,” International Telecommunications Symposium (ITS), pp. 1-4, 2014.

[8] E. Peixoto, B. Macchiavello, E.M. Hung, A. Zaghetto, T. Shanableh, and E. Lzquierdo, “An H.264/AVC to HEVC video transcoder based on mode mapping,” IEEE International Conference on Image Processing (ICIP), pp. 1972-1976, September 2013.

[9] A. J. Diaz-Honrubia, J. L. Martinez, J. M. Puerta, J. A. Gamez, J. De Cock, and P. Cuenca, “Fast Quadtree Level Decision Algorithm for H.264/HEVC Transcoder,” in IEEE International Conference on Image Processing (ICIP), pp. 2497–2501, 2014.

[10] A. Diaz-Honrubia, J. L. Martinez, J. M. Puerta, J. A. Gamez, J. De Cock, and P. Cuenca, “A Data-Driven Probabilistic CTU Splitting Algorithm for Fast H.264/HEVC Video Transcoding,” in the Proceedings of Data Compression Conference (DCC), pp. 449, 2015.

[11] F. Zheng, Z. Shi, X. Zhang and Z. Gao, “Effective H.264/AVC to HEVC transcoder based on Prediction Homogeneity,” in IEEE Visual Communications and Image Processing Conference (VCIP), pp. 233-236, December 2014.

[12] F. Zheng, Z. Shi, X. Zhang, and Z. Gao, “Fast H.264/AVC To HEVC Transcoding Based on Residual Homogeneity”, in IEEE International Conference on Audio, Language and Image Processing (ICALIP), pp. 765-770, 2014.

[13] A. Nagaraghatta, Y. Zhao, G. Maxwell, and S. Kannangara, “Fast H.264/AVC to HEVC transcoding using mode merging and mode mapping”, IEEE 5th International Conference on Consumer Electronics - Berlin (ICCE-Berlin), pp. 165-169, 2015.

[14] F. Duanmu, Z. Ma, W. Wang, M. Xu and Y. Wang, “A Novel Screen Content Fast Transcoding Framework Based on Statistical Study and Machine Learning”, International Conference of Image Processing (ICIP), pp. 4205-4209, Phoenix, Arizona, USA, 2016.

[15] F. Duanmu, Z. Ma, and Y. Wang, “Fast Mode and Partition Decision Using Machine Learning for Intra-Frame Coding in HEVC Screen Content Coding Extension,” in IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), Vol: 6, Issue: 4, Page:517-531, 2016.

[16] F. Duanmu, Z. Ma, and Y. Wang, “Fast CU partition decision using machine learning for screen content compression”, International Conference on Image Processing (ICIP), pp. 4972 - 4976, Quebec City, Canada, 2015.

[17] H. Yu, R. Cohen, K. Rapaka, J. Xu, “Common Test Conditions for Screen Content Coding,” Doc. JCTVC-T1015, February 2015.

[18] G. Bjontegaard, “Calculation of Average PSNR differences Between RD Curves (VCEG-M33),” in VCEG Meeting (ITU-T SG16 Q.6), Austin, Texas, USA, Apr. 2-4, 2001.

978-1-5386-0462-5/17/$31.00 ©2017 IEEE.