IMPROVED INTRA MODE SIGALING FOR HEVC

Glenn Van Wallendael, Sebastiaan Van Leuven,
Jan De Cock, Peter Lambert,
and Rik Van de Walle

Ghent University – IBBT, ELIS Department - Multimedia Lab,
Gaston Crommenlaan 8 bus 201, B-9050 Ghent, Belgium

Joeri Barbarien,
Adrian Munteanu

Vrije Universiteit Brussel – IBBT, ETRO Department,
Pleinlaan 2, B-1050 Brussels, Belgium

ABSTRACT

In the current development of HEVC, compression performance improved significantly compared to H.264/AVC for both inter pictures and intra pictures. With intra compression, the main reason for this improvement is the large increase in intra prediction directions (up to 34). The downside of having a larger number of modes is that they increase the signaling overhead in the bitstream. In this paper, a low complexity intra mode prediction algorithm is proposed which improves the mode prediction accuracy. This is achieved by exploiting the correlation between the prediction directions of the neighboring prediction units and that of the encoded prediction unit. As a result, more efficient intra mode signaling can be achieved with minimal impact on encoder and decoder complexity. On average, 0.33% bitrate improvement is obtained by employing the proposed algorithm. For sequences that are encoded with a high number of directional intra modes, around 1% bitrate improvement is measured.

Index Terms— HEVC, intra mode signaling, most probable mode prediction

1. INTRODUCTION

In April 2010, the Joint Collaborative Team on Video Coding (JCT-VC) started their work on a new video coding standard with the goal of outperforming the largely adopted H.264/AVC standard. The project is called High-Efficiency Video Coding or HEVC in short. A draft specification can be found at [1]. Since the beginning of the project, improvements in two different directions were stimulated, namely high compression efficiency and low computational complexity.

Recently, a comparison between H.264/AVC and the current draft HEVC model was conducted [2]. The results of this comparison show average bitrate improvements of 36.2% for high efficiency random access configurations. Remarkably, for the intra only case, a less considerable 17.1% bit rate reduction was obtained. This indicates that there is a clear need to further improve the intra coding efficiency.

Essentially, more efficient intra coding can be realized in two ways: by improving the intra prediction step or by improving the intra mode signaling. The intra prediction step tries to predict the image samples in the prediction unit as closely as possible based on the already coded and reconstructed neighboring samples. The better the original pixels are matched, the smaller the residue to be encoded and consequently the more efficient the compression will be.

To enable proper decoding of the video stream, the selected intra mode must be signaled to the decoder. By efficiently predicting this optimal mode, less signaling overhead is required, leading to improved performance. This is the focus of our paper. A new intra mode prediction and coding scheme is presented which reduces the signaling overhead for intra mode information.

The paper is organized as follows: in Section 1, the H.264/AVC and HEVC intra prediction processes, including the employed intra mode signaling are shortly revised. Thereafter, some alternative intra mode signaling algorithms are discussed in Section 3. Then, in Section 4, our improved HEVC intra mode signaling is presented. Section 5 details the experimental evaluation of the proposed technique. Finally, Section 6 concludes the paper.

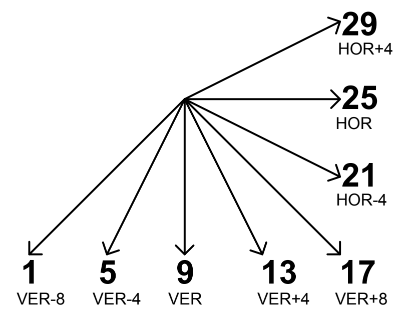

Fig. 1. Arrows indicating different intra prediction directions with the corresponding logical mode number and name.

Table 1. Intra mode ordering in HEVC

Nr	Logical order	Mode order	Nr	Logical order	Mode order
0	DC	VER	17	VER+8	HOR+6
1	VER-8	HOR	18	HOR-7	VER-7
2	VER-7	DC	19	HOR-6	VER-5
3	VER-6	VER-8	20	HOR-5	VER-3
4	VER-5	VER-4	21	HOR-4	VER-1
5	VER-4	VER+4	22	HOR-3	VER+1
6	VER-3	VER+8	23	HOR-2	VER+3
7	VER-2	HOR-4	24	HOR-1	VER+5
8	VER-1	HOR+4	25	HOR	VER+7
9	VER	HOR+8	26	HOR+1	HOR-7
10	VER+1	VER-6	27	HOR+2	HOR-5
11	VER+2	VER-2	28	HOR+3	HOR-3
12	VER+3	VER+2	29	HOR+4	HOR-1
13	VER+4	VER+6	30	HOR+5	HOR+1
14	VER+5	HOR-6	31	HOR+6	HOR+3
15	VER+6	HOR-2	32	HOR+7	HOR+5
16	VER+7	HOR+2	33	HOR+8	HOR+7

2. INTRA PREDICTION

The block structure employed by the newly emerging HEVC standard is distinctly different from the one used in H.264/AVC. We will therefore commence this section by giving an overview of this structure. Thereafter, the differences between HEVC's intra prediction process and that of H.264/AVC will be explained.

In HEVC, a frame is divided in Largest Coding Units (LCU) with a size of 64x64 instead of in macroblocks of size 16x16 as defined in H.264/AVC. These LCUs are coded in raster scan order, consistent with the macroblock coding order used in H.264/AVC. Every LCU can be recursively divided into Coding Units (CUs) forming a quad tree structure. In the bitstream, a minimal CU size is defined, indicating when further splitting of a CU is no longer allowed. Within every CU, two independent partitioning structures can be defined, one for the prediction and one for the transform. To predict the CU, it is quad tree partitioned in Prediction Units (PU). These PUs define the size on which inter or intra prediction is applied. As a next step, for inter coded CUs the transform block sizes can be independently chosen from PU sizes. For this purpose, a Transform Unit (TU) quad tree can be defined. For intra prediction, transform and prediction units are equal. Consequently, in the rest of the text, only PU blocks and sizes are relevant to explain the intra prediction process.

In H.264/AVC, for 4x4 and 8x8 blocks, intra prediction is performed by extrapolating neighboring pixels in one of eight directions (see Fig. 1) [3]. These eight directional modes are complemented with a DC mode, which predicts all values in the block as the mean of the neighboring samples. With the development of HEVC, these eight directional modes were extended to 33 modes [1], resulting in increased precision of the prediction.

Table 2. Relative bandwidth consumption of signaling mode information in bitstream.

	QP 22 [%]	QP 27 [%]	QP 32 [%]	QP 37 [%]	Avg [%]
Class A	1.82	2.60	3.88	5.23	3.38
Class B	4.71	6.98	9.86	12.24	8.45
Class C	6.69	9.03	12.62	16.17	11.13
Class D	6.38	8.78	12.61	16.73	11.12
Class E	7.48	10.03	13.97	17.36	12.21
Avg	5.42	7.48	10.59	13.55	9.26

In the HEVC intra encoding process, two different types of ordering for the intra prediction modes are defined. In *logical ordering*, the directional modes are enumerated based on their associated angle, starting with VER-8 and progressing counterclockwise, as shown in Fig. 1. The complete ordering is shown in the second column of Table 1. Logical ordering was defined to simplify calculations in the HEVC intra prediction process. The second ordering, called *mode order*, assigns a smaller code number to the more prevalent directions. This ordering was originally defined by JCT-VC to easily take a relevant subset of the modes when complexity-efficiency trade-offs must be made. The ordering is detailed in the third column of Table 1.

Besides allowing an increased number of prediction directions, HEVC also supports fine-grained directional prediction on larger block sizes than H.264/AVC. In H.264/AVC, intra-prediction considering all 9 modes was limited to 4x4 and 8x8 blocks. In HEVC, the total directional range can be used for block sizes of 8x8, 16x16, and 32x32. In both H.264/AVC and HEVC, a reduced set of the directions is applied to other block sizes well. For example, in HEVC, 4x4 block sizes can only be encoded with 17 intra modes. These 17 modes are the first 17 modes in mode order as seen in the third column of Table 1).

There is certainly no doubt that, when signaling overhead is not considered, an increased number of directional modes leads to better prediction accuracy and consequently to improved compression efficiency. However, instead of having to signal one out of 9 modes like in H.264/AVC, one out of 34 modes must be signaled in the PU header, resulting in increased overhead.

The overhead caused by signaling the mode information can be found in Table 2. The table shows the relative bitrate consumption of the mode information for four different quality parameters (QP) and five different classes of sequences. These classes correspond to different resolutions, but they will be described in more detail in Section 5. From the table it can be observed that on average 9.26% of the bitstream is occupied by intra mode information. For high QP values, up to 17.36% of the bitstream consists of this data. It can be observed that intra mode information consumes a significant part of the bitstream.

To reduce signaling overhead, both H.264/AVC and HEVC apply intra mode prediction. This process first calculates the most probable intra mode, which is derived from the intra modes employed in the left and top neighboring blocks or PUs. A one-bit flag is then used to indicate if the selected mode is equal to the most probable mode. If the selected mode is equal to the most probable one, the flag is set to 1 and no other information needs to be sent. When the mode to be encoded is different from the most probable one, the flag is set to 0 and one of the remaining 33 modes needs to be signaled additionally. This is performed by representing the selected mode's number in mode order with a fixed-length binarization. So, to represent one out of the 16 remaining modes for a 4x4 PU, four binary symbols need to be coded. For the 33 modes of 8x8, 16x16, and 32x32 PU sizes, a five bit binarization is chosen. The sequence 11111 is then used to indicate that another binary symbol will follow, which then allows to discern between the 32nd and 33rd modes. In all cases, the binary symbols obtained after binarization are entropy-coded using the same context model. Despite of the fact that the probability distribution of the intra prediction modes is non-uniform, all modes are thus signaled with a fixed length binarization whereby all binary symbols are entropy-coded using the same context model. This signaling is therefore likely to be suboptimal.

To derive the most probable mode as used for intra mode prediction in HEVC, the mode numbers of the top and left neighboring PUs are considered and the smallest of these mode numbers in mode order is retained. When a neighboring PU is not intra predicted, then the DC mode is chosen for this PU. Finally, when one of the neighboring blocks is not available, the most probable mode is always set to DC. One can see that this design is not flawless. For example, when a vertical mode is used in the top neighboring PU and the mode of the left neighboring PU is not available, then the algorithm chooses DC as the most probable mode.

To the design of H.264/AVC and HEVC intra mode prediction, two improvements can be applied. First, the most probable mode calculation can be improved and second, the signaling of the remaining modes should take into account the probability of the different modes.

3. RELATED WORK

Input contributions to previous JCT-VC meetings illustrate that both the most probable mode prediction and the signaling of the remaining modes do not perform optimal [4].

For signaling the most probable mode, an extension was made to include both the above and the left mode. So, first, a flag would be sent to signal if the chosen mode is one of the two most probable modes. Then, an index is included in the bitstream indicating whether the mode with the smallest or the largest mode number should be used for prediction [5] [6]. When the optimal mode is neither of the most probable modes, then the most probable modes are removed from the potential list and the index of the correct mode is included in the bitstream.

To improve the signaling when the optimal mode is not one of the most probable modes, multiple predictor sets [7][8] were introduced. By dividing the modes into four sets, the entropy coder could better predict the set in which the right mode is contained. Because of the improved prediction, a better compression of the mode could be obtained.

Alternatively, to predict the optimal intra direction, context dependent intra mode coding [9][10] was proposed as well. For every possible combination of the above and left modes, probabilities of occurrence for the different modes are calculated and stored. Depending on these neighboring modes and depending on the probabilities, one of nine binarization schemes is selected. This algorithm provided on average 0.8% bandwidth reduction, but complexity and memory usage were rather high.

Table 3. JCT-VC test sequences

Class	Sequence name	Frame count	Frame rate	Resolution
A	Traffic	150	30fps	crop 4K
A	PeopleOnStreet	150	30fps	crop 4K
A	Nebuta	300	60fps	crop 4K
A	SteamLocomotive	300	60fps	crop 4K
B	Kimono	240	24fps	1080p
B	ParkScene	240	24fps	1080p
B	Cactus	500	50fps	1080p
B	BQTerrace	600	60fps	1080p
B	BasketballDrive	500	50fps	1080p
C	RaceHorses	300	30fps	832x480
C	BQMall	600	60fps	832x480
C	PartyScene	500	50fps	832x480
C	BasketballDrill	500	50fps	832x480
D	RaceHorses	300	30fps	416x240
D	BQSquare	600	60fps	416x240
D	BlowingBubbles	500	50fps	416x240
D	BasketballPass	500	50fps	416x240
E	Vidyo1	600	60fps	720p
E	Vidyo3	600	60fps	720p
E	Vidyo4	600	60fps	720p

4. PROPOSED METHOD

The proposed method combines most probable mode prediction and mode signaling in one elegant solution. In general, instead of regarding all directional modes as equally important, a priority list is made, wherein the most probable mode appears first and the least probable mode last. The

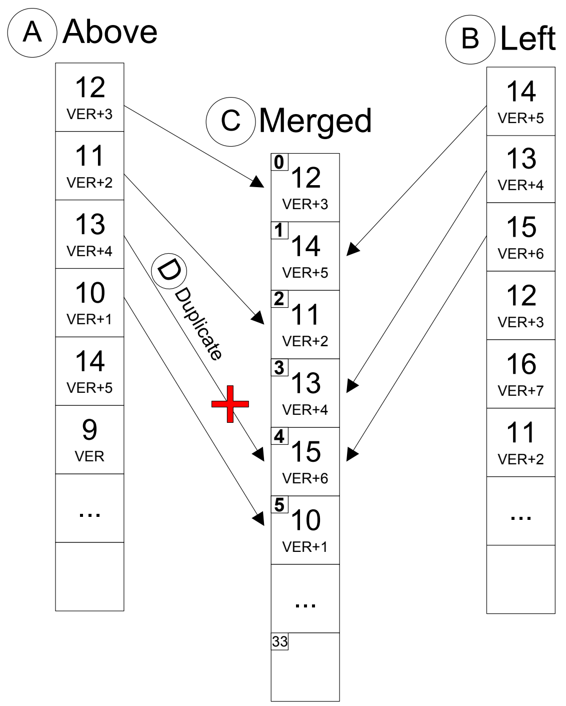

Fig. 2. Illustration of proposed algorithm with above PU having logical mode 12 (VER+3) and the left PU having mode 14 (VER+5).

most probable modes are then coded using the shortest binary symbol sequence.

More specifically, the algorithm consists of five steps as illustrated with an example in Fig. 2. First, from the top neighboring PU, the intra mode number in logical order is taken as the starting point to fill a prioritized list of directions. This prioritized list will contain the modes that are most related to the top neighboring mode in front and those that are least related in the end. As a first element, the top neighboring mode itself is taken, followed by the mode minus one in logical order. From Table 1, it can be observed that both the mode plus one and minus one give modes equally close to the considered mode, but in this approach, it was arbitrarily chosen to give the mode minus one a higher priority. After the top neighboring mode and the top neighboring mode minus one, the top neighboring mode plus one is added to the priority list and so on. The result of this process with mode 12 (VER+3) as the top neighboring mode can be found in Fig. 2 indicated with an 'A'.

As a second step, the same procedure is repeated for the logical intra mode number of the left neighboring PU. An example of this with mode 14 (VER+5) for the left neighboring PU can be found in Fig. 2 indicated with 'B'.

The third step is to interleave both of these lists into one merged priority list. Basically, the merged list is obtained by alternately adding an element from each of the two lists. It is obvious that when a horizontal mode (HOR-2...HOR+2) is used in the left neighboring PU, chances are high that a horizontal mode is used in the current PU. In this case, merging will start with the first element of the left-neighboring PU's priority list. For example, if a HOR+1 mode predicted the left neighboring PU, this HOR+1 direction is included first in the merged priority list. Thereafter, an element is taken from the priority list of the top neighboring PU, followed by an element form the left-neighboring PU's priority list and so on.

Equivalently, if a vertical mode (VER-2...VER+2) is used in the top neighboring PU, chances of having vertical prediction of the current PU are high. In this case, merging starts with the first element of the top-neighboring priority list.

Because every mode should occur only once in the merged list, duplicate entries must be removed. This is done as a fourth step.

As a result, without excessive complexity, a prioritized list is obtained which is dependent on the modes of both the top and left neighboring PU and which takes into account the correlation between the modes of the current and neighboring PUs. With this list created, the current intra mode is signaled by its index in the merged table. For example, when in Fig. 2, the current intra mode is selected to be mode 10, then because this mode occurs at the fifth position in the merged priority list, the index five will be signaled in the bitstream. Note that during the construction of the merged table, only directional modes were taken into account. To include the DC mode, this mode is inserted on the second position in the merged priority list.

In the encoder, the merged priority list should only be calculated once for every PU. Therefore, the impact on the encoding process is minimal.

As the last step, the index in the merged priority list is entropy coded by the CABAC algorithm. Previously, all modes were encoded as there binary representation. For information that is distributed with equal probability, this is an efficient solution. With our proposed algorithm, smaller indices are more likely to occur. Consequently, representing the indices in a binary form would lead to suboptimal entropy compression. By using a unary binarization, the entropy coder can represent smaller indices with less bits resulting in more optimal compression.

When there is no information about the directionality of intra modes of neighboring PUs, the algorithm cannot be applied. This situation occurs when the current PU is at a left or upper edge of the image and when neighboring blocks are predicted with DC prediction. In this case, the original intra mode coding is used.

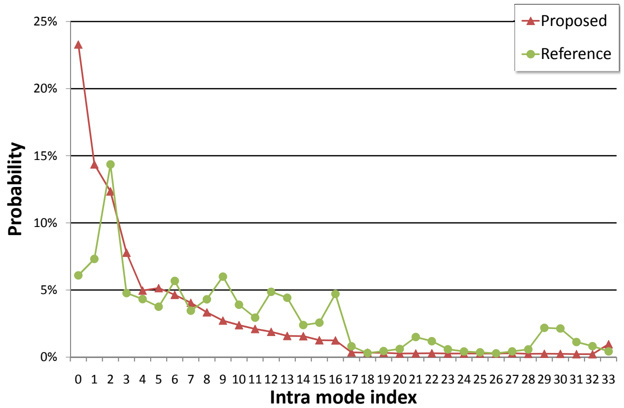

Fig. 3. Relative occurrence of mode indicator as encoded in the bitstream.

5. EXPERIMENTAL RESULTS

The proposed algorithm is experimentally evaluated using the entire set of test sequences prescribed by JCT-VC (see Table 3) [11]. This test set is divided in classes corresponding to the resolution. Class A is the largest resolution and these are sequences shot at approximately 4Kx2K, but cropped to 2560x1600 to facilitate faster encoding. The sequences are coded using Quantization Parameters (QP) values of 22, 27, 32, and 37 to cover a broad range of qualities and bitrates. Sequences are coded in an all intra configuration with high efficiency coding parameters as described in [11]. In all tests, the results are compared to the latest HM v2.0 reference codec [12]. Because the algorithm is implemented as an extension of the HEVC reference codec, encoding and decoding time is used as a measure for complexity.

First, the relative occurrence of the different modes in all the generated bitstreams is calculated and visualized in Fig. 3. This graph is a histogram of the modes as they are coded in the bitstream. For the reference sequences, modes were signaled as a binary representation of the modes in mode order. Therefore, as a reference, the relative occurrence of the modes in mode order is plotted. This graph shows that there is no clear trend in the occurrence of modes, except that modes between 0 and 16 occur more often than modes between 16 and 33.
In the graph labeled as 'proposed', the encoded indices resulting from the proposed algorithm are visualized. Here, a higher frequency of smaller indices is noticeable. Because these indices are more likely smaller numbers, fewer bits can be assigned to these more frequent indices. Consequently, improved entropy compression can be obtained, resulting in overall efficiency improvement.
Finally, to show effective bandwidth improvements, Bjøntegaard Delta (BD) [13] results are calculated (see Table 4). On average, 0.33% bandwidth reduction is obtained

Table 4. BD bandwidth reductions of individual sequences.

	Intra		
	Y BD-rate	U BD-rate	V BD-rate
Traffic	-0,23 %	-0,11 %	-0,04 %
PeopleOnStreet	-0,51 %	-0,47 %	-0,38 %
Nebuta	-0,12 %	-0,19 %	-0,14 %
SteamLocomotive	-0,24 %	-0,06 %	0,20 %
Kimono	-0,16 %	-0,04 %	-0,08 %
ParkScene	-0,24 %	-0,12 %	-0,14 %
Cactus	-0,43 %	-0,35 %	-0,22 %
BasketballDrive	-0,72 %	-0,46 %	-0,52 %
BQTerrace	-0,37 %	-0,18 %	-0,07 %
BasketballDrill	-1,06 %	-0,95 %	-0,93 %
BQMall	-0,09 %	0,13 %	0,15 %
PartyScene	0,21 %	0,19 %	0,20 %
RaceHorses	-0,40 %	-0,35 %	-0,29 %
BasketballPass	-0,23 %	-0,11 %	-0,17 %
BQSquare	-0,15 %	-0,10 %	-0,21 %
BlowingBubbles	-0,12 %	-0,09 %	-0,06 %
RaceHorses	-0,34 %	-0,19 %	-0,24 %
Vidyo1	-0,78 %	-0,26 %	-0,19 %
Vidyo3	-0,11 %	0,22 %	0,65 %
Vidyo4	-0,45 %	-0,10 %	0,03 %

Table 5. BD bandwidth reductions summarized by class.

	Intra		
	Y BD-rate	U BD-rate	V BD-rate
Class A	-0.28%	-0.21%	-0.09%
Class B	-0.38%	-0.23%	-0.21%
Class C	-0.34%	-0.24%	-0.21%
Class D	-0.21%	-0.12%	-0.17%
Class E	-0.45%	-0.05%	0.16%
All	-0.33%	-0.18%	-0.12%
Enc Time[%]	101%		
Dec Time[%]	104%		

by applying the proposed algorithm (see Table 5). Furthermore, it can be noticed that the reduction is equivalent for all classes, so the algorithm is resolution independent. On sequences like BasketballDrill, where a lot of directional modes are used, 1% bandwidth gain could be obtained. On the other hand, for sequences where the DC mode was frequently used, improvements were less significant.

6. CONCLUSION

We have proposed a low complexity intra mode prediction algorithm which combines most probable mode flag signaling and intra mode signaling in one elegant solution. The algorithm takes neighboring intra modes into account to obtain a prioritization of the different modes. As a result, entropy coding can be performed with a higher efficiency. We have shown that with a minimal complexity increase at encoder and decoder, a 0.33% bitrate reduction can be obtained.

7. ACKNOWLEDGEMENT

The research activities described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research–Flanders (FWO–Flanders), and the European Union. Furthermore, this work was carried out using the Stevin Supercomputer Infrastructure at Ghent University.

8. REFERENCES

[1] T. Wiegand, W.-J. Han, B. Bross, J.-R. Ohm, and G.J. Sullivan, "WD1: Working Draft 1 of High-Efficiency Video Coding," *JCTVC-C403*, Guangzhou, China, Oct. 2010.

[2] S. Park, J. Park, and B. Jeon, "Report on the evaluation of HM versus JM," *JCTVC-D181*, Daegu, Korea, Jan. 2011.

[3] Thomas Wiegand, Gary J. Sullivan, Gisle Bjøntegaard, and Ajay Luthra, "Overview of the H.264/AVC video coding standard," vol. 13, no. 7, pp. 560–576, 2003.

[4] S. Lei, M. Karczewicz, T.K. Tan, W. Ding, and G. Li, "Description of Core Experiment 14: Intra Mode Coding," *JCTVC-D614*, Daegu, Korea, Jan. 2011.

[5] S. Kanumuri, T.K. Tan, and F. Bossen, "Enhancements to Intra Coding," *JCTVC-D235*, Daegu, Korea, Jan. 2011.

[6] W. Ding, X. Che, Y. Shi, and B. Yin, "Improved Intra Mode Coding by Multiple Mode Candidates," *JCTVC-D253*, Daegu, Korea, Jan. 2011.

[7] G. Li, L. Liu, N. Ling, J. Zheng, and P. Zhang, "CE6.a.5: Santa Clara University and Hisilicon Report on Block Based Intra Prediction," *JCTVC-D026*, Daegu, Korea, Jan. 2011.

[8] L. Liu, "Multiple Predictor Sets for Intra Coding," *JCTVC-A022E*, Dresden, Germany, Apr. 2010.

[9] M. Guo, X. Guo, and S. Lei, "Improved Intra Mode Coding," *JCTVC-D166*, Daegu, Korea, Jan. 2011.

[10] Y.-W. Huang, C.-M. Fu, Y.-P. Tsai, J.-L. Lin, Y. Chang, J.-H. Guo, C.-Y. Chen, and S. Lei, "A Technical Description of MediaTek's Proposal to the JCT-VC CfP," *JCTVC-A109*, Dresden, Germany, Apr. 2010.

[11] F. Bossen, "Common test conditions and software reference configurations," *JCTVC-D600*, Daegu, Korea, Jan. 2011.

[12] Bossen, F. and Flynn, D. and Sühring, K., "HEVC Reference Software Manual ," *JCTVC-D404*, Daegu, Korea, Jan. 2011.

[13] G. Bjøntegaard, "Doc. VCEG-M33:Calculation of average PSNR differences between RD-curves," *ITU-T, Video Coding Experts Group (VCEG)*, Apr. 2001.