

JVET-J0018

Description of SDR video coding technology proposal by MediaTek

Chih-Wei Hsu, Ching-Yeh Chen,
Tzu-Der Chuang, Han Huang,
Shih-Ta Hsiang, Chun-Chia Chen,
Man-Shu Chiang, Chen-Yen Lai,
Chia-Ming Tsai, Yu-Chi Su,
Zhi-Yi Lin, Yu-Ling Hsiao
Jan Klopp, I-Hsiang Wang,
Yu-Wen Huang, Shaw-Min Lei

Overall Results

- Two operation points are provided, OP1 and OP2
 - Encoder-only non-normative changes between OP1 and OP2; Same decoders
 - OP1 is submitted for subjective evaluation

Anchor: HM-16.16		Constraint Set1			Constraint Set2		
		Y	U	V	Y	U	V
OP1	BD-rate	-43.8%	-45.6%	-47.4%	-31.3%	-37.5%	-38.3%
	run time	enc: 12.4x; dec: 13.0x			enc: 11.9x; dec: 9.3x		
OP2	BD-rate	-42.4%	-44.6%	-46.4%	-29.7%	-35.8%	-36.6%
	run time	enc: 6.1x			enc: 5.3x		

Anchor: JEM-7.0		Constraint Set1			Constraint Set2		
		Y	U	V	Y	U	V
OP1	BD-rate	-16.6%	-6.8%	-10.4%	-9.4%	-1.9%	-3.4%
	run time	enc: 1.6x; dec: 2.3x			enc: 1.4x; dec: 1.8x		
OP2	BD-rate	-14.4%	-5.1%	-8.8%	-7.2%	+0.2%	-1.0%
	run time	enc: 0.8x			enc: 0.6x		

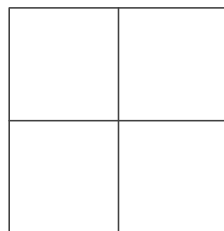
Outline

- Block structure
- Intra prediction
- Inter prediction
- Transform & quantization
- Reconstruction
- In-loop filter
- Entropy coding
- Encoder algorithm
- Performance
- Conclusion

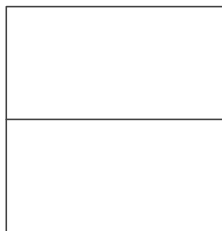
Block Structure

Multi-Type-Tree Block Structure (1/5)

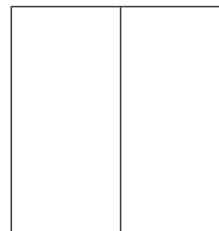
- Partition types
 - Non-split
 - Quad-Tree (QT) split
 - Horizontal/Vertical Binary-Tree (BT) split
 - Horizontal/Vertical Triple-Tree (TT) split



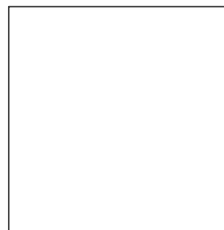
QT split



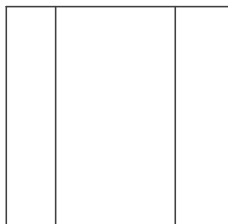
Horizontal BT split



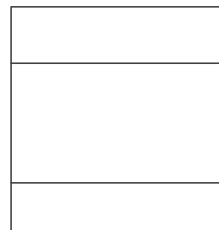
Vertical BT split



Non-split



Horizontal TT split



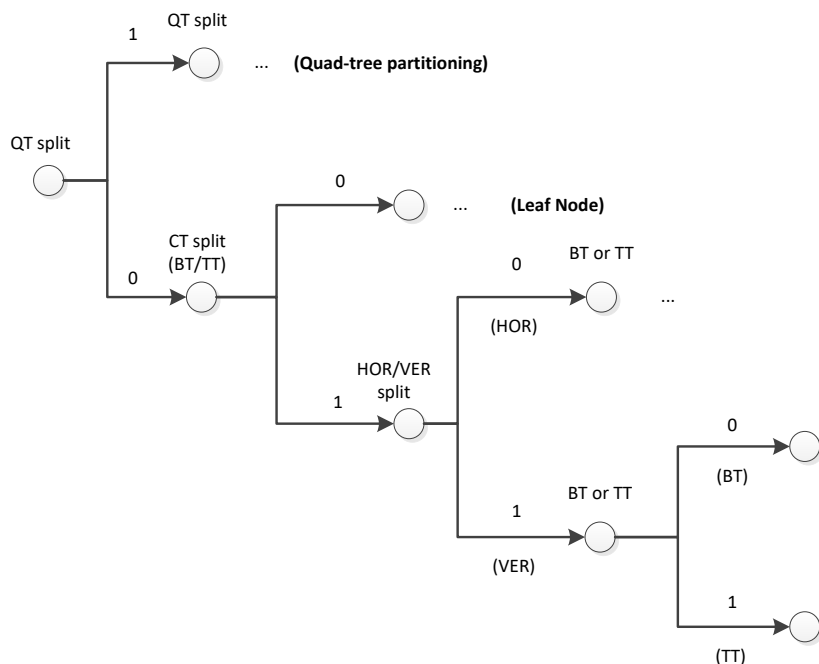
Vertical TT split

Multi-Type-Tree Block Structure (2/5)

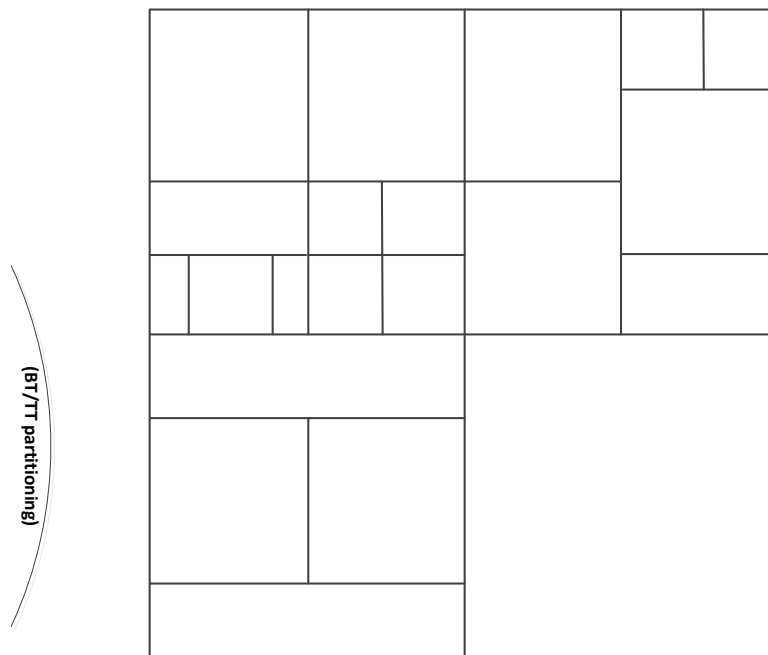
- Quad-Tree (QT) structure is firstly applied
- The leaf node of QT can be further split by a 2nd coding tree that can consist of Binary-Tree (BT) and Triple-Tree (TT) partitions
- The leaf node of the 2nd coding tree is coding unit (CU), which is the basic unit for both prediction and transform
 - No further PU or TU partition
- Separate block structure for luma and chroma in I slice

Multi-Type-Tree Block Structure (3/5)

- Signaling

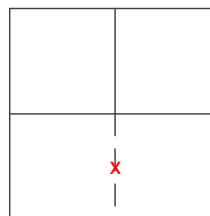


- Example

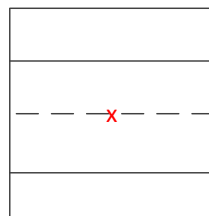


Multi-Type-Tree Block Structure (4/5)

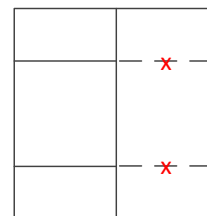
- Redundancy removal for P/B slice: 1) , ..., 6)
- Redundancy removal for I slice: 3) and 4)



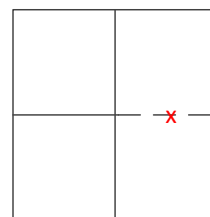
1)



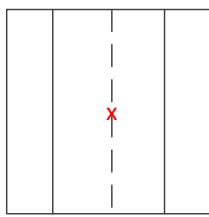
3)



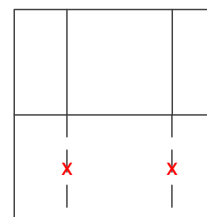
5)



2)



4)



6)

Multi-Type-Tree Block Structure (5/5)

- Parameters
 - CtuSize: the root node size of a QT, the same concept as in HEVC
 - *MinQTSIZE*: the minimum allowed QT leaf node size
 - *MaxCTSize*: the maximum allowed root node size for BT/TT partitioning
 - *MaxCTDepth*: the maximum allowed tree depth in BT/TT partitioning
 - *MinCTSize*: the minimum allowed leaf node size in BT/TT, i.e. minimum CU size

*CT: second Coding Tree, BT/TT

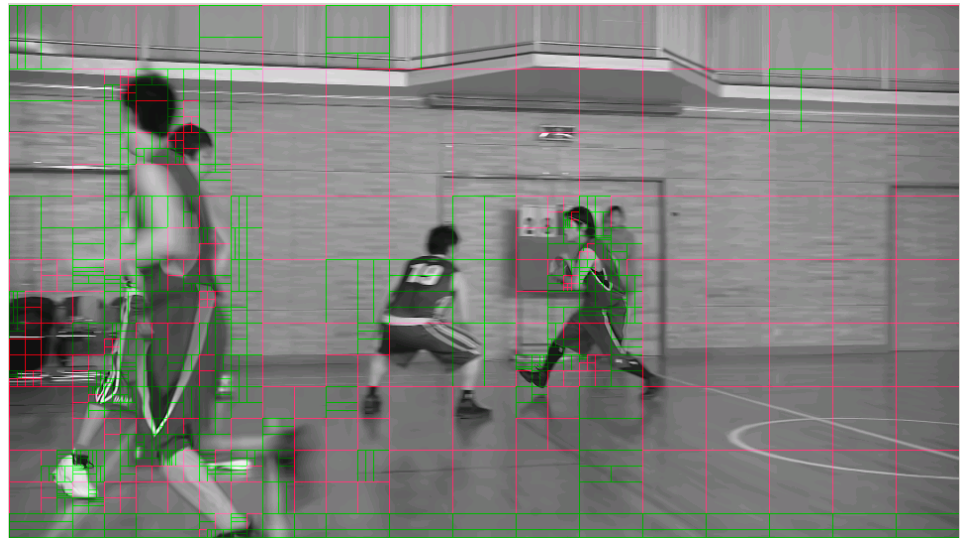
Inferred Split for CTUs at Picture Boundary

- When a CTU corresponds to the block region across the current picture boundaries
 - Forced to be further split into smaller-size child CUs
 - Can use QT split or BT split
 - Horizontal or vertical split is implicitly derived from the boundary conditions

QT split at picture boundary



QT or BT split at picture boundary



Large CU with CU Width or Height > 128

- Support maximum 128x128 transform
- Intra slices
 - Infer QT split for each 256x256 luma CTU
 - Support 128x128 chroma CU (in units of chroma samples)
- Inter slices
 - Skip mode is inferred in non-low-delay slices
 - Skip mode and AMVP mode without residual (CBF is inferred as 0) are supported in low-delay slices

*low-delay slice: all reference frames are prior to the current frame in display order

MEDIATEK

Intra Prediction

Intra Prediction for Luma

- 67 intra prediction modes are supported, including planar, DC, and 65 angular modes
 - 4-tap interpolation filters and smooth filters are the same as JEM-7.0
 - Planar mode is replaced by unequal weight planar (UWP), which is the same as JVET-E0068
- Intra boundary filters are supported
 - Gradient filters and bi-boundary filters are the same as JEM-7.0

Intra Prediction for Chroma

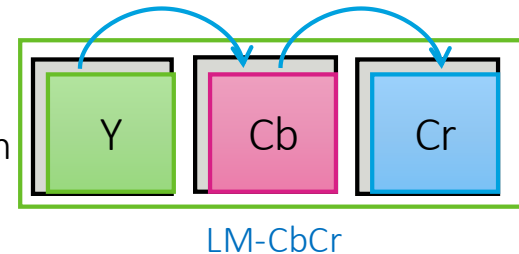
- Chroma prediction includes
 - Multi-direct modes (DM), same as in JEM-7.0
 - 5 modes
 - Apply CB-to-CR prediction to residuals
 - LM-based modes
 - 10 modes
 - 6 modes in JEM-7.0 (LM, MMLM, 4 MFLMs)
 - 4 new modes are added
 - LM-top, LM-left, LM-CbCr, LM fusion mode (DM + LM)

Intra Chroma Prediction

- 4 new LM-based modes are added

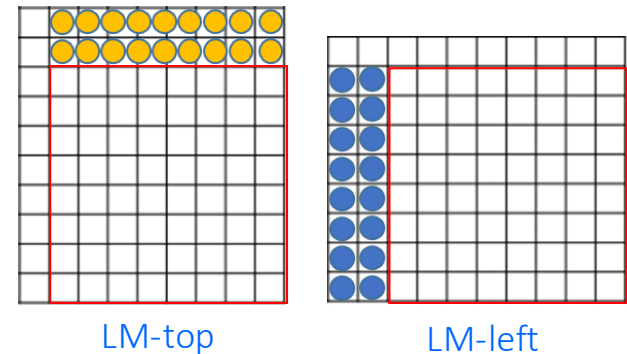
- LM-CbCr

- Two linear models are used
- One for Y to Cb, the other for Cb to Cr prediction



- LM-top and LM-left

- Only use the above/left neighboring samples
- 2-line samples



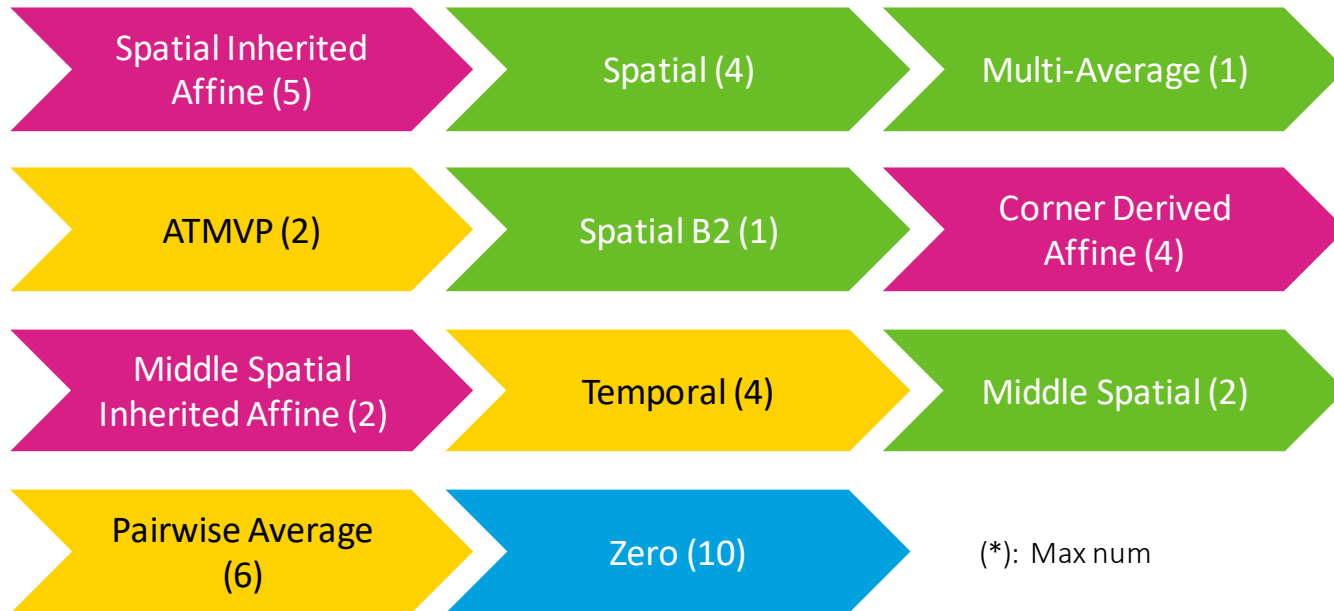
- LM fusion mode

- Predictor = $(DM + LM + 1) \gg 1$
- Signal a flag right after DM to indicate whether to apply LM fusion mode or not

Inter Prediction

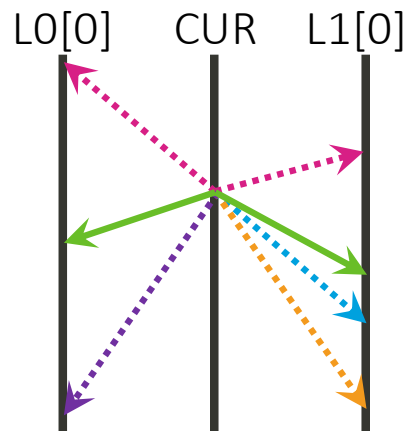
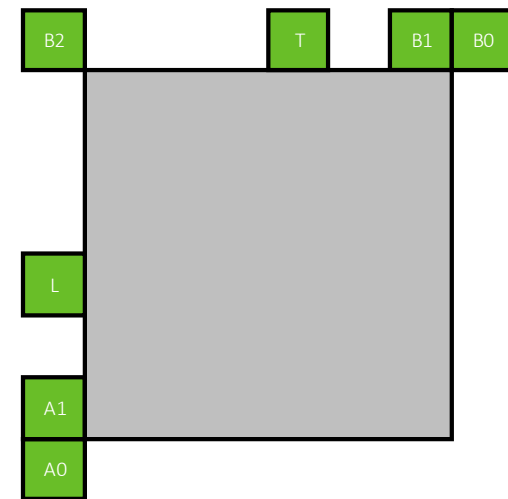
Unified Merge Mode

- Merge candidate list consists of spatial, temporal, and affine type candidates
- For unification, affine type candidate is indicated by a merge index, instead of using additional flag in JEM-7.0
- Availability and redundancy check before adding to candidate list
- The number of merge candidates is fixed to 10

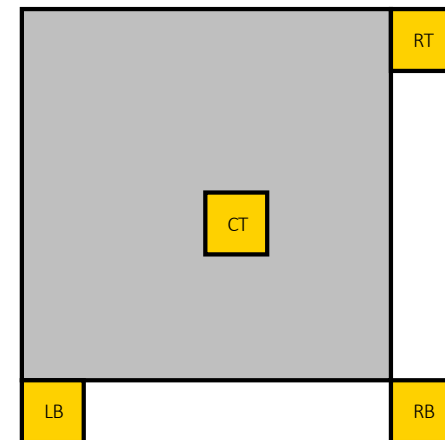


Spatial Type Candidates

- Use motion information from neighboring block
- **Spatial**: Insertion order: {A1, B1, B0, A0}
 - B2 (when the number of merge candidate in the current list is smaller than a threshold)
- **Middle spatial**: Insertion order: {L, T}
- **Multi-average**: Average all candidates with reference index equal to 0 in the current list and the averaged motion vectors are calculated separately for each reference list



Temporal Type Candidates



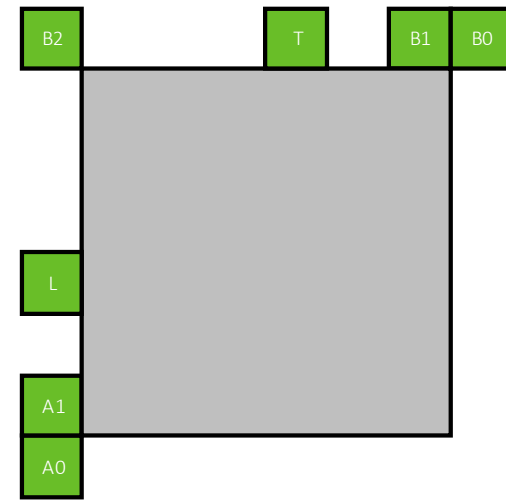
- Use motion information from temporal collocated block
- **Temporal**: Insertion order: {RB, CT, LB, RT}
- **ATMVP**: Basically the same as JEM-7.0 except two differences
 - Two temporal vectors are used to generate two ATMVP candidates instead of one
 - Derivations for the temporal vectors are different
 - First temporal vector: Average all available motion vectors with the same reference frame as the first available motion vector
 - Second temporal vector: Second available motion vector

Prediction direction of first candidate	First available motion vector	Second available motion vector
Bi	Motion vector of L0	Motion vector of L1
Uni (L0/L1)	Motion vector of L0/L1	Motion vector of second candidate

- **Pairwise average**: Average every pair of candidates in the current list by a predefined order
 - For example, 4 candidates in the current list, the pair order is (0, 1) (1, 0) (0, 2) (2, 0) (1, 2) (2, 1) (0, 3) (3, 0) (1, 3) (3, 1) (2, 3) (3, 2), where 4 candidates are indexed from 0 to 3

Affine Type Candidates

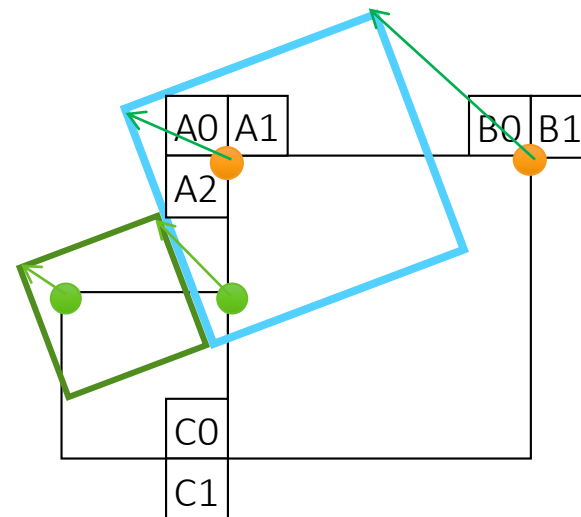
- **Spatial inherited affine, Middle spatial inherited affine:**
Insertion order: {A1, B1, B0, A0, B2}, {L, T}
 - Insert spatial candidates which are coded using affine mode
 - Control points of the current CU are derived according to the control points of spatial candidate
- **Corner derived affine:** Insertion order: {B2, A1, B1}, {B2, A0, B1}, {B2, A1, B0}, {B2, A0, B0}
 - Set three spatial candidates as control points
 - Three spatial neighbors with the same reference frame for each reference list



Affine Inter

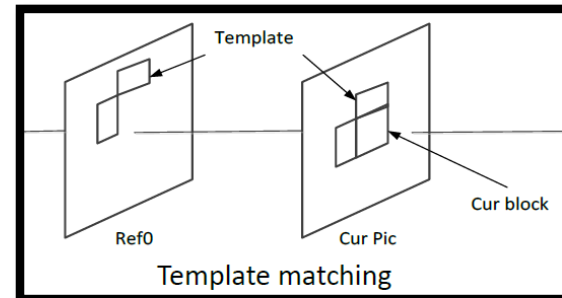
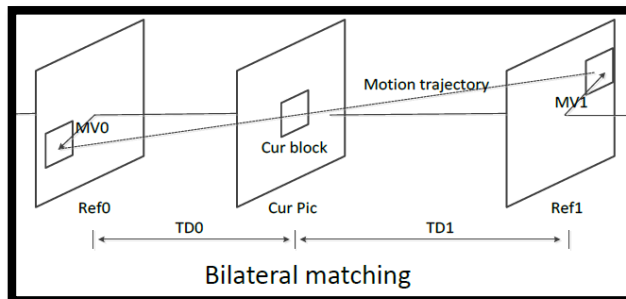
- 4-parameter (2 control points) affine model for affine inter prediction
 - Signal an MVP pair index and two MVDs
- Two types of affine MVP pairs to form a MVP pair list
 - Spatial inherited MVP pairs
 - Corner derived MVP pairs (from {set A, set B}, and {set A, set C})

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w} x + \frac{(v_{2x} - v_{0x})}{h} y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w} x + \frac{(v_{2y} - v_{0y})}{h} y + v_{0y} \end{cases}$$



Pattern-matched Motion Vector Refinement (PMVR)

- Use pattern-matched MV refinement to refine a signaled MV candidate
 - CU-level template matching refinement
 - CU-level and sub-CU-level bilateral matching refinement
 - Template and bilateral matching are similar to PMVD in JEM-7.0
- Low motion compensation bandwidth consumption
 - CU-level and sub-CU-level search share the same search window
 - Disable PMVR for small CUs
 - JEM PMVD BW: 45.9x → Proposed PMVR BW: 3.1x (compared with HEVC)

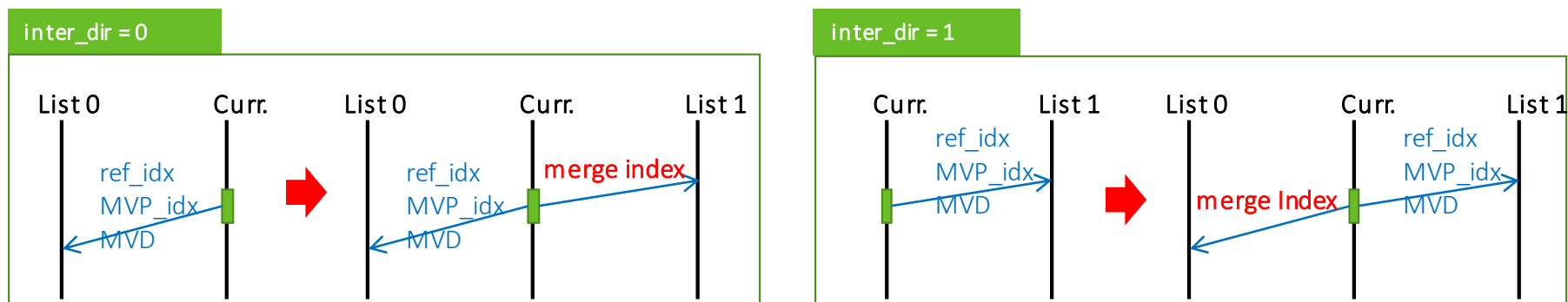


Merge Assistant Prediction (MAP)

- Combine existing prediction modes with extra merge indexed prediction, including
 - AMVP/uni-prediction mode
 - Skip/merge mode
 - Intra mode
- Merge indexed prediction is performed as in merge mode, where merge candidate lists are partially reused
- Different weights are applied to generate the final prediction depending on the combinations

Merge Assistant Prediction

- When applied to improve AMVP/uni-prediction
 - One flag is signaled for $\text{inter_dir} = 0$ or 1 (uni-prediction)
 - One more merge index is signaled when the flag is true
 - 1:1 weights are applied to the final prediction as bi-prediction



- Merge candidate lists are derived from merge mode with sub-CU candidates (e.g., affine, ATMVP) excluded
 - The number of merge candidates is fixed to 4

Merge Assistant Prediction

- When applied to improve skip/merge mode
 - Whether to enable MAP is explicitly signaled
 - Extra merge indexed prediction is selected in addition to the original one
 - Implicitly decided as the succeeding merge candidate (i.e., the already signaled merge index plus one) without signaling any additional merge index
 - 5:3 weights are applied to the 1st and 2nd merge indexed predictions
 - Only support non-sub-CU candidates (no affine, ATMVP)

1. Merge candidates w/o affine and ATMVP

merge index	merge candidates
0	MV0(L0)
1	MV1(L1)
2	MV2(L0) MV2(L1)

merge index



2. Consecutive merge candidates selection

merge index	merge candidates
0	MV0(L0)
1	MV1(L1)
2	MV2(L0) MV2(L1)

1st merge indexed Prediction

2nd merge indexed Prediction

Merge Assistant Prediction

- When applied to improve intra mode
 - For inter CU, one flag is signaled for merge mode
 - One more intra mode index is signaled when the flag is true
 - The number of intra mode candidates is fixed to 19

Weights depends on mode and positions

Merge candidates (same as merge)	
merge index	merge candidates
0	MV0(L0)
1	MV1(L1)
2	MV2(L0) MV2(L1)
...	...
9	MV9(L0) MV9(L1)



Intra candidate list		
intra mode index	intra candidates	syntax
0	MPM0	<u>1</u> 0
1	MPM1	<u>1</u> 10
2	MPM2	<u>1</u> 11
...	Remaining mode	<u>0</u> xxxx
18	Remaining mode	<u>0</u> xxxx

- For intra CU, one flag is signaled for luma component
 - One more merge index is signaled when the flag is true
 - The number of merge candidates is fixed to 4

Weights depends on mode and positions

Intra candidates (same as intra)	
Intra mode index	merge candidates
0	MPM0
...	...
5	MPM5
6~21	Selected mode
22~66	Non-selected mode



Merge candidates w/o Affine and ATMVP	
merge index	merge candidates
0	MV0(L0)
1	MV1(L1)
2	MV2(L0) MV2(L1)
3	MV3(L0) MV3(L1)

Adaptive Motion Vector Resolution (AMVR)

- Similar to locally adaptive motion vector resolution (LAMVR) in JEM-7.0
- Support different MV resolutions: quarter-luma sample, integer-luma sample, four-luma sample
- Difference from JEM-7.0
 - Remove the non-zero MVD constraint:
Always signal one AMVR index to indicate the resolution of MVs for one CU
 - Allow integer MV when MVD is zero (perform integer MC)

Local Illumination Compensation

- The same as JEM-7.0

Generalized Bi-Prediction (GBi)

- Similar to JVET-C0047
- Allow different weights for L0 and L1 predictors

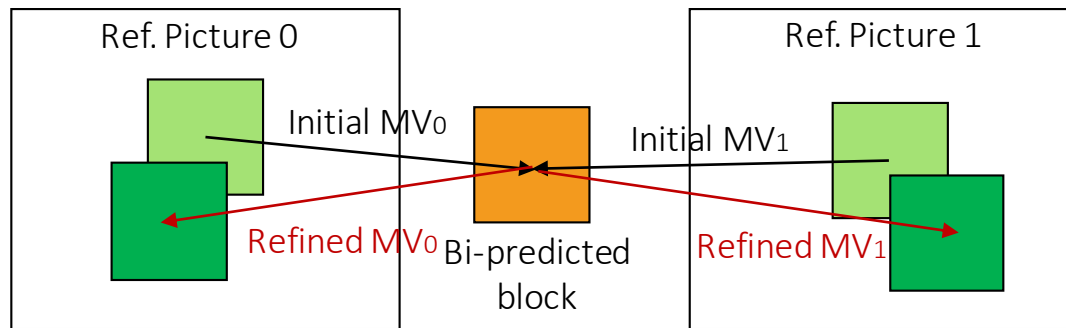
$$P_{TraditionalBiPred} = (P_{L0} + P_{L1} + RoundingOffset) \gg shiftNum$$

$$P_{GBi} = (w_0 * P_{L0} + w_1 * P_{L1} + RoundingOffset_{GBi}) \gg shiftNum_{GBi}$$

- Supported weights and signaling
 - $\{w_0, w_1\} = \{4, 4\}, \{5, 3\}, \{3, 5\}, \{2, 6\}, \{6, 2\}$
 - Explicitly signal the selection in AMVP mode
 - Inherited from the selected merge candidate in merge mode

Decoder-side Motion Vector Refinement (DMVR)

- Similar to DMVR in JEM-7.0
- Refine motion vectors using bilateral template matching



- Difference from JEM-7.0
 - Applied to CU coded with template matching PMVR besides merge mode
 - Terminate refinement of L1 if $\text{Initial MV}_0 == \text{Refined MV}_0$

Bi-directional Optical Flow (BIO)

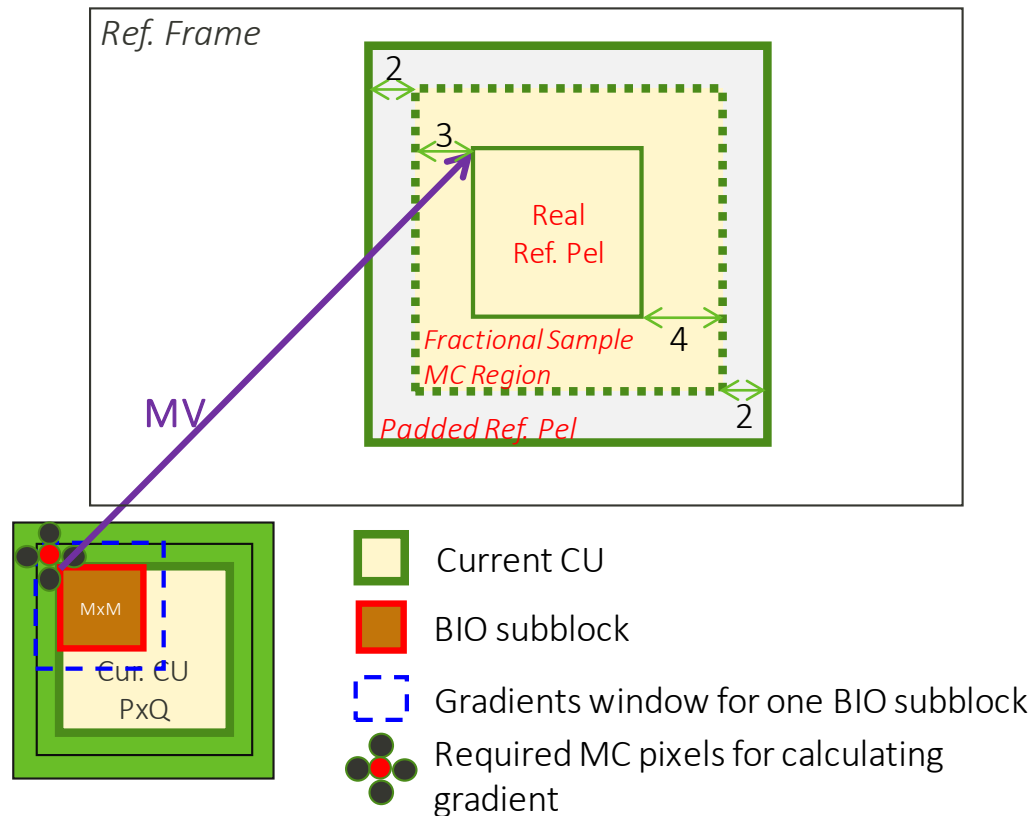
- Similar to BIO in JEM-7.0
- Refine the bi-prediction predictor by using sample gradients and the derived displacements

$$P_{\text{BIO}}[i, j] = (P^{(0)}[i, j] + P^{(1)}[i, j] + v_x[i, j](I_x^{(0)}[i, j] - I_x^{(1)}[i, j])/2 + v_y[i, j](I_y^{(0)}[i, j] - I_y^{(1)}[i, j])/2 + 1) \gg 1$$

- Difference from JEM-7.0
 - BIO with adaptive block size: 4x4 block for video resolution > 720p; 2x2 block for video resolution <= 720p or OBMC region
 - Calculate gradients by using [-1, 0, 1] filter on MC results
 - Apply padding to reduce memory bandwidth
 - Extend BIO to chroma components by reusing v_x and v_y

BIO: Memory Bandwidth Reduction

- Pad the required reference samples when they are out of $(W+7) \times (H+7)$ MC region



Overlapped Block Motion Compensation

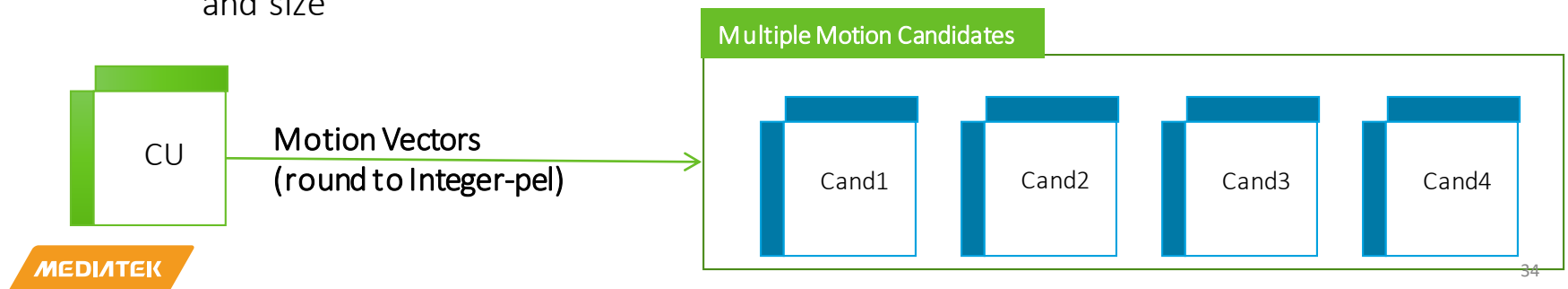
- The same as JEM-7.0

Motion Candidate Reorder

- For one CU, use template matching cost to reorder the possible motion candidates for efficient syntax signaling
 - Applied to merge and AMVP modes

Mode	Possible Motion Candidates
merge	Spatial merge candidates (non-LIC)
PMVR	PMVR merge candidates in the front (non-LIC + index ≤ 4)
AMVP	MVD sign pairs: (+, +), (+, -), (-, +), (-, -)

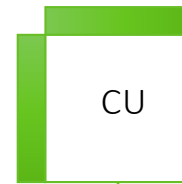
- Template:
 - CU: lines of reconstructed neighboring samples on the top and left
 - The template of the current CU and that of the motion candidate have the same shape and size



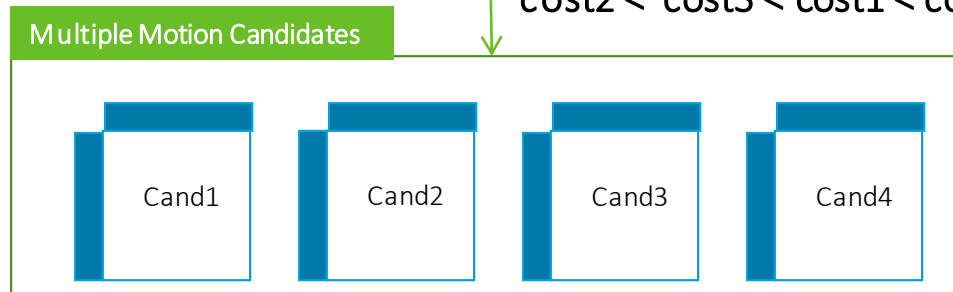
Motion Candidate Reorder

- Template matching process is to evaluate the SAD cost among all the possible motion candidates

- Merge/PMVR mode:
 - Merge index reordered by the cost
- AMVP mode:
 - MVD sign pairs reordered by table lookup

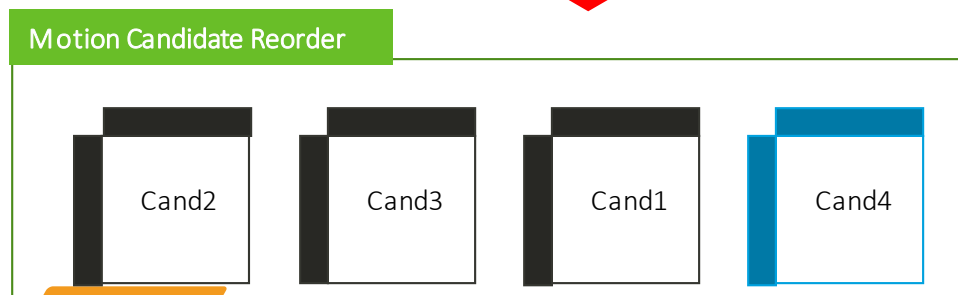


template matching process
cost2 < cost3 < cost1 < cost4



↓ Merge/PMVR

↓ AMVP



Smallest template cost position	Hit (0)	Vertical flip (10)	Horizontal flip (110)	Diagonal flip (111)
(++)	(++)	(+ -)	(- +)	(- -)
(+ -)	(+ -)	(++)	(- -)	(- +)
(- +)	(- +)	(- -)	(++)	(+ ₃₅ -)
(- -)	(- -)	(- +)	(+ -)	(++)

Motion Vector Constraint for Sub-CU Motion Vector Field (1/2)

- Several prediction modes have sub-CU motion compensation
 - Bilateral matching PMVR, affine, and ATMVP
 - Significantly increase memory bandwidth, especially when OBMC is enabled for sub-CUs
- Set a primary MV for one CU coded with sub-CU MC
 - Constrain the sub-CU MV to be within primary MV $\pm$ threshold
 - For decoder to load a larger reference block once for all sub-CUs

Motion Vector Constraint for Sub-CU Motion Vector Field (2/2)

- Primary MV
 - Bilateral matching PMVR: The CU-level best MV
 - Affine: The derived center MV of the current CU
 - ATMVP: The scaled center MV of the ATMVP collocated reference block
- Threshold
 - 4 luma samples for ATMVP and affine, when the CU area ≤ 64
 - 16 luma samples for ATMVP and affine, when the CU area > 64
 - 2 luma samples for bilateral matching PMVR

Transform and Quantization

Transform

- Transforms consist of core transform and secondary transform
- Compared to HEVC, the core transform in this proposal includes the following new features:
 - Large block-size (up to 128x128) transform with high-frequency zeroing
 - Adaptive multiple core transform
- The secondary transform uses non-separable secondary transform
- Both core transform and secondary transform contain transform syntax reorder (TSR)

Large Block Size Transform with High-frequency Zeroing

- Up to 128x128 transform
- The high frequency transform coefficients are zeroed out
 - For the transform blocks with width or height, or both width and height larger than 32
 - Only the low-frequency coefficients at the top 32 rows and left 32 columns of transform coefficients are maintained

Adaptive Multiple Core Transform (AMT)

- AMT can select the horizontal transform and vertical transform from DCT-II, DST-VII and flipped DST-VII (FDST-VII)
- An *AMT flag* is signaled
 - AMT flag == 0: DCT-II for both horizontal and vertical transform
 - AMT flag == 1: other transform type is used depending on *AMT index*

Transform Type (Transform Size: N)	Basic function $T_i(j), i, j = 0, 1, \dots, N-1$
DCT-II	$T_i(j) = \omega_0 \times \sqrt{\frac{2}{N}} \times \cos\left(\frac{\pi \times i \times (2 \times j + 1)}{2 \times N}\right)$ where $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}}, & i=0 \\ 1, & i \neq 0 \end{cases}$
DST-VII	$T_i(j) = \sqrt{\frac{4}{2 \times N + 1}} \times \sin\left(\frac{\pi \times (2 \times i + 1) \times (j + 1)}{2 \times N + 1}\right)$
FDST-VII	$T_i(j) = \sqrt{\frac{4}{2 \times N + 1}} \times \sin\left(\frac{\pi \times (2 \times i + 1) \times (N - j)}{2 \times N + 1}\right)$

Adaptive Multiple Core Transform (AMT)

- AMT is only allowed when CU width and height are both smaller than or equal to 64
- An AMT flag needs to be signaled
 - When the number of non-zero coefficients is larger than a threshold
 - For inter CU, threshold = 2
 - For intra CU, threshold = 0
- An AMT index needs to be signaled
 - When the AMT flag is equal to 1, and the number of non-zero coefficients is larger than 2

Adaptive Multiple Core Transform (AMT)

- AMT index selection and signaling for intra CU

CU		Number of AMT	AMT type	Code word
Intra CU	$W > 4$ and $H > 4$	3	Index 0 (DST-VII, DST-VII)	0
			Index 1 (DCT-II, DST-VII)	10
			Index 2 (DST-VII, DCT-II)	11
	$W > 4$ and $H \leq 4$	2	Index 0/1	0/1
	$W \leq 4$ and $H > 4$	2	Index 0/2	0/1
	$W \leq 4$ and $H \leq 4$	1	Index 0	-

Adaptive Multiple Core Transform (AMT)

- AMT index selection and signaling for inter CU

CU			Number of AMT	AMT type	Code word
Inter CU	Area > 1024 (Area = W x H)		4	Index 0 (FDST-VII, FDST-VII)	00
				Index 1 (DST-VII, FDST-VII)	10
				Index 2 (FDST-VII, DST-VII)	01
				Index 3 (DST-VII, DST-VII)	11
	Area <= 1024	W > 4 and H > 4	4	Index 0/1/2/3	Dynamic by TSR
		W > 4 and H <= 4	2	Index 0/1	
		W <= 4 and H > 4	2	Index 0/2	
		W <= 4 and H <= 4	1	Index 0	-

Non-Separable Secondary Transform (NSST)

- The process of NSST is the same as JEM-7.0
- An NSST index needs to be signaled
 - In intra slices
 - For luma CU, when the number of non-zero coefficients is larger than 1
 - For chroma CU, when the number of non-zero AC coefficients is larger than 0
 - In inter slices (luma and chroma share NSST index)
 - When the number of non-zero coefficients of luma plus the number of non-zero AC coefficients of chroma components is larger than 2
- For intra luma CU with area smaller than or equal to 512
 - NSST index is dynamically coded using transform syntax reorder (TSR)

Transform Syntax Reorder (TSR)

- Re-map the syntax for signaling multiple transforms
 - A cost is given to each transform, which is specified by AMT or NSST index
 - One with the smallest cost is chosen as the predicted transform with the shortest code word
 - For the rest transforms, code words are assigned depending on costs

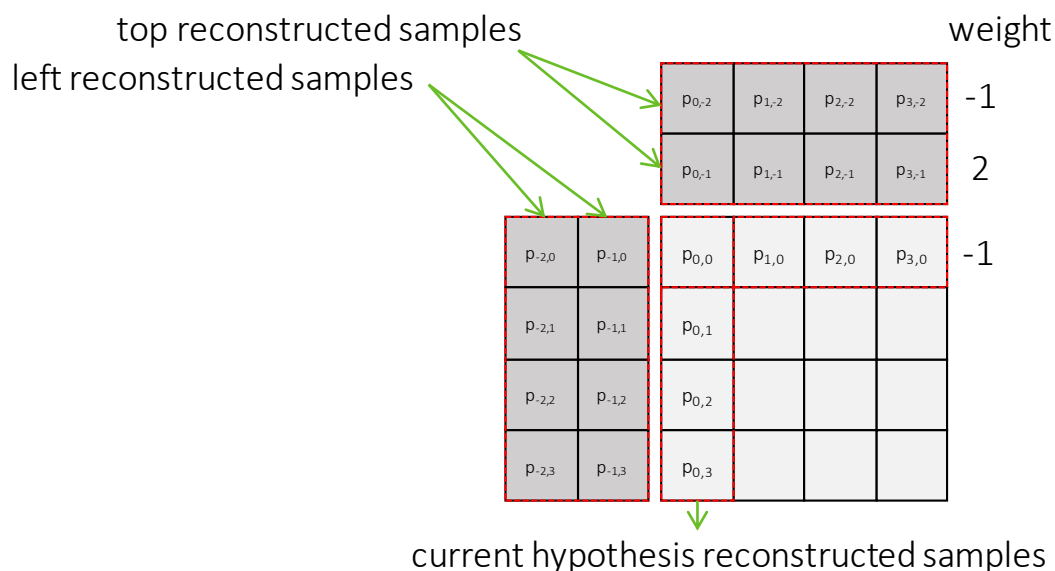
Example: index 3 transform has smallest cost

Coded transform mode index	Real transform mode index	Code word
0	3	0
1	0	10
2	1	110
3	2	111

Transform Syntax Reorder (TSR)

- Boundary matching is used to generate the costs
 - Cost: difference between the reconstructed samples of neighboring blocks and the current block for one transform

$$\text{cost} = \sum_{x=0}^{W-1} \left| (2 \times p_{x,-1} - p_{x,-2}) - p_{x,0} \right| + \sum_{y=0}^{H-1} \left| (2 \times p_{-1,y} - p_{-2,y}) - p_{0,y} \right|$$



Quantization

- The QP range is extended to [0, 57]
- Modified chroma QP mapping

qPi	< 30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	> 43
QpC	= qPi	29	30	31	32	33	33	34	35	36	37	38	39	40	41	= qPi - 2

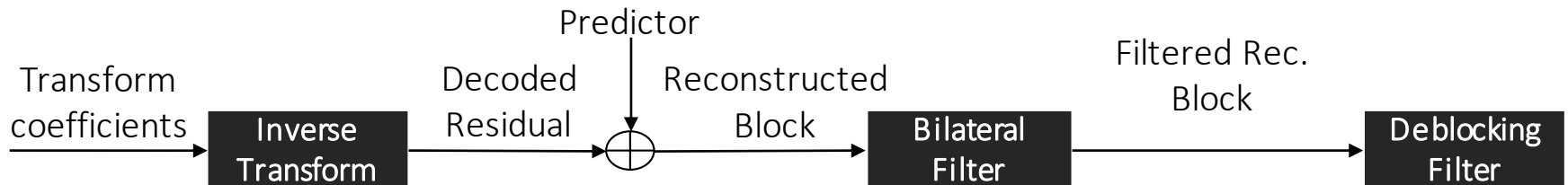
Reconstruction

Bilateral Filter

Content Adaptive Clipping

Bilateral Filter (BF)

- Similar to JEM-7.0



- Difference from JEM-7.0
 - Center weight is determined by **TU area size** and **prediction mode**:

Prediction \ TU area	16	32	64	≥ 128 & < 256	≥ 256
Intra Prediction	65	65	81	113	113
Inter Prediction	81	113	113	196	

Content Adaptive Clipping

- Same as JEM-7.0

In-loop Filter

Deblocking filter

Non-local mean loop filter

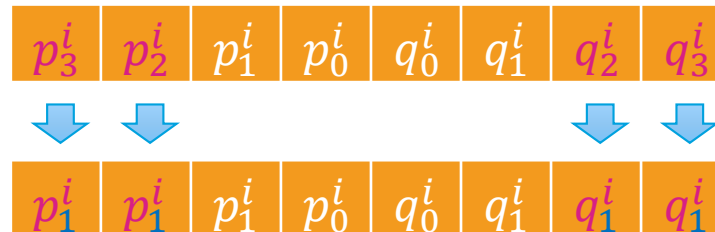
Sample adaptive offset

Adaptive loop filter

Convolution neural network loop filter

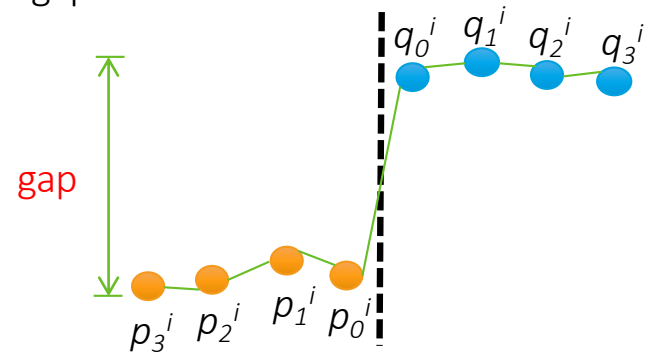
In-loop Deblocking

- Luma
 - The filtering decisions for deciding normal and strong filters are the same as HEVC
 - Include 2 major extensions for the strong filter
 - Parallel deblocking process for 4xN or Nx4 CUs
 - Length-adaptive filtering to handle blocking artifact of large CUs
- Chroma
 - Deblocking for chroma components is the same as HEVC
- Parallel deblocking for 4xN and Nx4 CUs
 - Use p1/q1 to replace the samples outside p1/q1 when doing filtering decisions and operations

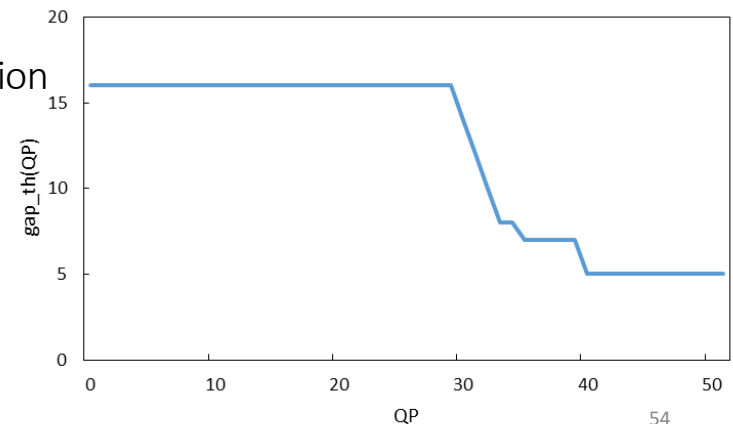


Length-adaptive Deblocking Filter

- Deblocking for large CU boundary
 - Apply longer tap filter to the boundaries with large gap value
 - 3 new long tap filters
 - L4 (R4/W4 for each side)
 - L8 (R8/W7 for each side)
 - L16 (R16/W16 for each side)

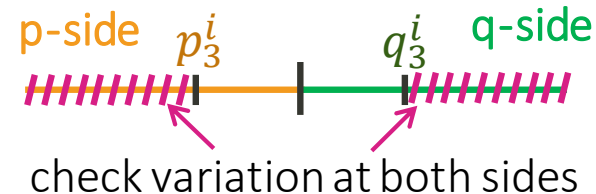


- Applied when the following conditions are trues:
 - HEVC strong filter is true
 - Inter-slice
 - Both the 1st and 4th lines meet the following condition
 - $(|p_0^i - q_1^i| + |p_1^i - q_0^i|)/2 > \text{gap_th} * \text{BitDepthScale}$,
where $i = 0$ or 3



Length-adaptive Deblocking Filter

- Set **L16/L8**, **L8/L4**, or **HEVC-Strong** label to the current segment (4-sample length boundary)
- If “ $len_p \geq 128$ and $len_q \geq 32$, or $len_p \geq 32$ and $len_q \geq 128$ ” and “variation” is small:
 - Current segment is labeled as **L16/L8**
- Otherwise
 - Current segment is labeled as **L8/L4**
- Variation
 - SAD between the 4th sample and a farther sample from the boundary on the first line and the 4th line



$$\sum_{s=4}^{m-1} |p_s^i - p_3^i| + \sum_{t=4}^{n-1} |q_t^i - q_3^i| < ss_{th}$$

$$ss_{th} = \left(\beta \times \frac{((m-4) + (n-4))}{\text{\# of outside samples}} \right) \gg 4,$$

$$\text{where } i = 0 \text{ and } 3, \text{ and } \begin{cases} m = 16 \text{ and } n = 16, \text{ if } len_p \geq 128 \text{ and } len_q \geq 128 \\ m = 16 \text{ and } n = 8, \text{ if } len_p \geq 128 \text{ and } 128 > len_q \\ m = 8 \text{ and } n = 16, \text{ if } 128 > len_p \text{ and } len_q \geq 128, \end{cases}$$

Check Filter Decision Coherence

- L16/L8 coherence check
 - When a current segment is labeled as L16/L8, check if any four consecutive segments containing the current segment are all L16/L8
 - Yes: No label change (use L16/L8)
 - No: Re-label the current segment as L8/L4
- L8/L4 coherence check
 - When a current segment is labeled as L8/L4, check if any two consecutive segments containing the current segment are both L8/L4
 - Yes: No label change (use L8/L4)
 - No: Re-label the current segment as HEVC-Strong

Length-adaptive Deblocking Filter

- If labeled as **L16/L8**:
$$\left\{ \begin{array}{l} \text{For p-side:} \left\{ \begin{array}{l} \text{When } len_p \geq 128 \rightarrow \text{apply L16} \\ \text{Otherwise} \rightarrow \text{apply L8} \end{array} \right. \\ \text{For q-side:} \left\{ \begin{array}{l} \text{When } len_q \geq 128 \rightarrow \text{apply L16} \\ \text{Otherwise} \rightarrow \text{apply L8} \end{array} \right. \end{array} \right.$$
- If labeled as **L8/L4**:
$$\left\{ \begin{array}{l} \text{For p-side:} \left\{ \begin{array}{l} \text{When } len_p \geq 16 \rightarrow \text{apply L8} \\ \text{When } 16 > len_p \geq 8 \rightarrow \text{apply L4} \\ \text{Otherwise} \rightarrow \text{apply the strong filtering in HEVC} \end{array} \right. \\ \text{For q-side:} \left\{ \begin{array}{l} \text{When } len_q \geq 16 \rightarrow \text{apply L8} \\ \text{When } 16 > len_q \geq 8 \rightarrow \text{apply L4} \\ \text{Otherwise} \rightarrow \text{apply the strong filtering in HEVC} \end{array} \right. \end{array} \right.$$
- If labeled as **HEVC-Strong**: The strong filtering in HEVC is applied

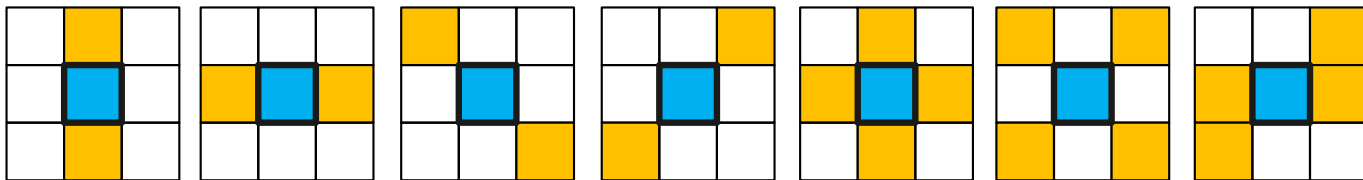


Non-Local Mean Loop Filter (NLMLF)

- NLMLF includes two image denoising technologies, non-local mean (NLM) and singular value decomposition with low rank optimization (LowRank-SVD)
- Divide one slice into several patches with size equal to 8x8
- Receive on/off control flags (slice, CTB, and 32x32 block levels)
- For each to-be-processed patch
 - Find 15 nearest patches in the search range to form one patch group
 - Apply NLM or LowRank-SVD to modify samples in one patch group
 - Accumulate modified sample value and weight for every sample in one patch group
- Normalize the accumulated sample value with accumulated weight for every sample in the frame

Sample Adaptive Offset (SAO)

- Similar to JEM-7.0
- Difference from JEM-7.0
 - Add three 2D EO patterns
 - Remove sign constraints in EO modes: the signs of EO offsets are signaled instead of inferred as in JEM-7.0



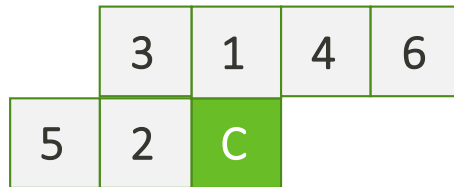
Adaptive Loop Filter (ALF)

- Propose CTB-based ALF with slice-level filter sets
- 4 modes are supported for each CTB
 - *Off*: disable ALF
 - *NewFilter*: signal one new filter
 - *MergeFilter*: inherit one filter from one selected neighboring CTB
 - *SliceFilter*: use one selected slice-level filter set, where one filter set contains multiple filters for Y, one for Cb, and one for Cr

ALF: *NewFilter* & *MergeFilter* Modes

- *NewFilter* mode
 - Signal one new filter
 - Apply the new filter to every sample in the current CTB

- *MergeFilter* mode
 - Inherit one filter from one selected neighboring CTB



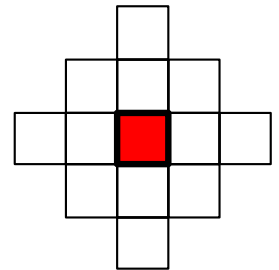
- Apply the inherited filter to every sample in the current CTB

ALF: *SliceFilter* Mode

- Use one selected slice-level filter set
- Perform sample classification in the CTB
- Apply a filter in the filter set to each sample based on the classification
- Filter sets at slice header
 - Store up to 4 decoded filter sets in a FIFO
 - Add one new filter set into the FIFO if signaled
 - One filter set: contain up to 16 filters and signal the selected sample classifier if necessary

Sample Classifiers in *SliceFilter* Mode

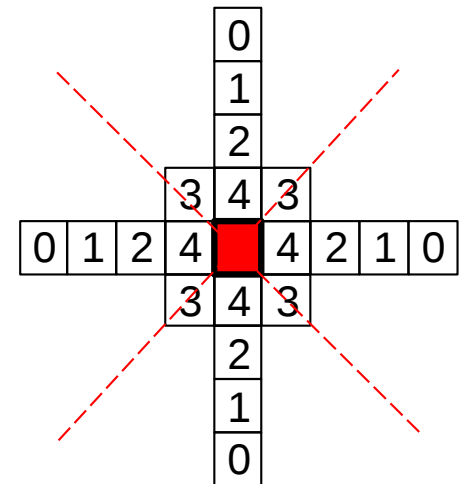
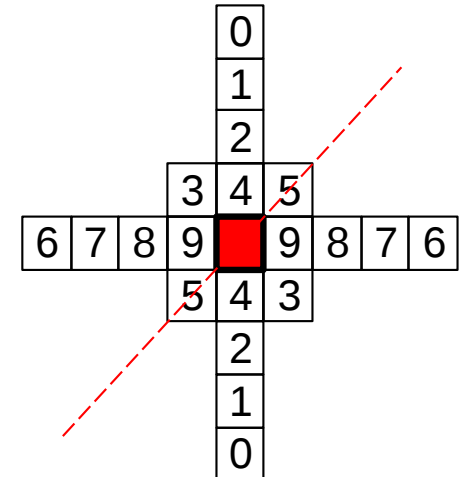
- Intensity Classifier
 - Divide the range of sample intensity into 32 groups (similar to BO in SAO)
 - Decide a group index (0-31) according to sample intensity for each sample
- Histogram Classifier
 - Calculate the differences between current and neighboring samples
 - Decide a group index (0-24) according to the diffCount for each sample
- Directional Activity Classifier
 - Similar to JEM-7.0
 - Decide a group index (0-24) for each 2x2 block



If ($\text{abs}(\text{curPel} - \text{neiPel}) > \text{TH}$)
 diffCount ++ ;
If ($\text{abs}(\text{curPel} - \text{neiPel}) > 2 * \text{TH}$)
 diffCount ++ ;

ALF: Filter Footprints

- One filter shape
 - 9x9 Cross + 3x3 Square
- Two symmetry types
 - Half symmetry: 10 signaled coefficients
 - Full symmetry: 5 signaled coefficients
 - Center coefficient is derived for filter gain is 1
 - No offset
- Coefficients
 - 8 bits
 - Center: [0, 255]; others: [-128, 127]

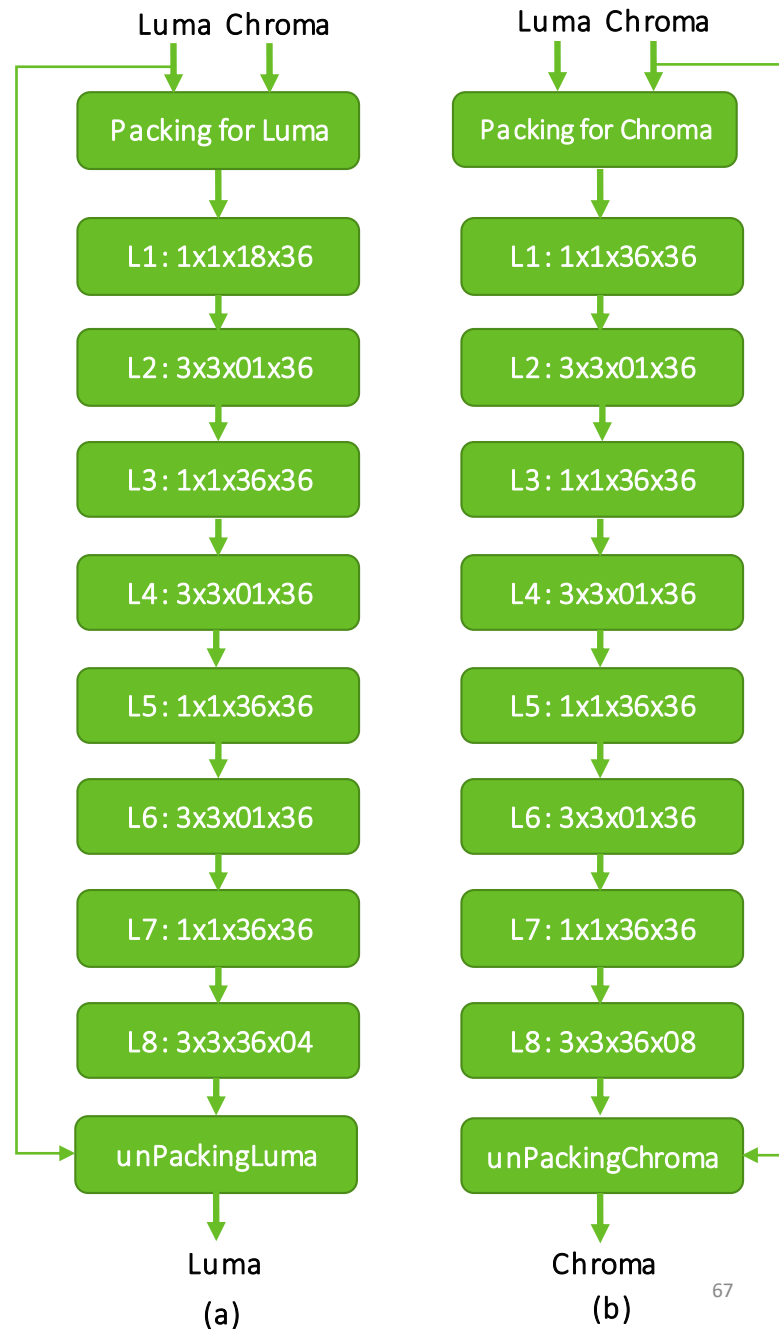
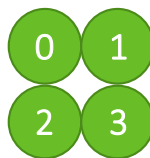


Convolution Neural Network Loop Filter (CNNLF)

- To reduce sample distortions
 - The CNN outputs are estimated errors between uncompressed samples and reconstructed samples
 - Add CNN outputs and reconstructed samples as CNNLF outputs
- Multi-type CNN inputs: reconstructed samples (ALF output), predictors, and residuals of Y, Cb, and Cr
- One CNN for Y and another shared CNN for Cb and Cr
- All fixed-point operations
- Multi-level on/off control (slice, CTB, and 32x32 block levels)
- Signal CNN parameters at PPS for each test point

CNN Structure

- $LK: M \times M \times N \times P$
 - K is layer id
 - $M \times M$ is spatial filter shape
 - N is number of input channels
 - P is number of filters
- Apply Rectified Linear Unit (ReLU) after each layer except the output layer
- Packing
 - 2x2 sub-sampling to 4 sub-frames
 - Pack into 3D input data
- Unpacking
 - Integrate 4 filtered sub-frames into one output frame



MEDIATEK

Entropy Coding

Multi-parameter CABAC with Reduced RangeOne Table

- Use modified multi-parameter CABAC in JEM-7.0
- Only store the RangeOne value of probability (P) in [0, 0.5]
 - For [0.5, 1.0], use 1.0-P to derive the RangeZero
 - $\text{RangeOne} = \text{range} - \text{RangeZero}$
 - Use an 8-bit*8-column*32-entry lookup table (144x smaller than the RangeOne lookup table in JEM-7.0)

$\text{probLPS} = (P \geq 16384) ? 32767 - P : P$

$\text{rangeLPS} = \text{LUT}[\text{probLPS} \gg 9][(\text{range} \gg 5) \& 7]$

$\text{rangeOne} = (P \geq 16384) ? \text{range} - \text{rangeLPS} : \text{rangeLPS}$

$\text{rangeZero} = (P \geq 16384) ? \text{rangeLPS} : \text{range} - \text{rangeLPS}$

	(Range>>5)&7			
	0	1	...	7
probLPS>>9	Range LPS			
31	136	152	...	248
...	...	...	...	...
23	102	114	...	186
22	97	109	...	178
21	93	104	...	170
20	89	99	...	162
19	85	95	...	155
...	...	...	...	...
0	4	4	...	7

Encoder Algorithms

Hierarchical GOP Structure

- Use the same hierarchical GOP structure as JEM-7.0
 - Hierarchical GOP size = 16 for constraint set 1
 - Hierarchical GOP size = 4 for constraint set 2
- Only change the configuration parameters, Qpoffset, for Temporal ID equal to 0 and 2

	JEM-7.0	Modification
Temporal ID = 0	1	0
Temporal ID = 2	4	2

Adaptive Encoder Flow for QT/BT

- QT, BT partitions are tested in an adaptive order
 - QT partition is tested before BT partition
 - If the size of a current CU is larger than 32
 - If the current QT depth is not larger than the minimum QT depth of its neighboring CUs
 - Otherwise BT partition is tested before QT partition
 - TT partition is always tested last
- The results of partitions tested first can be used to skip later partitions

Additional Encoding Pass for CNN Training

- Additional encoding pass is required for CS1 configuration
 - Generate the training data for CNN, including reconstructed samples (ALF output), predictors, and residuals in the first pass encoding (CNNLF disabled)
- After deriving CNN parameters
 - Signal CNN parameters at PPS
 - Generate the final bitstream in the second pass (CNNLF enabled)
- Disable CNNLF for CS2

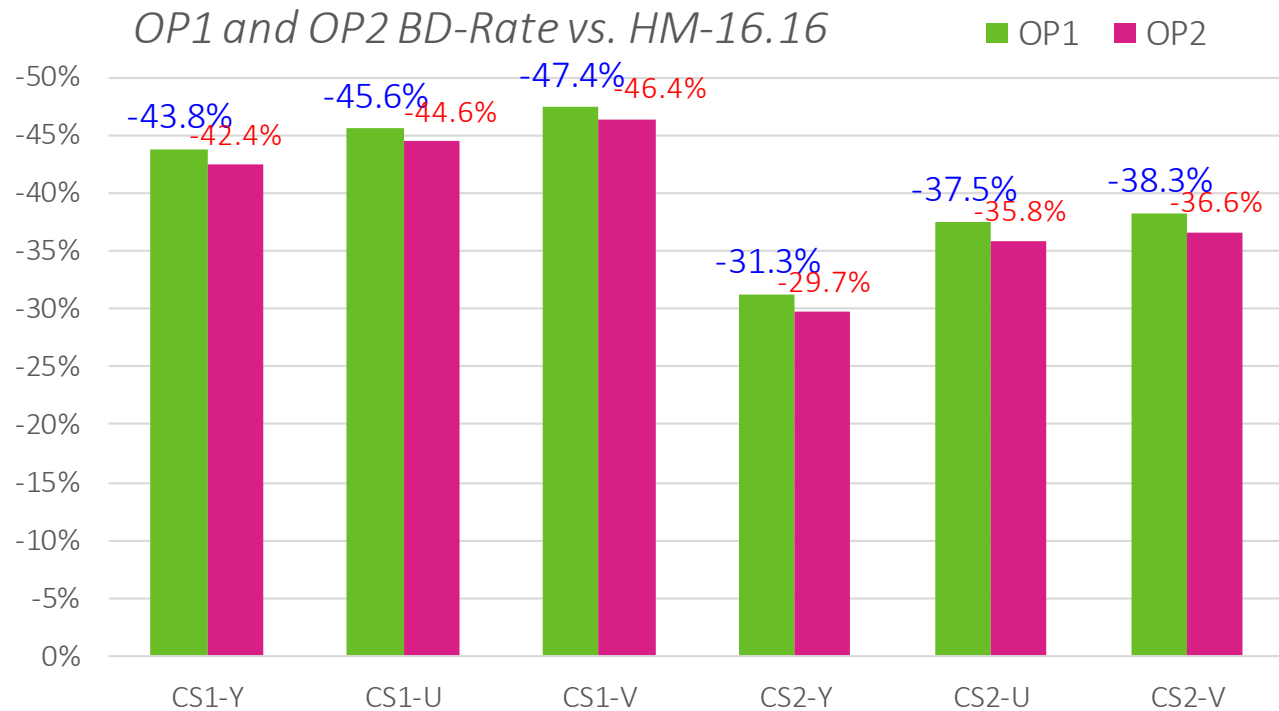
Settings in Operation Point 1 (OP1) and Operation Point 2 (OP2)

- OP1 pursues high compression performance
- OP2 pursues good compression performance with relatively low encoding time
 - The differences between the two operation points are merely non-normative encoder-only changes; Same decoders
 - The OP1 bitstreams were submitted for subjective evaluation
 - The detailed performance regarding BD-rates and run time percentages for both OP1 and OP2 are shown in the following

MEDIATEK

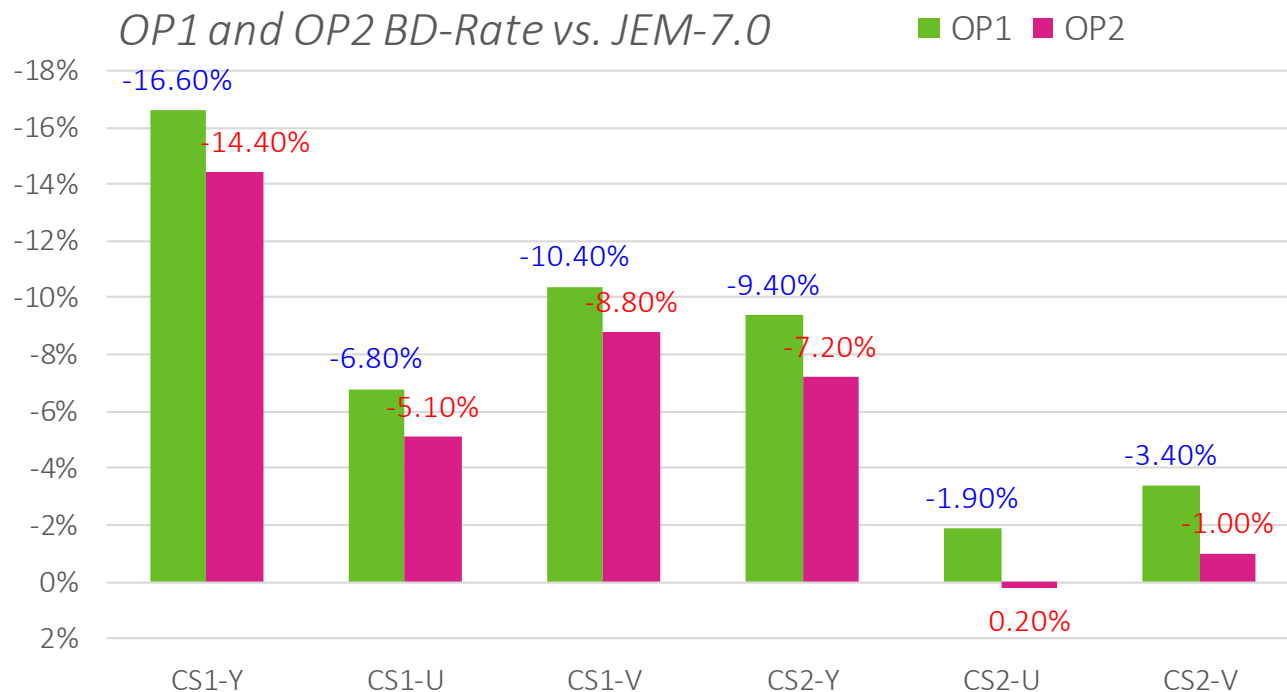
Performance

Comparison with HM-16.16



Anchor: HM-16.16		Constraint Set1			Constraint Set2		
		Y	U	V	Y	U	V
OP1	BD-rate	-43.8%	-45.6%	-47.4%	-31.3%	-37.5%	-38.3%
	run time	enc: 1235%; dec: 1300%			enc: 1192%; dec: 929%		
OP2	BD-rate	-42.4%	-44.6%	-46.4%	-29.7%	-35.8%	-36.6%
	run time	enc: 610%			enc: 530%		

Comparison with JEM-7.0



Anchor: JEM-7.0		Constraint Set1			Constraint Set2		
		Y	U	V	Y	U	V
OP1	BD-rate	-16.6%	-6.8%	-10.4%	-9.4%	-1.9%	-3.4%
	run time	enc: 158%; dec: 227%			enc: 139%; dec: 177%		
OP2	BD-rate	-14.4%	-5.1%	-8.8%	-7.2%	+0.2%	-1.0%
	run time	enc: 78%			enc: 62%		

Comments on CNNLF

- After CNN training, the enc. time increase for CNNLF is low
 - Only 1% increase in the 2nd pass encoding
- The multi-pass encoding can be significantly accelerated
 - Training needs shorter than 2 hours on a GPU
 - The 1st pass encoding intermediate results (e.g., mode decisions, costs, ...) can be considered for the 2nd pass encoding in practical cases
- The dec. time increase without using any GPUs is 50%
 - In the near future GPUs are available in most devices
 - CNNLF can be easily accelerated, e.g., 196x in JVET-I0022
- We suggest to standardize predefined CNNLF parameter sets in addition to allowing signaled CNNLF parameter sets

Memory Usage and Raster Scan

- Memory usage (FoodMarket4 at RatePoint1)

	HM-16.16	JEM-7.0	Proposal
Encoding	1.84 GB	5.57 GB	3.29 GB
Decoding	0.98 GB	1.26 GB	0.87 GB

- Memory bandwidth

	HM-16.16	JEM-7.0	Proposal
MC	1 x	45.9 x	3.1 x

- Only raster scan is used
 - Remove double z-scan to simplify the address mapping

MEDIATEK

Conclusion

Conclusion

- 43.81% and 31.27% luma BD-rate savings over HM-16.16 for CS1 and CS2, respectively
- 16.60% and 9.41% BD-rate savings over JEM-7.0 for CS1 and CS2, respectively
- Recommend considering our tools in the starting framework
 - One of the best performing proposals in terms of BD-rate savings
 - JEM based; experts are familiar with JEM
 - Low memory usage
 - Tools can be turned on/off by SPS flags
 - Include many promising techniques up to date

Backup

Performance of OP1

	Constraint Set 1					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-39.99%	-36.21%	-34.85%	2.07	1.02	1.06
CatRobot1	-48.11%	-57.27%	-50.55%	1.95	1.07	1.49
DaylightRoad2	-48.08%	-65.92%	-58.63%	1.38	1.53	1.25
ParkRunning3	-49.48%	-22.05%	-38.91%	1.86	0.36	0.61
Campfire	-59.04%	-16.42%	-69.66%	1.96	0.54	2.55
BQTerrace	-39.91%	-61.48%	-67.02%	1.24	1.38	1.51
RitualDance	-34.41%	-39.63%	-40.86%	1.92	1.16	1.39
MarketPlace	-36.51%	-47.81%	-33.26%	1.35	0.93	0.65
BasketballDrive	-40.49%	-51.91%	-39.08%	1.52	1.21	1.17
Cactus	-42.09%	-57.34%	-41.23%	1.81	1.04	1.12
Average SDR-A	-48.94%	-39.58%	-50.52%	1.84	0.90	1.39
Average SDR-B	-38.68%	-51.63%	-44.29%	1.57	1.14	1.17
Average all	-43.81%	-45.61%	-47.41%	1.71	1.02	1.28
Enc Time[%]	1235%					
Dec Time[%]	1300%					
Max Delta Rate[%]	0.00%					
Min Delta Rate[%]	-0.70%					
Max Delta PSNR (YUV)	2.64		1.65		2.72	
Min Delta PSNR (YUV)	0.87		0.27		0.22	

	Constraint Set 2					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
BQTerrace	-33.16%	-48.93%	-60.78%	1.02	0.65	0.83
RitualDance	-27.24%	-35.28%	-35.58%	1.45	0.94	1.12
MarketPlace	-27.94%	-22.61%	-19.67%	0.93	0.33	0.32
BasketballDrive	-32.50%	-46.22%	-42.27%	1.27	1.09	1.37
Cactus	-35.52%	-34.64%	-33.05%	1.49	0.55	0.80
Average SDR-B	-31.27%	-37.54%	-38.27%	1.23	0.71	0.89
Enc Time[%]	1192%					
Dec Time[%]	929%					
Max Delta Rate	0.00%					
Min Delta Rate	-0.51%					
Max Delta PSNR (YUV)	1.76		1.42		1.75	
Min Delta PSNR (YUV)	0.72		-0.10		-0.13	

Constraint Set 1					
Over JEM7.0					
BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
-10.60%	10.57%	16.10%	0.41	-0.24	-0.41
-15.12%	-11.07%	-9.49%	0.41	0.12	0.20
-13.06%	-21.75%	-30.11%	0.25	0.30	0.49
-27.26%	4.92%	-14.71%	0.87	-0.07	0.22
-40.18%	26.84%	-41.49%	0.96	-0.75	1.03
-13.42%	-30.13%	-30.27%	0.33	0.56	0.48
-9.98%	-4.95%	-3.12%	0.48	0.11	0.07
-10.68%	-6.99%	7.27%	0.32	0.10	-0.15
-14.06%	-12.29%	5.12%	0.41	0.21	-0.14
-11.63%	-22.61%	-3.64%	0.37	0.32	0.05
-21.24%	1.90%	-15.94%	0.58	-0.13	0.31
-11.95%	-15.39%	-4.93%	0.38	0.26	0.06
-16.60%	-6.75%	-10.43%	0.48	0.07	0.18
			158%		
			227%		
1.19		0.81		1.11	
0.16		-0.85		-0.88	

Constraint Set 2					
Over JEM7.0					
BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
-9.98%	-11.31%	-22.65%	0.24	0.10	0.21
-7.23%	-5.82%	-5.60%	0.34	0.11	0.14
-9.32%	12.61%	13.28%	0.28	-0.22	-0.26
-9.65%	-5.95%	-2.67%	0.30	0.11	0.07
-10.84%	0.88%	0.88%	0.36	-0.06	-0.08
-9.41%	-1.92%	-3.35%	0.30	0.01	0.02
			139%		
			177%		
0.53		0.60		0.73	
0.12		-0.75		-0.89	

Performance of OP2

	Constraint Set 1					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-38.57%	-34.92%	-33.44%	1.98	0.97	1.00
CatRobot1	-46.61%	-55.91%	-48.78%	1.88	1.04	1.42
DaylightRoad2	-46.19%	-64.92%	-57.47%	1.32	1.49	1.21
ParkRunning3	-48.18%	-20.53%	-37.73%	1.79	0.33	0.58
Campfire	-57.70%	-15.05%	-68.90%	1.91	0.49	2.50
BQTerrace	-38.46%	-61.63%	-67.27%	1.19	1.38	1.51
RitualDance	-32.95%	-38.73%	-39.98%	1.82	1.12	1.34
MarketPlace	-35.16%	-47.18%	-32.40%	1.28	0.92	0.64
BasketballDrive	-39.18%	-51.05%	-37.65%	1.46	1.18	1.12
Cactus	-40.84%	-56.53%	-40.11%	1.74	1.02	1.08
Average SDR-A	-47.45%	-38.27%	-49.26%	1.77	0.87	1.34
Average SDR-B	-37.32%	-51.02%	-43.48%	1.50	1.12	1.14
Average all	-42.38%	-44.64%	-46.37%	1.64	0.99	1.24
Enc Time[%]	610%					
Dec Time[%]	1346%					
Max Delta Rate[%]	0.00%					
Min Delta Rate[%]	-1.05%					
Max Delta PSNR (YUV)	2.54		1.62		2.66	
Min Delta PSNR (YUV)	0.84		0.24		0.18	

	Constraint Set 2					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
BQTerrace	-31.71%	-48.68%	-59.97%	0.97	0.66	0.81
RitualDance	-25.61%	-33.45%	-33.66%	1.34	0.88	1.04
MarketPlace	-26.07%	-19.99%	-18.35%	0.86	0.28	0.29
BasketballDrive	-30.99%	-44.09%	-40.15%	1.20	1.02	1.29
Cactus	-34.07%	-32.86%	-30.90%	1.41	0.51	0.74
Average SDR-B	-29.69%	-35.82%	-36.60%	1.16	0.67	0.83
Enc Time[%]	530%					
Dec Time[%]	939%					
Max Delta Rate	-0.01%					
Min Delta Rate	-0.57%					
Max Delta PSNR (YUV)	1.65		1.34		1.69	
Min Delta PSNR (YUV)	0.66		-0.13		-0.17	

	Constraint Set 1					
	Over JEM7.0					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
	-8.37%	12.61%	18.26%	0.32	-0.29	-0.47
	-12.54%	-8.39%	-6.51%	0.34	0.09	0.13
	-9.88%	-19.13%	-28.02%	0.19	0.27	0.46
	-25.35%	6.76%	-13.06%	0.80	-0.09	0.20
	-38.14%	28.92%	-39.87%	0.90	-0.80	0.98
	-11.42%	-30.42%	-30.82%	0.28	0.56	0.48
	-7.93%	-3.37%	-1.64%	0.37	0.07	0.02
	-8.71%	-6.51%	8.04%	0.26	0.09	-0.17
	-12.06%	-10.81%	7.28%	0.35	0.18	-0.19
	-9.59%	-20.98%	-1.92%	0.30	0.30	0.01
	-18.86%	4.16%	-13.84%	0.51	-0.16	0.26
	-9.94%	-14.42%	-3.81%	0.31	0.24	0.03
	-14.40%	-5.13%	-8.82%	0.41	0.04	0.14
	78%					
	235%					
	1.12		0.79		1.07	
	0.13		-0.92		-0.91	

	Constraint Set 2					
	Over JEM7.0					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
	-7.87%	-12.21%	-20.99%	0.18	0.10	0.19
	-5.09%	-3.88%	-3.45%	0.24	0.06	0.06
	-6.90%	16.50%	15.51%	0.20	-0.28	-0.29
	-7.52%	-2.85%	-0.09%	0.23	0.04	-0.01
	-8.63%	3.57%	4.21%	0.28	-0.09	-0.15
	-7.20%	0.23%	-0.96%	0.23	-0.03	-0.04
	62%					
	179%					
	0.43		0.53		0.65	
	0.05		-0.78		-0.92	

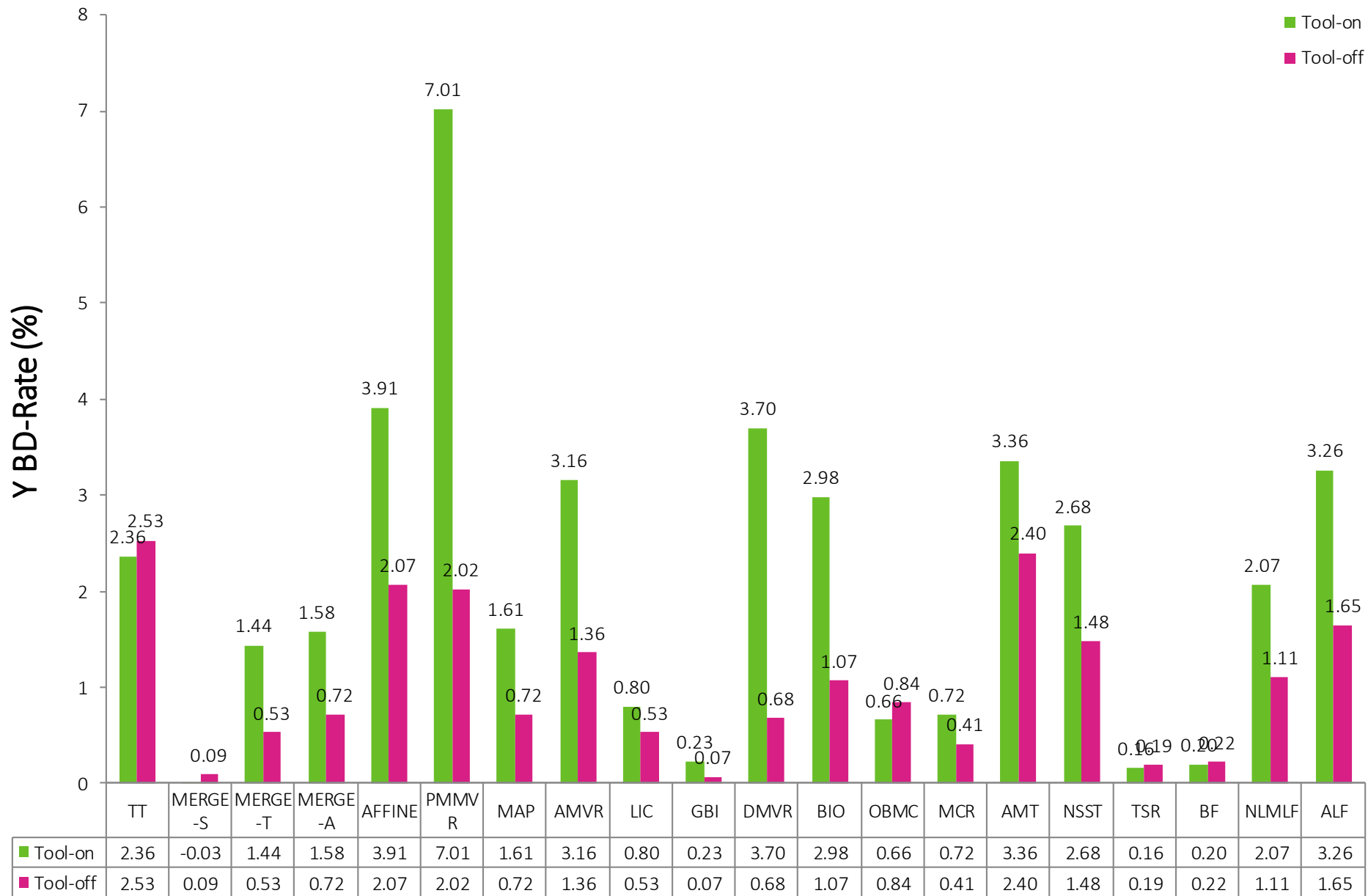
CS1 and OP1

Tested tool name	“single tool-off” test results					“single tool-on” test results				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
TT	2.53%	4.35%	4.61%	60%	100%	-2.36%	-4.27%	-4.06%	160%	103%
MERGE-S	0.09%	0.00%	0.00%	101%	98%	0.03%	-0.11%	0.18%	100%	101%
MERGE-T	0.53%	0.42%	0.58%	102%	96%	-1.44%	-1.23%	-1.34%	102%	141%
MERGE-A	0.72%	0.44%	0.57%	101%	95%	-1.58%	-1.70%	-1.40%	100%	107%
AFFINE	2.07%	1.52%	1.69%	95%	96%	-3.91%	-2.89%	-2.86%	120%	103%
PMVR	2.02%	2.13%	2.18%	89%	91%	-7.01%	-6.31%	-6.47%	125%	201%
MAP	0.72%	0.63%	0.64%	87%	98%	-1.61%	-1.74%	-1.78%	132%	104%
AMVR	1.36%	1.98%	2.03%	94%	98%	-3.16%	-3.63%	-3.57%	110%	100%
LIC	0.53%	0.03%	0.04%	100%	96%	-0.80%	-0.02%	-0.14%	101%	100%
GBI	0.07%	0.01%	0.09%	96%	98%	-0.23%	-0.40%	-0.11%	110%	101%
DMVR	0.68%	0.45%	0.55%	96%	94%	-3.70%	-3.07%	-3.14%	116%	151%
BIO	1.07%	0.77%	0.82%	96%	86%	-2.98%	-2.38%	-2.35%	108%	134%
OBMC	0.84%	0.96%	1.08%	96%	85%	-0.66%	-1.32%	-1.17%	103%	112%
MCR	0.41%	0.82%	0.47%	100%	98%	-0.72%	-0.63%	-0.80%	101%	102%
AMT	2.40%	1.79%	1.93%	77%	96%	-3.36%	-2.93%	-2.85%	119%	100%
NSST	1.48%	3.32%	3.29%	87%	97%	-2.68%	-4.87%	-4.43%	117%	100%
TSR	0.19%	0.01%	-0.16%	97%	96%	-0.16%	-0.41%	0.05%	104%	106%
BF	0.22%	-0.07%	-0.11%	98%	98%	-0.20%	-0.15%	0.57%	101%	101%
NLMLF	1.11%	5.81%	5.62%	100%	61%	-2.07%	-5.12%	-4.76%	106%	329%
ALF	1.65%	3.43%	3.28%	100%	94%	-3.26%	-2.72%	-2.58%	102%	124%

CS2 and OP1

Tested tool name	“single tool-off” test results					“single tool-on” test results				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
TT	2.36%	5.17%	5.16%	59%	99%	-2.20%	-5.81%	-6.22%	166%	100%
MERGE-S	-0.03%	0.29%	0.16%	101%	103%	-0.02%	0.12%	-0.06%	100%	100%
MERGE-T	0.97%	1.30%	1.00%	102%	95%	-0.72%	-1.48%	-0.83%	105%	130%
MERGE-A	0.62%	1.19%	1.24%	101%	95%	-1.42%	-1.00%	-0.87%	101%	104%
AFFINE	2.18%	2.11%	2.22%	88%	97%	-3.23%	-2.71%	-2.49%	144%	102%
PMVR	0.91%	1.12%	1.38%	89%	97%	-2.47%	-3.20%	-3.28%	152%	148%
MAP	0.87%	1.27%	1.51%	91%	102%	-1.04%	-2.96%	-3.22%	135%	106%
AMVR	0.78%	1.31%	1.37%	90%	101%	-1.61%	-2.38%	-2.41%	132%	99%
LIC	0.93%	0.23%	0.39%	101%	98%	-1.34%	-0.69%	-0.32%	101%	100%
GBI	0.07%	-0.32%	0.51%	92%	99%	-0.19%	-0.26%	-0.25%	125%	101%
OBMC	0.96%	1.09%	1.28%	97%	87%	-0.82%	-1.33%	-1.39%	105%	112%
MCR	0.29%	0.55%	0.14%	100%	98%	-0.38%	-0.57%	-0.33%	103%	103%
AMT	2.09%	0.01%	0.24%	76%	98%	-2.67%	-1.63%	-1.44%	126%	100%
NSST	1.17%	3.55%	4.02%	89%	100%	-2.06%	-4.78%	-4.27%	125%	101%
TSR	0.28%	0.27%	0.25%	97%	97%	-0.25%	0.31%	0.33%	104%	103%
BF	0.16%	-0.26%	0.21%	99%	99%	-0.16%	-0.07%	0.13%	102%	101%
NLMLF	1.30%	9.73%	9.37%	100%	49%	-2.23%	-6.83%	-7.02%	111%	379%
ALF	2.32%	9.58%	6.62%	101%	94%	-3.03%	-4.07%	-3.75%	102%	124%

CS1 and OP1



CS2 and OP1

