



CE12: SK Telecom/SKKU Deblocking Filter

J. Yang, K. Won, B. Jeon (SKKU), J. Lim (SK Telecom)

JCTVC-F258

Overall summary (1)

- Prior proposals: JCTVC-C130, D334, E417 ~ CE12 (Deblocking filter)
- Key concepts:
 - Simple unified deblocking structure for luma and chroma
 - Same 8x8 processing, same filter, and same deblocking mechanism for luma and chroma
 - Minimum number of filters : strong filter for luma(1), weak filter for luma&chroma (1)
 - ➔ Enhanced reusability of HW (SW) blocks (function)
 - Unified concept in bS decision bet. intra and inter
 - Inter : bS = 0 (same inter prediction case), 1 (different inter prediction case)
 - Intra : bS = 3 (same intra prediction case), 4 (different intra prediction case)
- Performance (BDBR gain & Complexity):
 - 1.2% (HE_IO), 1.1% (HE_RA), 1.6% (HE_LD)
 - 1.0% (LC_IO), 0.8% (LC_RA), 1.0% (LC_LD)
 - No increase of decoding time compared to HM3.0.
- Cross-verification: (greatly appreciated)
 - JCTVC-F218 [Sony], JCTVC-F529 [Ericsson], JCTVC-F625 [ETRI]

Overall summary (2)

- Item1: Unified concept in bS decision bet. intra and inter
 - bS derivation for intra blocks depends on relative intra-prediction directions of P&Q in the same spirit as inter prediction → makes unified design
 - tc_offset for intra block is either 1 or 3 depending on the bS strength → gives better rendition of edges
- Item2: Filter type decision for luma
 - A condition for strong filtering is added: one decision for 8-pel boundary
 - The condition for the 8-pel edge is the same test as in a test for filtering the block boundary as in HM, but shifted 1 pixel away from the block boundary, i.e., $|p_3 - 2 \cdot p_2 + p_1|$. → reuse of same logic
- Item3: modified weak filter for luma
 - The blocking pattern of N is solved with just one filter instead of two filters
 - Separate decisions whether to filter one or two pixels on each side of the block boundary (on 8-line edge level).
 - Δ or half of Δ values are used for p_0 and q_0 depending on the fact if one or two pixels are filtered

Motivation: Why blocking artifacts? [P.List]

- Reason1: Independent block-based transform and quantization over P & Q
 - Therefore, no deblocking needed in following cases
 - inside TU(*however*, blocking artifacts can be carried inside due to prediction)
 - P, Q blocks both without coded transform coefficients
 - P, Q blocks both with no transform
- Reason2: Different prediction over P & Q
 - Note it can be carried inside block interior
 - How to know whether or not different prediction
 - Inter blocks: different MV and/or reference picture (well addressed in HM3.0)
 - Intra blocks: different intra prediction direction (not addressed in HM3.0)
 - Current HEVC for inter blocks
 - bS can be either 1 (different inter prediction) or 0 (same inter prediction)
 - Current HEVC for intra blocks
 - bS not well reflecting difference in intra prediction → *What we like to address*

Ref: P. List and et. Al, "*Adaptive Deblocking Filter*," IEEE Trans. On CSVT, July 2003, pp. 614-619.

Luma bS decision – HM3.0

- For intra coded block, bS decision and t_c control of HM3.0

Conditions	bS	t_c offset
P and/or Q are intra coded block, and its boundary is CU boundary	4	4
P and/or Q are intra coded block, and its boundary is not CU boundary	3	4
P and Q are inter coded block, and P and/or Q has non-zero coded coefficients	2	0
P and Q are inter coded block, and P and Q have no non-zero coded coefficients, and P and Q refer to the different ref. pic., or have MV values that differ by one sample or more	1	0
Otherwise	0	n/a

Luma bS decision – Proposed

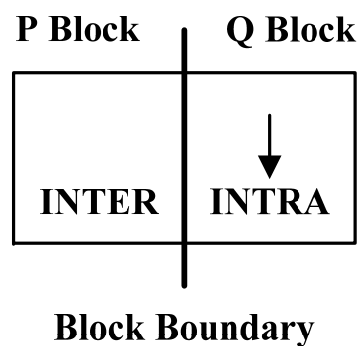
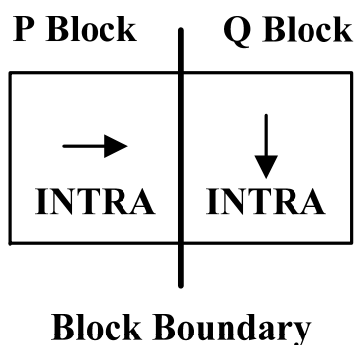
- For intra coded block, bS decision and t_c control of the proposed method

Conditions	bS	t_c offset	wrt HM3.0
P and Q have <i>different</i> intra prediction modes*	4	3	Modified
P and Q have the <i>same</i> intra prediction mode*	3	1	Modified
P and Q are <i>inter</i> coded block, <i>and</i> P and/or Q has non-zero coded coefficients	2	0	No change
P and Q are <i>inter</i> coded block, <i>and</i> P and Q have no non-zero coded coefficients, <i>and</i> P and Q refer to the different ref. pic., or have MV values that differ by one sample or more	1	0	No change
Otherwise	0	n/a	No change

* If only one of P and Q is intra coded, then treat the case as having *different* intra prediction modes.

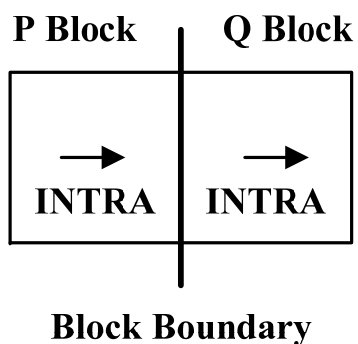
Illustration: Luma bS decision

- Example of bS=4 case



If only one of P and Q is intra coded, then treat the case as having *different* intra prediction modes

- Example of bS=3 case



Visual Comparison
BasketballDrive_IOLC_QP37 – 157th frame



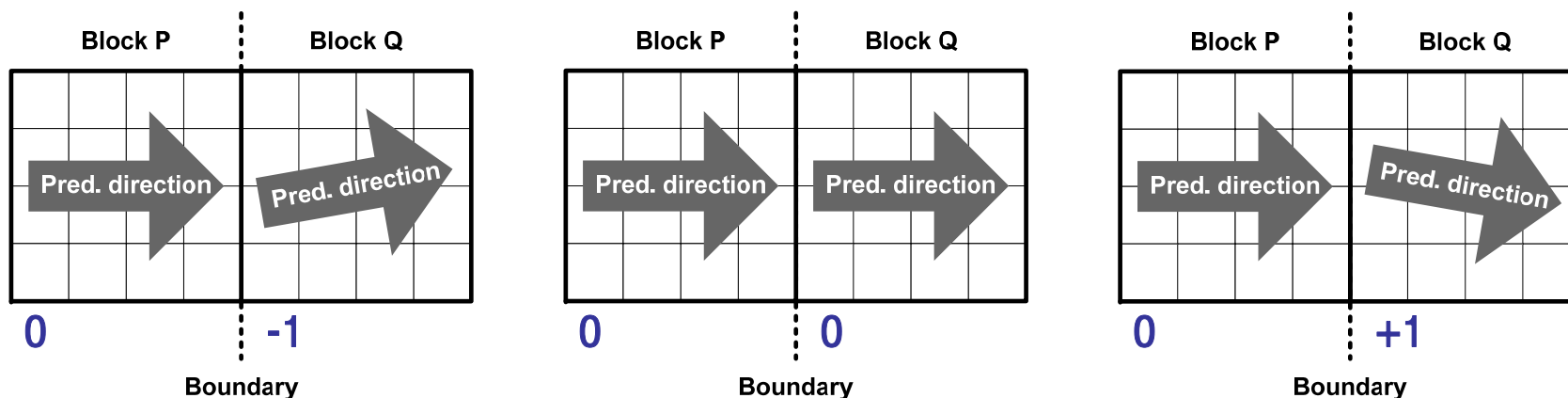
Anchor



Proposed (F258)

Equality of IntraPred Modes

- Equality of IntraPredModes of P and Q



If ($|IntraPredAngleID \text{ of } P - IntraPredAngleID \text{ of } Q| < 2$) same direction;
else different direction

- If only one of P and Q is intra coded, then treat the case as having different intra prediction modes.

Chroma bS decision

- In HM3.0, the chroma bS value is set same as its luma bS value.
- Unified bS decision in luma & chroma: in this proposal, an independent chroma bS decision is performed in the same manner as the luma bS decision

Conditions	bS	t _c offset	wrt HM3.0
P and Q have <i>different</i> intra prediction modes*	4	3	Modified
P and Q have the <i>same</i> intra prediction mode*	3	1	Modified
P and Q are <i>inter</i> coded block, <i>and</i> P and/or Q has non-zero coded coefficients	2	0	Modified
Otherwise	0	n/a	No change

* No need to check bS=1 case since the same motion data(ref. pic. and MV) is shared bet. luma and chroma blocks.

Luma filtering decision

- Luma filtering is performed when the following filtering decision (block-level) conditions are satisfied
 - Note: same way as the HM3.0 design

bS > 0 &

$$d = |p_{2,2} - 2 * p_{1,2} + p_{0,2}| + |q_{2,2} - 2 * q_{1,2} + q_{0,2}| + |p_{2,5} - 2 * p_{1,5} + p_{0,5}| + |q_{2,5} - 2 * q_{1,5} + q_{0,5}| < \beta$$

- The value of d is calculated by a sum of d_{p0} and d_{q0} as follows:
 - The value of d_{p0} and d_{q0} are later used also for luma weak filtering process

$$d_{p0} = |p_{2,2} - 2 * p_{1,2} + p_{0,2}| + |p_{2,5} - 2 * p_{1,5} + p_{0,5}| < \beta_p$$

$$d_{q0} = |q_{2,2} - 2 * q_{1,2} + q_{0,2}| + |q_{2,5} - 2 * q_{1,5} + q_{0,5}| < \beta_q$$

	p3,0	p2,0	p1,0	p0,0	q0,0	q1,0	q2,0	q3,0	
	p3,1	p2,1	p1,1	p0,1	q0,1	q1,1	q2,1	q3,1	
	p3,2	p2,2	p1,2	p0,2	q0,2	q1,2	q2,2	q3,2	
Block	p3,3	p2,3	p1,3	p0,3	q0,3	q1,3	q2,3	q3,3	Block
P	p3,4	p2,4	p1,4	p0,4	q0,4	q1,4	q2,4	q3,4	Q
	p3,5	p2,5	p1,5	p0,5	q0,5	q1,5	q2,5	q3,5	
	p3,6	p2,6	p1,6	p0,6	q0,6	q1,6	q2,6	q3,6	
	p3,7	p2,7	p1,7	p0,7	q0,7	q1,7	q2,7	q3,7	

Block Boundary

Luma filter type decision

- In HM3.0, a strong filter is applied when the three inequalities are satisfied

$$d < (\beta \gg 2)$$

→ Block-level decision

$$|p_3 - p_0| + |q_0 - q_3| < (\beta \gg 2)$$

→ Line-level decision

$$|p_3 - p_0| < (5 \cdot t_c + 1) \gg 1$$

→ Line-level decision

- In this proposal, one additional **block-level inequality condition** is added to the above three inequality conditions as follows:

$$d_{p1} < \beta_s \text{ and } d_{q1} < \beta_s$$

$$\text{where } d_{p1} = |p_{3,2} - 2 \cdot p_{2,2} + p_{1,2}| + |p_{3,5} - 2 \cdot p_{2,5} + p_{1,5}|$$

$$d_{q1} = |q_{3,2} - 2 \cdot q_{2,2} + q_{1,2}| + |q_{3,5} - 2 \cdot q_{2,5} + q_{1,5}|$$

	p _{3,0}	p _{2,0}	p _{1,0}	p _{0,0}	q _{0,0}	q _{1,0}	q _{2,0}	q _{3,0}	
	p _{3,1}	p _{2,1}	p _{1,1}	p _{0,1}	q _{0,1}	q _{1,1}	q _{2,1}	q _{3,1}	
	p _{3,2}	p _{2,2}	p _{1,2}	p _{0,2}	q _{0,2}	q _{1,2}	q _{2,2}	q _{3,2}	
Block	p _{3,3}	p _{2,3}	p _{1,3}	p _{0,3}	q _{0,3}	q _{1,3}	q _{2,3}	q _{3,3}	Block
P	p _{3,4}	p _{2,4}	p _{1,4}	p _{0,4}	q _{0,4}	q _{1,4}	q _{2,4}	q _{3,4}	Q
	p _{3,5}	p _{2,5}	p _{1,5}	p _{0,5}	q _{0,5}	q _{1,5}	q _{2,5}	q _{3,5}	
	p _{3,6}	p _{2,6}	p _{1,6}	p _{0,6}	q _{0,6}	q _{1,6}	q _{2,6}	q _{3,6}	
	p _{3,7}	p _{2,7}	p _{1,7}	p _{0,7}	q _{0,7}	q _{1,7}	q _{2,7}	q _{3,7}	

Block Boundary

- Note: it is checked in block-level only for those blocks decided to be filtered in < luma filtering decision > and very flat ($d < (\beta \gg 2)$) case.

Luma filtering – Strong Filter

- Exactly same as HM3.0, 6 pixels (3 for each side) are filtered as following.

$$\begin{aligned}
 p'_0 &= \text{Clip0-255} ((p_2 + 2 * p_1 + 2 * p_0 + 2 * q_0 + q_1 + 4) \gg 3) \\
 q'_0 &= \text{Clip0-255} ((p_1 + 2 * p_0 + 2 * q_0 + 2 * q_1 + q_2 + 4) \gg 3) \\
 p'_1 &= \text{Clip0-255} ((p_2 + p_1 + p_0 + q_0 + 2) \gg 2) \\
 q'_1 &= \text{Clip0-255} ((p_0 + q_0 + q_1 + q_2 + 2) \gg 2) \\
 p'_2 &= \text{Clip0-255} ((2 * p_3 + 3 * p_2 + p_1 + p_0 + q_0 + 4) \gg 3) \\
 q'_2 &= \text{Clip0-255} ((p_0 + q_0 + q_1 + 3 * q_2 + 2 * q_3 + 4) \gg 3)
 \end{aligned}$$

Luma filtering – Weak Filtering

- Compute delta $\Delta = \text{Clip3}(-t_c, t_c, ((q_0 - p_0) << 2) + (q_1 - p_1) + 4) >> 3)$
 - This is same as H.264/AVC filter which has 4 tap of power-of-two coefficients.
 - **Note:** the same filter is used also for chroma.
 - **Note:** One computation of Δ is used both for 1st (p_0, q_0) & 2nd pixels(p_1, q_1)
- For pixels at the second position from the boundary (p_1 and q_1)
 - Note the condition check is at block-level.

$$p_1' = \begin{cases} \text{Clip0-255}(p_1 + \Delta/2) & \text{if } d_{p_0} < \beta_p \\ p_1 & \text{else} \end{cases}$$

$$q_1' = \begin{cases} \text{Clip0-255}(q_1 - \Delta/2) & \text{if } d_{q_0} < \beta_q \\ q_1 & \text{else} \end{cases}$$

- For pixels at the first position from the boundary
 - Note the condition check is at block-level.

$$\text{if } d_{p_0} < \beta_p \text{ and } d_{q_0} < \beta_q$$

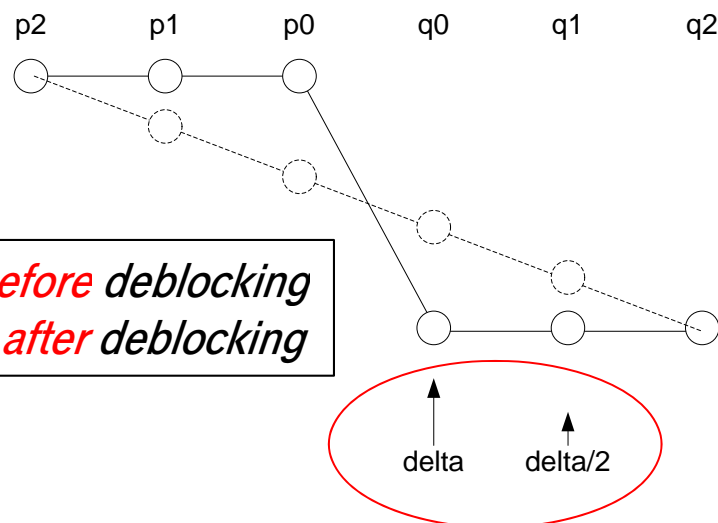
$$p_0' = \text{Clip0-255}(p_0 + \Delta); \quad q_0' = \text{Clip0-255}(q_0 - \Delta)$$

else

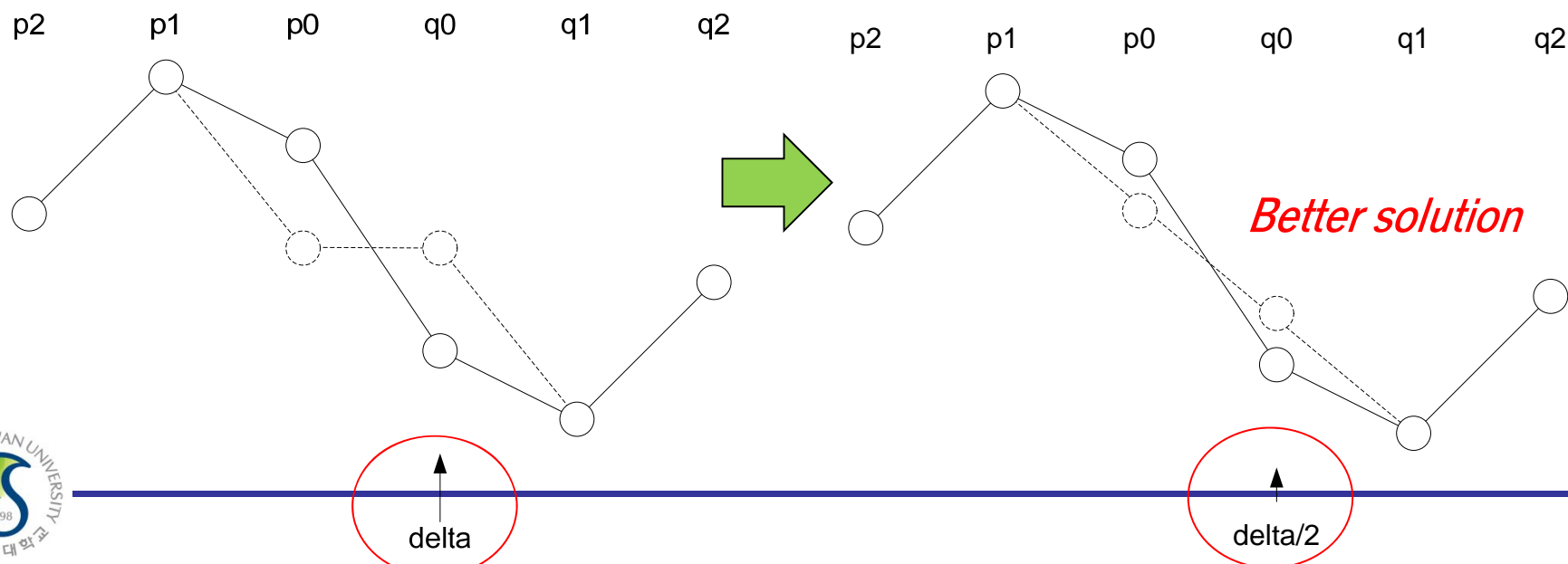
$$p_0' = \text{Clip0-255}(p_0 + \Delta/2); \quad q_0' = \text{Clip0-255}(q_0 - \Delta/2)$$

Illustration: Luma Weak Filtering

- Case 1: (p_0, q_0) is filtered with Δ



- Case 2: (p_0, q_0) is filtered with $\Delta/2$





Simulation Conditions

- Reference SW : HM3.0
- The recommended sequences, configurations and 6 (AIHE, AILC, RAHE, RALC, LDBHE, LDBLC) test cases described in JCTVC-E700 was used.

Simulation Results (1)

	Intra Only HE			Intra Only LC		
	Y	U	V	Y	U	V
Class A	-1.7	-0.5	-0.5	-1.1	-0.6	-0.5
Class B	-1.3	-0.5	-0.5	-1.1	-0.7	-0.6
Class C	-1.0	-0.8	-0.7	-1.0	-0.9	-0.8
Class D	-0.9	-0.7	-0.7	-0.9	-0.7	-0.7
Class E	-1.3	0.0	-0.1	-1.0	0.0	-0.1
Overall	-1.2	-0.5	-0.5	-1.0	-0.6	-0.6
Enc Time[%]	102%			100%		
Dec Time[%]	100%			101%		

Simulation Results (2)

	Random Access HE			Random Access LC		
	Y	U	V	Y	U	V
Class A	-1.5	-0.4	-0.1	-0.9	0.2	0.6
Class B	-1.3	-0.8	-0.8	-1.0	-0.7	-0.5
Class C	-1.0	-0.6	-0.7	-0.8	-0.4	-0.6
Class D	-0.8	-0.8	-0.8	-0.7	-0.4	-0.4
Class E						
Overall	-1.1	-0.7	-0.6	-0.8	-0.3	-0.3
Enc Time[%]	109%			100%		
Dec Time[%]	99%			100%		

Simulation Results (3)

	Intra Only HE			Intra Only LC		
	Y	U	V	Y	U	V
Class A						
Class B	-1.6	-1.2	-1.4	-1.0	-1.1	-0.7
Class C	-1.1	-0.6	-0.8	-0.8	0.4	-0.2
Class D	-1.0	-0.3	-0.3	-0.7	-0.1	-0.2
Class E	-3.0	-2.7	-2.8	-1.6	-4.7	-3.8
Overall	-1.6	-1.1	-1.2	-1.0	-1.1	-1.1
Enc Time[%]	96%			100%		
Dec Time[%]	99%			100%		

Visual Comparison
ParkScene_LDHE_QP35 –55th frame



Anchor



Proposed (F258)

Visual Comparison

BQMall_LDHE_QP37 – 47th frame



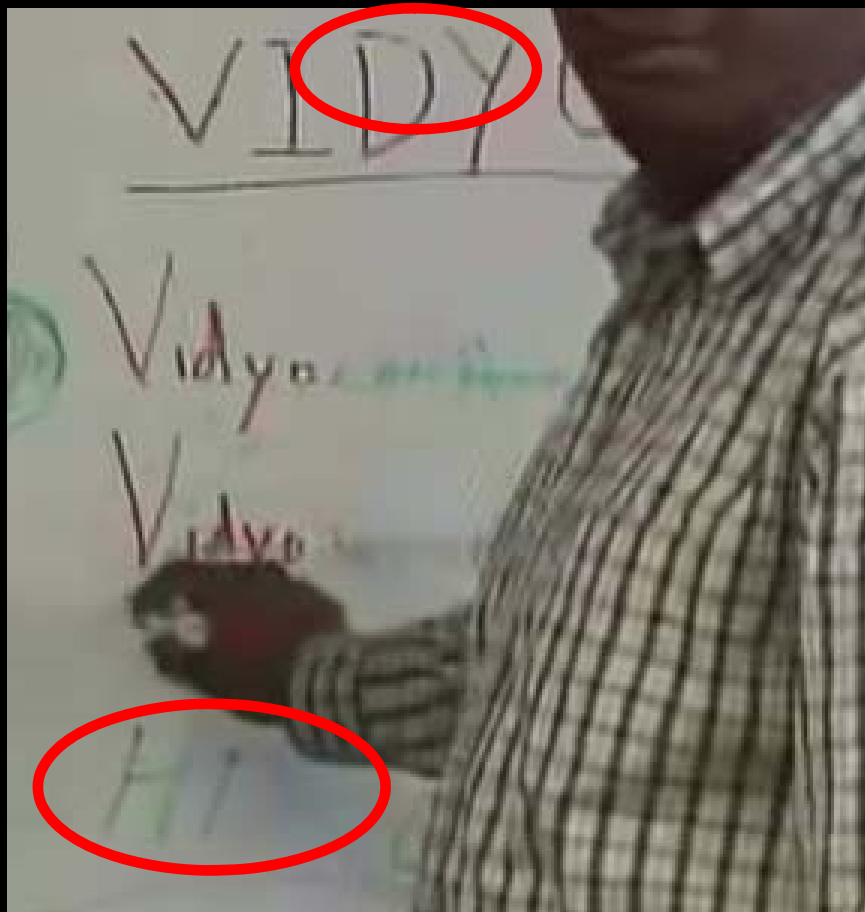
Anchor



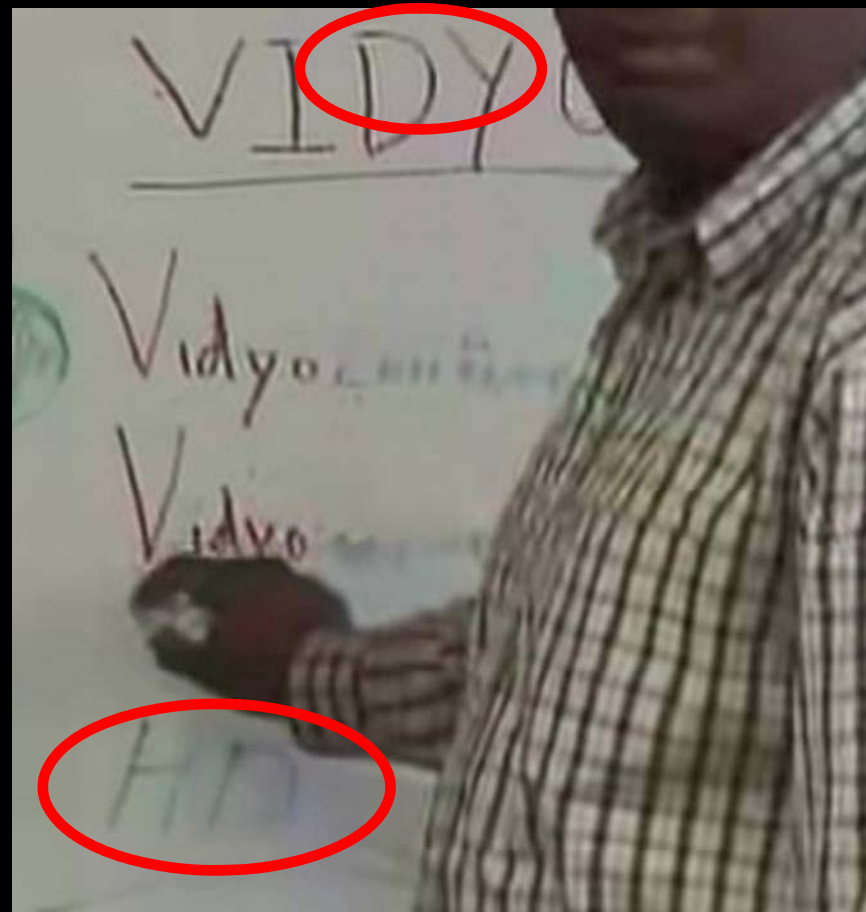
Proposed (F258)

Visual Comparison

Vidyo3_LDHE_QP37 – 205th frame



Anchor

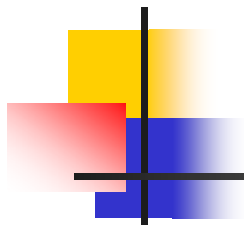


Proposed (F258)



Conclusion Remarks

- This contribution proposed a deblocking filter modified from the one in the HM3.0
- According to the test condition agreed by CE12, it is reported that the proposed filter has BDBR gain of 1.2% (HE_IO), 1.1% (HE_RA), 1.6% (HE_LD) and 1.0% (LC_IO), 0.8% (LC_RA), 1.0% (LC_LD) with approximately similar decoding time compared to HM 3.0
- The proposal shows advantage in BDBR gains with only a few modifications from the current HM3.0.
- Recommended to employ this simple mechanism in coming HM design



Thank you !