



北京大学数字媒体研究所
INSTITUTE OF DIGITAL MEDIA, PEKING UNIVERSITY



J0011: Description of SDR video coding technology proposal

Peking University

DJI

Apr, 2018

Outline

- General Overview
- Brief Description of coding tools
- Hardware design analysis
- Highlight and Conclusion



General Overview

- ❑ Software implementation base: JEM7.0
 - Integration of proposed coding tools only
 - Keep other encoder and decoder design exactly same as JEM7.0
- ❑ Propose four coding tools
 - Tools with substantial changes: Non-local Structure-based Filter (NLSF), Adaptive-update Long-Term Reference (ALTR)
 - Tools with minor changes: OBMC Improvement, Multi-hypothesis probability estimation



General Overview (2/3)

□ General performance compared to HM16.16

	Constraint Set 1					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
Average SDR-A	-36.35%	-42.86%	-42.75%	1.29	1.04	1.11
Average SDR-B	-32.03%	-44.64%	-46.00%	1.25	0.90	1.15
Average all	-34.19%	-43.75%	-44.37%	1.27	0.97	1.13
Enc Time[%]	765%					
Dec Time[%]	2911%					
	Constraint Set 2					
	Over HM16.16					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
Average SDR-B	-26.87%	-42.96%	-44.53%	1.03	0.71	0.94
Enc Time[%]	917%					
Dec Time[%]	3675%					

General Overview (3/3)

□ General performance compared to JEM7.0

	Constraint Set 1					
	Over JEM7.0					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
Average SDR-A	-1.06%	-0.33%	-1.28%	0.03	0.01	0.02
Average SDR-B	-2.08%	-1.09%	-2.15%	0.06	0.02	0.04
Average all	-1.57%	-0.71%	-1.72%	0.04	0.01	0.03
Enc Time[%]	100%					
Dec Time[%]	381%					
	Constraint Set 2					
	Over JEM7.0					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
Average SDR-B	-3.30%	-0.67%	-4.26%	0.10	0.01	0.07
Enc Time[%]	113%					
Dec Time[%]	631%					

Outline

- General Overview
- Brief Description of coding tools
- Hardware design analysis
- Highlight and Conclusion



NLSF: Non-local Structure-based Filter

□ Motivation

■ Current in-loop filter in JEM7.0

- Bilateral Filter (BF)
- Deblocking filter (DF)
- Sample Adaptive Offset (SAO)
- Adaptive Loop Filter (ALF)

■ Local filter

- Focus on local correlation within image patches

□ Proposal

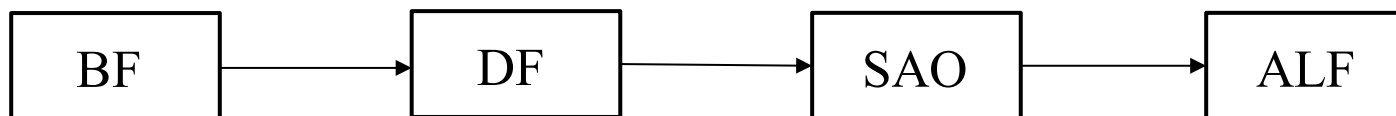
■ Non-local Structure-based Filter

- Based on image non-local self similarity

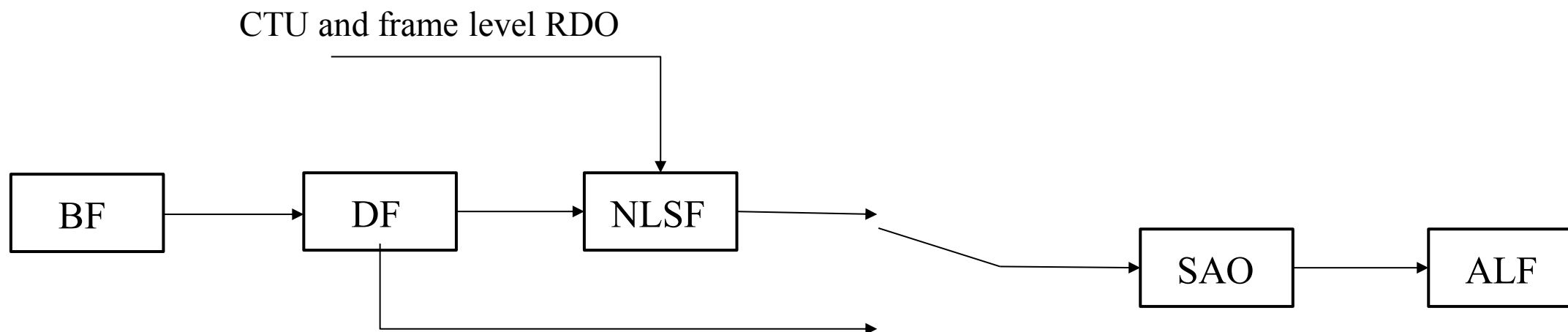


NLSF: Non-local Structure-based Filter(2/5)

❑ In-loop filter in JEM7.0

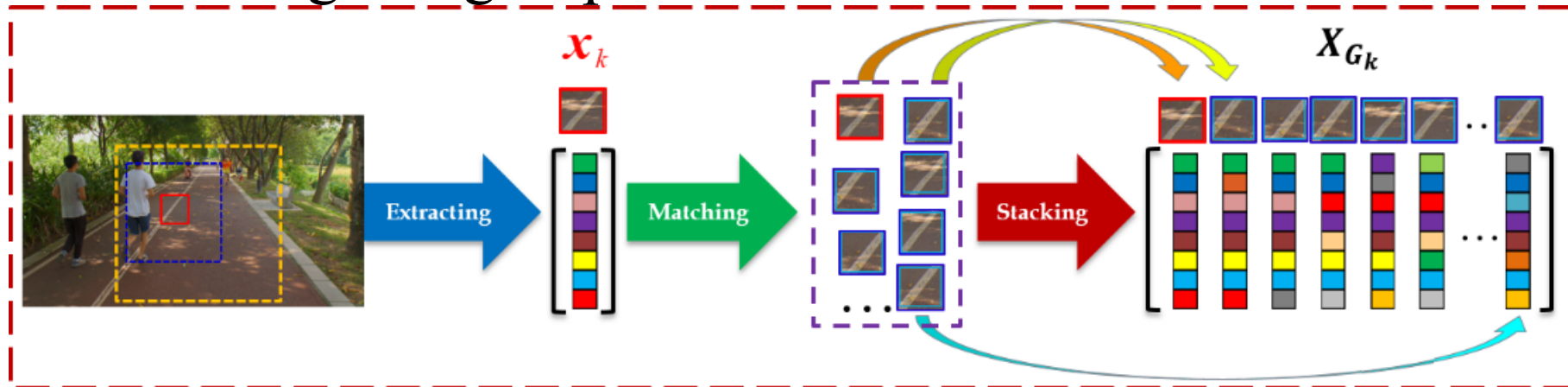


❑ Proposed in-loop filter



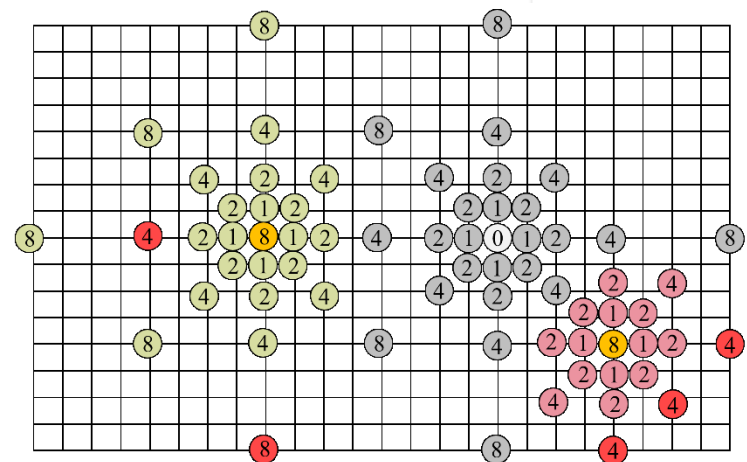
NLSF: Non-local Structure-based Filter(3/5)

□ Block matching and group construction



■ Block matching algorithm

□ Adaptive two-step block matching algorithm

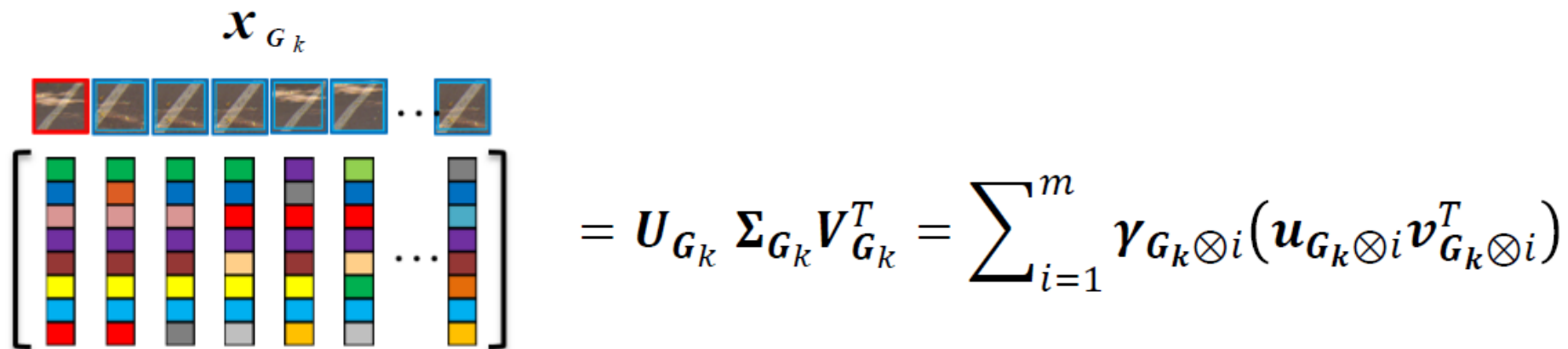


① Reference block ● Searched block in step1 ● Similar block selected in step1
● Searched block in step2 ● Similar block selected in step2

NLSF: Non-local Structure-based Filter(4/5)

□ Group-based filtering

- Decomposed each group $\mathbf{x}_{G_k}$ by SVD


$$\begin{bmatrix} \text{img patches} \\ \vdots \end{bmatrix} = U_{G_k} \Sigma_{G_k} V_{G_k}^T = \sum_{i=1}^m \gamma_{G_k \otimes i} (u_{G_k \otimes i} v_{G_k \otimes i}^T)$$

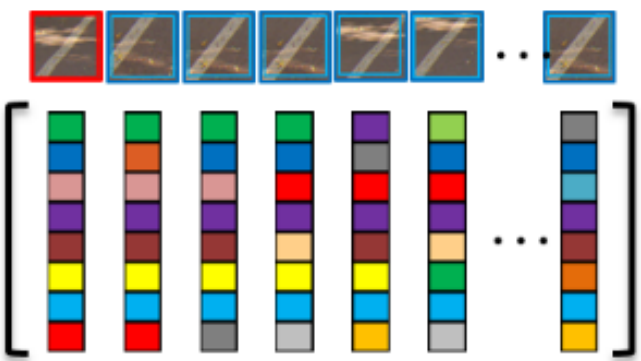
- Apply hard threshold operation to singular value γ_{G_k}

$$\alpha_{G_k} = \text{hard}(\gamma_{G_k}, c\tau)$$

NLSF: Non-local Structure-based Filter(5/5)

□ Image reconstruction

- Reconstruct group $\hat{x}_{G_k}$ from the shrunk singular values α_{G_k}

$$\sum_{i=1}^m \alpha_{G_k \otimes i} (\mathbf{u}_{G_k \otimes i} \mathbf{v}_{G_k \otimes i}^T) = \begin{bmatrix} \text{img}_1 & \text{img}_2 & \text{img}_3 & \text{img}_4 & \text{img}_5 & \text{img}_6 & \dots & \text{img}_n \\ \text{vec}_1 & \text{vec}_2 & \text{vec}_3 & \text{vec}_4 & \text{vec}_5 & \text{vec}_6 & \dots & \text{vec}_n \end{bmatrix} \hat{x}_{G_k}$$


- Reconstruct the whole frame

- The whole frame can be reconstructed by reallocating the blocks in all the reconstructed groups $\hat{x}_{G_k}$ to their corresponding positions in the frame

ALTR : Adaptive-update Long-Term Reference

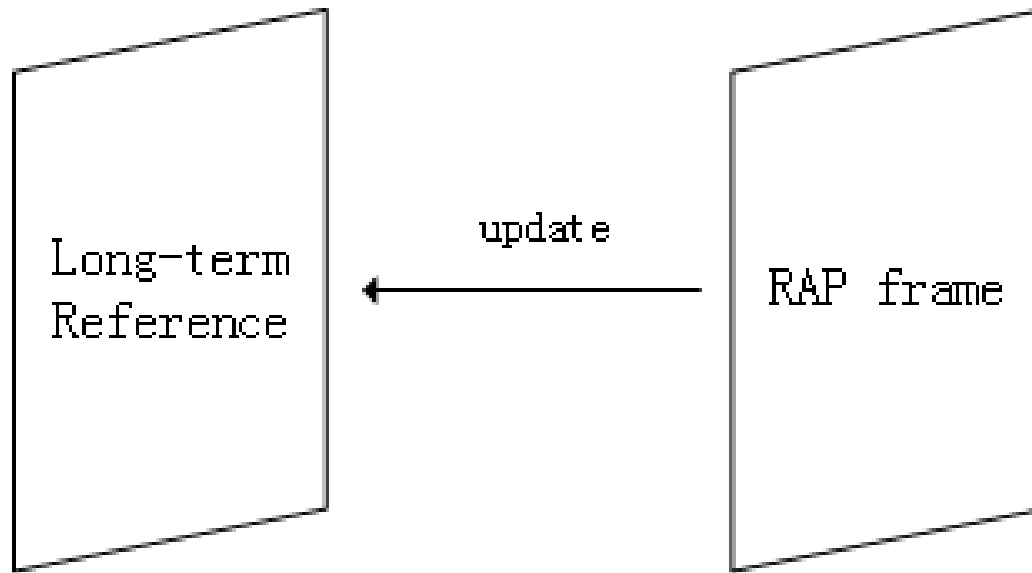
- Long-term reference update mechanism
 - Update partial CTUs from current reconstruction frame
 - Criteria: content similarity between current and co-located CTUs
 - Syntax: CTU-level flag

- Three process flows
 - Initialization, pre-analyze and update

- Only applied for CS2 category at the CfP response

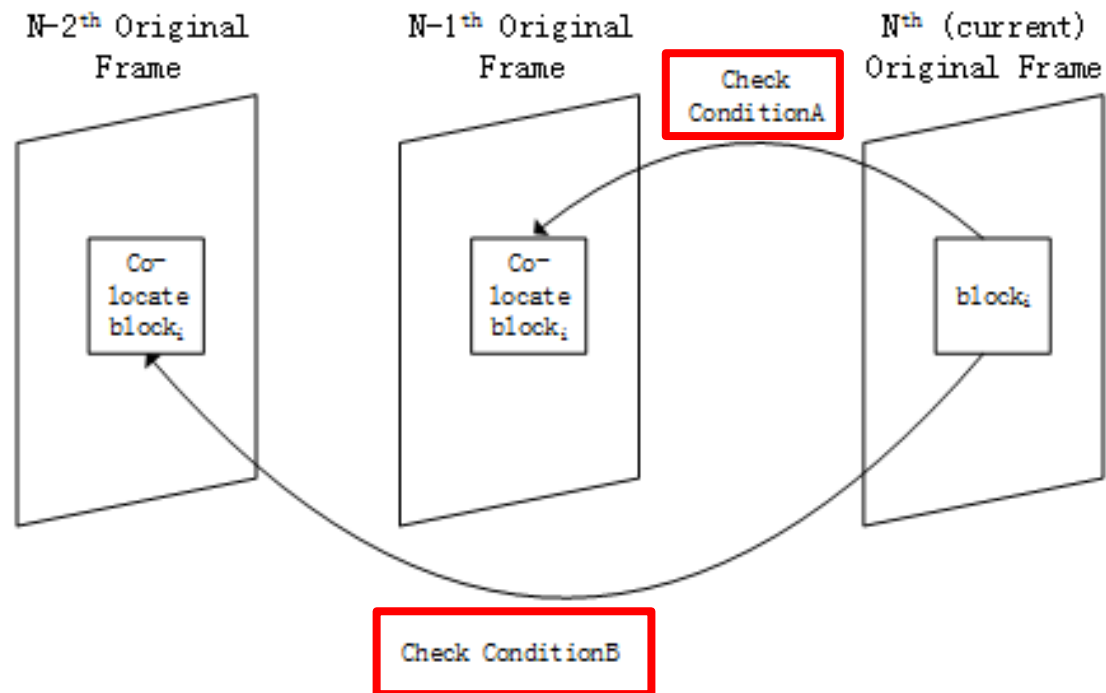


ALTR: Initialization



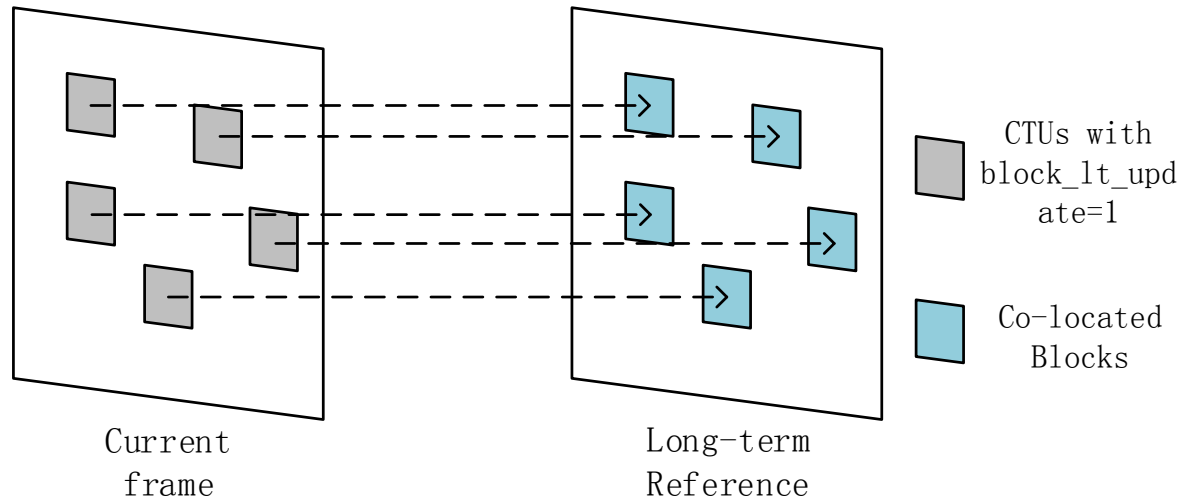
- Initialize long-term reference with RAP frame

ALTR: Pre-analyze



- ❑ Criteria (each CTU)
 - Ratio of similar pixels more than R .
 - Similar pixel: difference between **current** and **co-located** pixel is less than T
- ❑ Apply criteria
 - ConditionA: between frame N and frame $N-1$
 - ConditionB: between frame N and frame $N-1$ & frame $N-2$
- ❑ If either conditionA or conditionB is true, record CTU index

ALTR: Update



- Update CTUs based on recorded indices
- If more than half CTUs are recorded
 - Sort CTUs by ratio R (ascend)
 - Only half number of total CTUs will be updated.

Compatibility

- Related modification of tools in JEM7
 - Disable BIO when BIO uses long-term as reference
 - Disable decoder-side MV refinement (DMVR) at long-term reference



OBMC Improvement: OBMC in JEM

□ For the processed 4x4 block

■ Normal case:

- CU size ≥ 64 : processing the boundary 4 rows/columns
- CU size < 64 : processing the boundary 2 rows/columns

■ Sub-PU case:

- Processing the boundary 2 rows/columns

■ Final Prediction Value

$$P_i = (1 - w_i) \cdot P_{mv_c} + w_i \cdot P_{mv_n}$$

- P_{mv_c} : prediction value by the current MV; P_{mv_n} : prediction value by the neighbouring MV
- w_i : weight factors
 $\{1/4, 1/8, 1/16, 1/32\}$ for the 1st, 2nd, 3rd, 4th rows(columns)



OBMC Improvement (2/3)

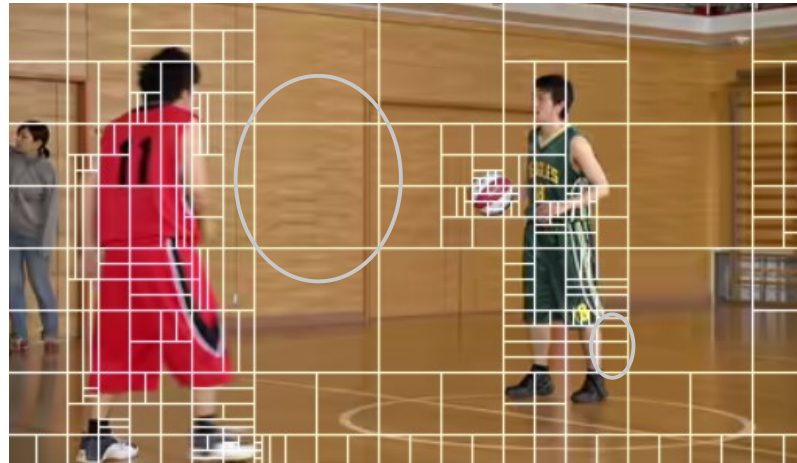
□ Observation & Motivation

■ Large CU

- Current CU and the neighbouring CUs belong to the background area, and the motion field is easily captured.

■ Small CU

- High-texture area, the boundary pixels may belong to the object located in the neighbouring blocks



OBMC Improvement (3/3)

□ Adjusted weight factors according to the CU size

■ CU length

- Width: processing the top/bottom rows
- Height: processing the left/right columns

■ Weight factors of the neighbouring MV

	Small	←	CU length	→	Large	
1 st line:	1/2		1/3	1/4	1/5	1/6
2 nd line:	1/6		1/7	1/8	1/9	1/10
3 rd line:	1/12		1/14	1/16	1/18	1/20
4 th line:	1/24		1/28	1/32	1/36	1/40



CABAC Improvement

- In JEM-7.0, the binary arithmetic coder multi-parameter probability update method:

$$\text{bin} = 1 \rightarrow U[k + 1] = U[k] + \left((2^{15} - U[k]) \gg a \right), \quad V[k + 1] = V[k] + \left((2^{15} - V[k]) \gg b \right),$$

$$\text{bin} = 0 \rightarrow U[k + 1] = U[k] - (U[k] \gg a), \quad V[k + 1] = V[k] - (V[k] \gg b)$$

$$P[k] = (U[k] + V[k]) \gg 1$$

- Where $P[k]$ is the probability estimate, scaled by 2^{15}
- Motivation: better results with
 - Each context has its own statistical properties
 - per-context adaptation speed (proposals G0112, H0061)



CABAC Improvement (2/2)

□ In JEM, the context is initialized at the beginning of each slice

$$\text{bin} = 1 \rightarrow U[k + 1] = U[k] + \left((2^{15} - U[k]) \gg a \right), \quad V[k + 1] = V[k] + \left((2^{15} - V[k]) \gg b \right),$$

$$\text{bin} = 0 \rightarrow U[k + 1] = U[k] - (U[k] \gg a), \quad V[k + 1] = V[k] - (V[k] \gg b)$$

$$P[k] = (U[k] + V[k]) \gg 1$$

- Fixed table with (a_0, b_0) for each context (2 bytes/context)
- Parameters can change only when slice starts
 - Use $(a, b) = (a_0 + 1, b_0 + 1)$ when $c > Thr$, otherwise $(a, b) = (a_0, b_0)$
- The bin counter threshold Thr is set to be 64, 128, 256, 512 (Since the $pStateIdx$ in CABAC of JEM is 512 currently).



Outline

- General Overview
- Brief Description of coding tools
- Hardware design analysis
- Highlight and Conclusion



Brief analysis: Decoder

- ❑ Bandwidth: same as JEM7.0
- ❑ Additional memory usage compared to JEM7.0
 - one long-term reference (ALTR)
 - 16 lines and 16 columns reconstructed pixels outside current CTU (NLSF)
- ❑ Additional computing operations compared to JEM7.0 (NLSF)
 - Singular value calculation with 36x21 matrix: 10 times (maximum) / 4.6 times (average)
 - SSD calculations for each filter unit: 1024 times (maximum) / 260 times (average)



Brief analysis: Encoder

- Additional bandwidth compared to JEM7.0
 - Two original references read (for ALTR pre-analysis process)
- Additional memory usage compared to JEM7.0
 - One long-term reference (ALTR)
 - Two original frames buffer (N-1, N-2 frame) (ALTR)
 - 16 lines and 16 columns reconstructed pixels outside current CTU (NLSF)
- Additional computing operations compared to JEM7.0 (NLSF)
 - Singular value calculation with 36x21 matrix: 10 times (maximum) / 4.6 times (average)
 - SSD calculations for each filter unit: 1024 times (maximum) / 260 times (average)



Outline

- General Overview
- Brief Description of coding tools
- Hardware design analysis
- Highlight and Conclusion



Highlight

- The response is based on JEM7.0 and integrates four coding tools
- The integrated tools show good objective and subjective quality improvement with low encoder time increase compared to JEM7.0
 - Keep almost same encoding time as JEM7.0 for CS1
 - 13% encoding time increase for CS2
- The decoding time increase mainly comes from NLSF and can be further optimized
 - Speed up block matching process
 - Simplify SVD decomposition (JVET-J0071)



Conclusion

- The proposed four tools show substantial quality improvement and do not have conflicts with other state-of-the-art tools.
- The proposed coding tools can be implemented to other platforms feasibly and robustly.
- Suggest considering the proposed tools as candidates for the future video coding standard.



Thanks!

