

차세대 비디오 코덱(JEM) 성능 향상 기법

# Coding Tools for Improving Coding Efficiency of Future Video Coding (JVET)

김신재, 도지훈

지도교수 : 김재곤 교수님





#### I. Introduction

- Joint Video Exploration Team (JVET)
  - □ ITU-T VCEG (Q6/16) and ISO/IEC MPEG (JTC 1/SC 29/WG 11)
  - Significantly exceeding the compression capability of HEVC/H.265
  - The scope of technology
    - SDR/HDR(High-Dynamic-Range) video, VR/360 video
  - Requirements
    - Compression performance: 30% 50% bit-rate reductions vs. HEVC
    - Video Formats: VGA (640x480) ~ 8K UHD
  - Timeline
    - Publication of a future video coding specification by approximately 2020
- JVET releases the Joint Exploration Model (JEM) Software codec for technical verification
- Released Call for Proposal (CfP) of JVET in Oct., 2017

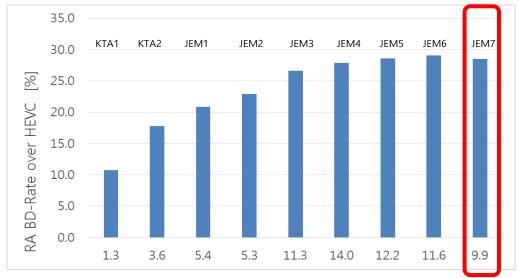




#### I. Introduction

- Coding efficiency of JEM 7.0 in the comparison with HEVC
  - 29% BD-rate saving with about 10 times encoding run-time

Test configuration	BD-rate			Time	
	Y	U	V	Enc.	Dec.
All Intra	-20%	-28%	-27%	×36	×2
Random Access	-29%	-35%	-34%	×10	×7
Low Delay-B	-22%	-28%	-29%	×9	×7
Low Delay-P	-26%	-31%	-32%	×7	×5



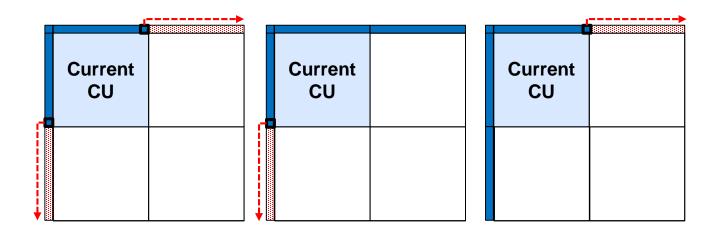
**Encoding run time relatively to HEVC** 







- Reference sample padding (RSP)
  - The reference sample padding is performed when no reference samples exist
    - There are no reference samples or partially available
  - The basic idea of reference sample padding
    - Fill in a sample that does not exist using the closest sample available

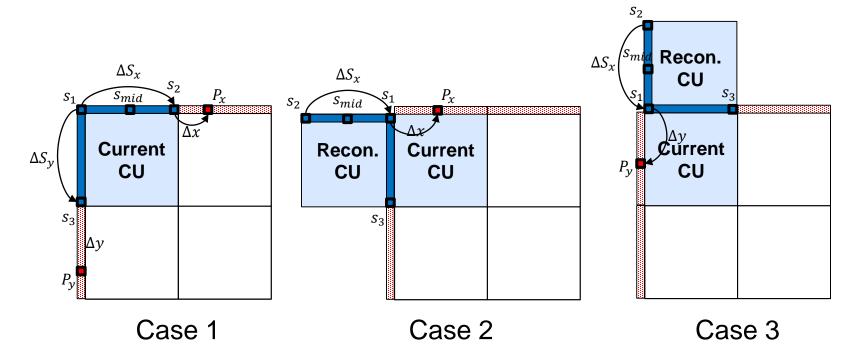








- Proposed Method
  - Assumption for reflecting the linearity of the available samples
    - If the available samples are linear, performance improvement is expected by filling the non-existing samples linearly







- Proposed Method
  - Determine linearity

$$Abs(s_1 + s_2 - 2s_{mid}) < (1 \ll (BitDepth_Y - x))$$

- As Increase linearity factor x, the condition of linearity is more strict
- If the above condition is satisfied, linear padding is performed
  - The variation is reflected according to the position of the padding area

$$\begin{array}{ll} s_1 - s_2 &= \Delta S_x \\ s_1 - s_3 &= \Delta S_y \end{array} \qquad P_x = s_2 - \frac{\Delta x \Delta S_x}{width_{CU}} \qquad P_y = s_3 - \frac{\Delta y \Delta S_y}{height_{CU}} \end{array}$$

 In case of the CU blocks boundary, reference the reconstruct block (Case 2, 3)







- Experimental Result
  - Test conditions
    - JEM 7.0 rc1, All Intra, HD (1920x1080) 64 frames
    - linear factor x = 8

	BD-rate				
HD sequence	Υ	U	V		
ParkScene	0.02%	-0.10%	-0.13%		
Cactus	0.03%	-0.32%	0.16%		
BasketballDrive	-0.09%	-0.14%	-0.32%		
BQTerrace	-0.03%	0.23%	0.27%		

< Test 1. Apply only Case 1>

	BD-rate				
HD sequence	Υ	U	V		
ParkScene	0.01%	-0.08%	-0.04%		
Cactus	0.02%	0.07%	0.13%		
BasketballDrive	0.02%	-0.26%	-0.10%		
BQTerrace	0.05%	0.17%	-0.14%		

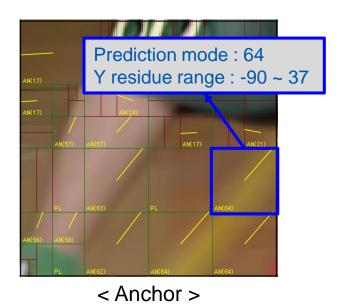
< Test 2. Apply Case 1, 2, 3>

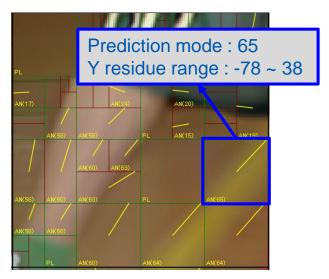






- Experimental Result
  - Analyze with JEM Bitstream Analyzer
    - Proposed method reduces the range of residue value. This improves quality by reducing quantization error. But the inverse cases also exist





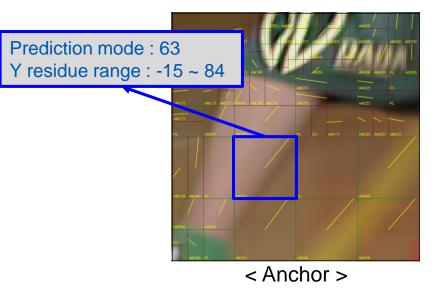
< Proposed >

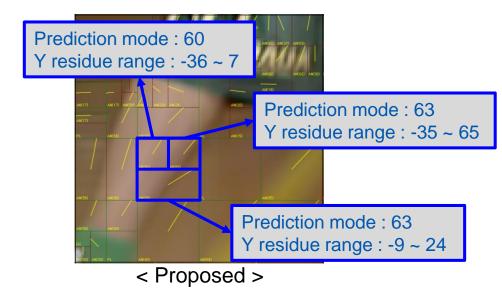






- Experimental Result
  - Analyze with JEM Bitstream Analyzer
    - An increase in bit amount can also be caused when divided into smaller blocks than Anchor









- Merge mode
  - Merge mode to derive the motion information from spatially or temporally neighboring blocks
  - Transmit index information to select one out of several available candidates (5 candidates)

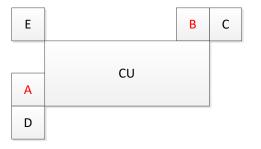
$S_0$	$S_1$	$S_2$	$S_3$	Col
0			3	

- spatial candidates (max 4 candidates)
  - Search for spatial candidates  $S_0$ ,  $S_1$ ,  $S_2$ ,  $S_3$
  - Remove redundant candidates
- temporal candidates (max 1 candidate)
  - Search for temporal candidate Col
- combined bi-predictive candidates
- zero motion vector candidate

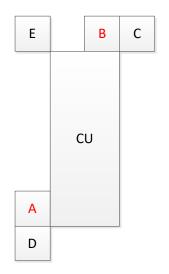




- Proposed Method
  - Change the List of Merge by Block Size
    - If width ≥ height
      Existing merge sequence in JEM is followed
      - $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$



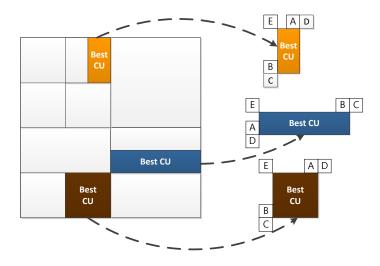
- If width < height</li>
  Change the derivation order of Left and Above
  - $B \rightarrow A \rightarrow C \rightarrow D \rightarrow E$







- Proposed Method
  - The order of MERGE candidates is different according to the shape of CU block and the CU's partition information
    - when width = height, it was not always QT split.
  - The width, height do not represent all CU split structures
    - Determined only by the CU partitioning







- Experimental Result of Method
  - Test conditions
    - JEM 7.0, Random Access, HD (1920x1080 for 1sec)

	BD-rate (piecewise cubic)			BD-rate (cubic)		
	Υ	U	V	Υ	U	V
FoodMarket	0.06%	0.36%	0.35%	0.08%	0.37%	0.36%
Tango	0.01%	-1.23%	0.03%	0.01%	-1.25%	0.01%
CatRobot	0.07%	-0.03%	0.37%	0.05%	-0.05%	0.32%
DaylightRoad	-0.02%	-0.25%	0.08%	-0.01%	-0.25%	0.09%
BuildingHall	-0.01%	-0.17%	-0.13%	-0.01%	-0.16%	-0.13%
UHD Total	0.02%	-0.26%	0.14%	0.02%	-0.27%	0.13%
BQTerrace	-0.06%	-0.33%	0.38%	-0.07%	-0.32%	0.37%
RitualDance	-0.15%	-0.04%	-0.21%	-0.15%	-0.07%	-0.35%
Timelapse	-0.12%	0.19%	-0.45%	-0.12%	0.12%	-0.55%
BasketballDrive	-0.16%	0.47%	0.57%	-0.18%	0.45%	0.56%
Cactus	-0.10%	0.10%	0.01%	-0.10%	0.10%	0.02%
HD Total	-0.12%	0.08%	0.06%	-0.12%	0.05%	0.01%
Total	-0.05%	-0.09%	0.10%	-0.05%	-0.11%	0.07%

< Test1 = Block size >

	BD-rate (piecewise cubic)			BD-rate (cubic)		
	Y	U	V	Y	U	V
BQTerrace	-0.03%	-0.37%	-0.13%	-0.05%	-0.38%	-0.13%
RitualDance	-0.03%	-0.18%	0.00%	-0.03%	-0.23%	0.10%
Timelapse	-0.14%	-0.13%	-0.43%	-0.15%	-0.14%	-0.42%
BasketballDrive	-0.06%	0.49%	-0.07%	-0.05%	0.50%	-0.05%
Cactus	0.03%	0.03%	-0.26%	0.04%	0.03%	-0.26%
HD Total	-0.05%	-0.03%	-0.18%	-0.05%	-0.04%	-0.15%

#### < Test2 = Test1 + partitioning >

	BD-rate (piecewise cubic)			BD-rate (cubic)		
	Y	U	V	Y	U	V
BQTerrace	-0.11%	-0.27%	-0.06%	-0.11%	-0.28%	-0.06%
RitualDance	0.00%	-0.10%	-0.07%	0.00%	-0.04%	-0.03%
Timelapse	-0.05%	0.41%	0.05%	-0.05%	0.39%	0.08%
BasketballDrive	-0.04%	0.38%	-0.20%	-0.02%	0.34%	-0.20%
Cactus	-0.06%	-0.09%	-0.24%	-0.06%	-0.09%	-0.23%
HD Total	-0.05%	0.06%	-0.11%	-0.05%	0.06%	-0.09%

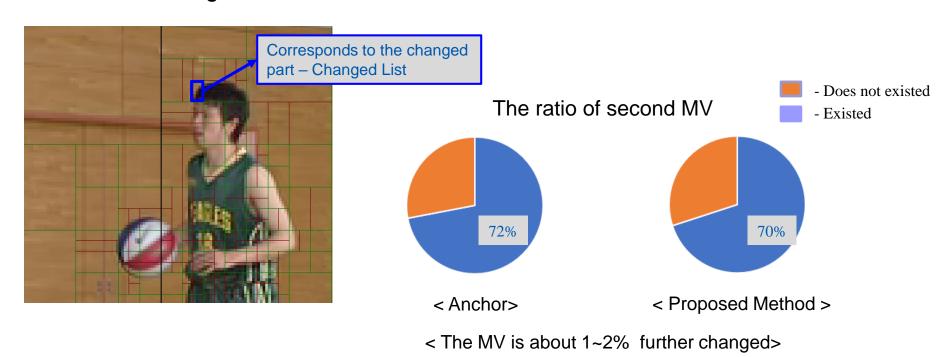
< Test3 = CU partitioning(Ver/Hor) >







- Experimental Result of Method
  - Analyze
    - By using proposed method, the percent of taking Above MV is changed









#### IV. Conclusion

- The proposed method of RSP has almost the same performance as the conventional method
  - Several sequence get a small gain when proposed method only applies to Case1
- The proposed method of taking Merge list has the best performance when considering only the block size
  - Logically, It is accurate when adding multiple conditions(ex. Partitioning and Second CU), but not in terms of performance.
  - Although the performance improvement is 0.12% in HD size, performance loss occurs in UHD.



