

PEM (Panasonic Exploration Model)

SDR video coding technology proposal by Panasonic

The Panasonic Video Coding Team

Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - Detailed explanation of major PEM coding tools
 - Some more results for information
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

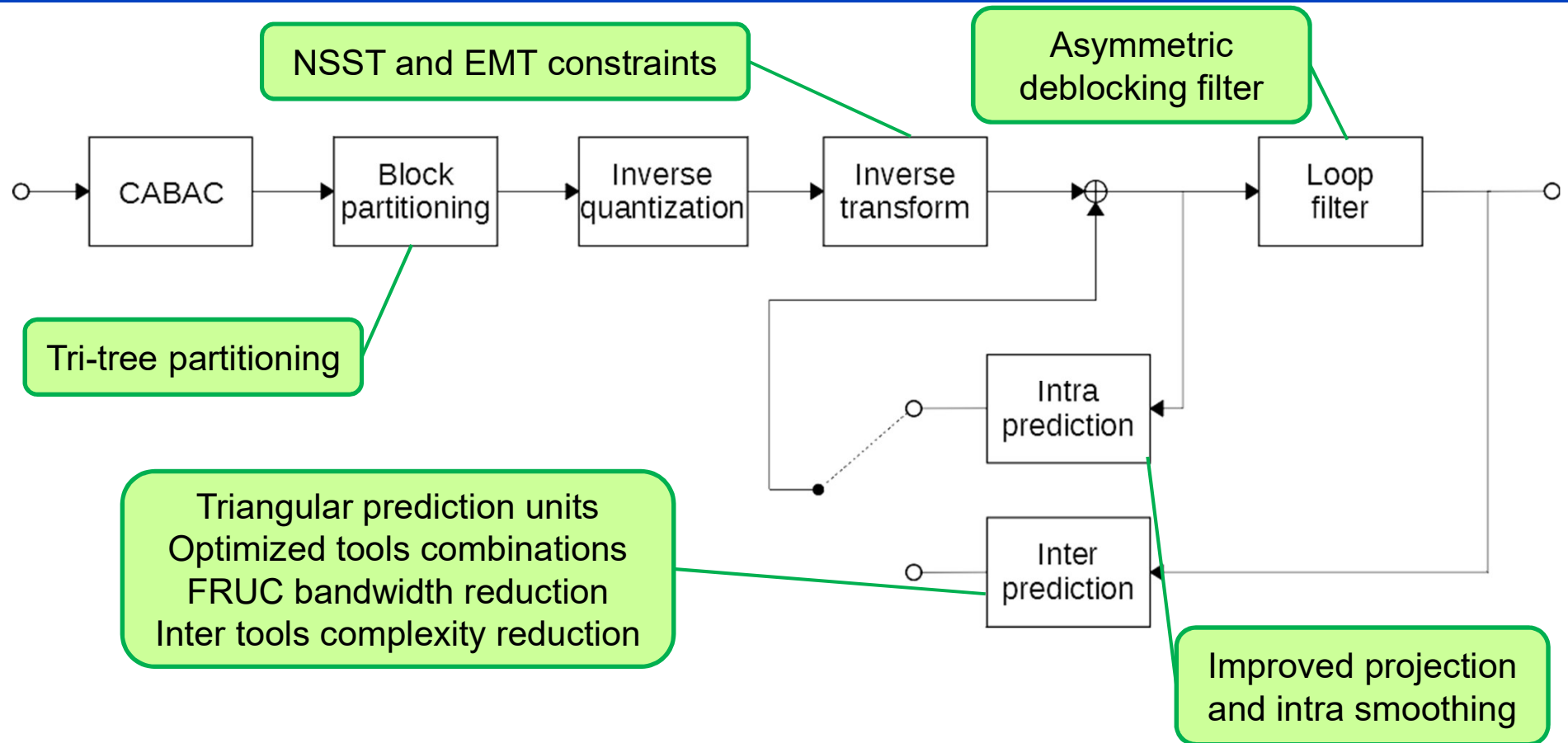
What is PEM?

- Based on JEM7.0
- Increase in coding efficiency
 - 2% coding gain over JEM7.0 with similar encoding complexity
 - 34.6% gain over HM (constraint set 1)
- Significant decoder complexity reduction
 - 56% of runtime of JEM7.0
 - Reduced memory bandwidth
 - Increased capability for parallel processing

Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - Detailed explanation of major PEM coding tools
 - Some more results for information
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

What are the key features in PEM?



Outline

- What is PEM?
 - Overview of PEM coding tools
 - **Results**
 - Detailed explanation of major PEM coding tools
 - Some more results for information
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

Results for constraint set 1 compared to HM16.6

	Constraint Set 1					
	Over HM16.16(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-34.66%	-44.83%	-48.56%	1.74	1.31	1.55
CatRobot1	-40.71%	-54.01%	-47.38%	1.60	0.99	1.35
DaylightRoad2	-42.04%	-58.59%	-42.85%	1.18	1.27	0.80
ParkRunning3	-33.13%	-27.22%	-30.25%	1.07	0.47	0.43
Campfire	-35.82%	-36.19%	-51.69%	1.05	1.37	1.61
BQTerrace	-31.69%	-47.18%	-56.49%	0.95	0.87	1.08
RitualDance	-29.47%	-39.14%	-42.97%	1.59	1.13	1.42
MarketPlace	-30.48%	-48.71%	-46.93%	1.08	0.87	0.87
BasketballDrive	-32.99%	-47.77%	-45.42%	1.19	1.07	1.39
Cactus	-35.29%	-46.91%	-42.70%	1.45	0.76	1.12
Average SDR-A	-37.27%	-44.17%	-44.15%	1.33	1.08	1.15
Average SDR-B	-31.99%	-45.94%	-46.90%	1.25	0.94	1.18
Average all	-34.63%	-45.06%	-45.52%	1.29	1.01	1.16
Enc Time[%]	817%					
Dec Time[%]	384%					

Results for constraint set 1 compared to JEM7.0

	Constraint Set 1					
	Over JEM7.0(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-2.08%	-2.18%	-3.32%	0.08	0.04	0.07
CatRobot1	-2.40%	-4.43%	-3.49%	0.06	0.05	0.07
DaylightRoad2	-2.93%	-4.16%	-3.19%	0.05	0.05	0.04
ParkRunning3	-2.97%	-2.86%	-4.05%	0.08	0.04	0.05
Campfire	-2.34%	-2.38%	-4.39%	0.05	0.08	0.09
BQTerrace	-1.94%	-3.98%	-5.12%	0.05	0.05	0.05
RitualDance	-3.01%	-3.76%	-4.25%	0.14	0.08	0.10
MarketPlace	-1.94%	-2.93%	-4.39%	0.06	0.04	0.06
BasketballDrive	-2.72%	-4.42%	-3.64%	0.08	0.06	0.08
Cactus	-0.44%	-3.27%	-2.95%	0.01	0.04	0.05
Average SDR-A	-2.54%	-3.20%	-3.69%	0.07	0.05	0.06
Average SDR-B	-2.01%	-3.67%	-4.07%	0.07	0.05	0.07
Average all	-2.28%	-3.44%	-3.88%	0.07	0.05	0.07
Enc Time[%]	107%					
Dec Time[%]	56%					

Results for constraint set 1 compared to JEM7.0

	Constraint Set 1					
	Over JEM7.0(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-2.08%	-2.18%	-3.32%	0.08	0.04	0.07
CatRobot1	-2.40%	-4.43%	-3.49%	0.06	0.05	0.07
DaylightRoad2	-2.93%	-4.16%	-3.19%	0.05	0.05	0.04
ParkRunning3	-2.97%	-2.86%	-4.05%	0.08	0.04	0.05
Campfire	-2.34%	-2.38%	-4.39%	0.05	0.08	0.09
BQTerrace	-1.94%	-3.98%	-5.12%	0.05	0.05	0.05
RitualDance	-3.01%	-3.76%	-4.25%	0.14	0.08	0.10
MarketPlace	-1.94%	-2.93%	-4.39%	0.06	0.04	0.06
BasketballDrive	-2.78%	-4.12%	-3.64%	0.08	0.06	0.08
Cactus	-2.54%	-2.27%	-2.95%	0.01	0.04	0.05
Average SDR	-2.01%	-3.20%	-3.69%	0.07	0.05	0.06
Average SDR	-2.01%	-3.57%	-4.07%	0.07	0.05	0.07
Average all	-2.01%	-3.44%	-3.88%	0.07	0.05	0.07
Enc Time[%]	-2.28%			107%		
Dec Time[%]				56%		

Results for constraint set 2 compared to HM16.6

	Constraint Set 2					
	Over HM16.16(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
BQTerrace	-27.39%	-47.34%	-54.06%	0.82	0.56	0.65
RitualDance	-23.69%	-35.04%	-36.57%	1.23	0.85	1.02
MarketPlace	-22.62%	-42.91%	-41.64%	0.73	0.56	0.60
BasketballDrive	-27.41%	-47.70%	-45.64%	1.04	1.04	1.36
Cactus	-28.98%	-42.46%	-38.80%	1.16	0.61	0.86
Average SDR-B	-26.02%	-43.09%	-43.34%	0.99	0.72	0.90
Enc Time[%]	830%					
Dec Time[%]	346%					

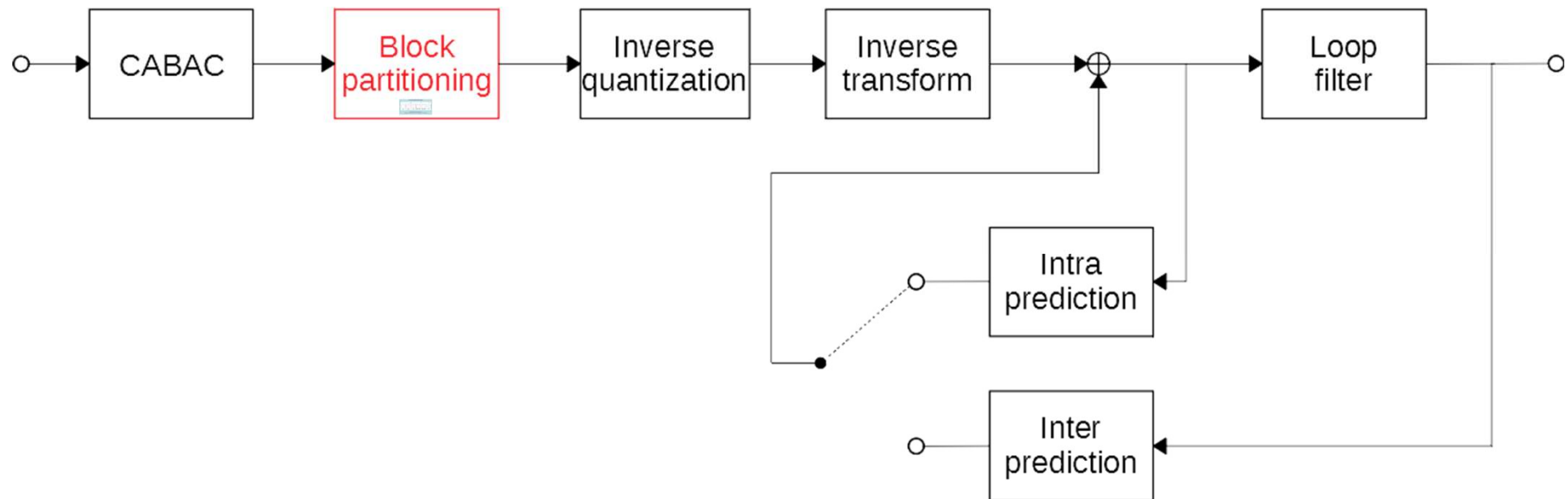
Results for constraint set 2 compared to JEM7.0

	Constraint Set 2					
	Over JEM7.0(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
BQTerrace	-1.63%	-0.19%	-6.94%	0.04	0.00	0.04
RitualDance	-2.57%	-1.12%	-1.77%	0.12	0.02	0.04
MarketPlace	-2.47%	-0.23%	-1.17%	0.07	0.00	0.02
BasketballDrive	-2.45%	-3.57%	-3.17%	0.08	0.05	0.07
Cactus	-1.00%	0.02%	1.23%	0.03	0.00	-0.02
Average SDR-B	-2.02%	-1.02%	-2.36%	0.07	0.02	0.03
Enc Time[%]	100%					
Dec Time[%]	61%					

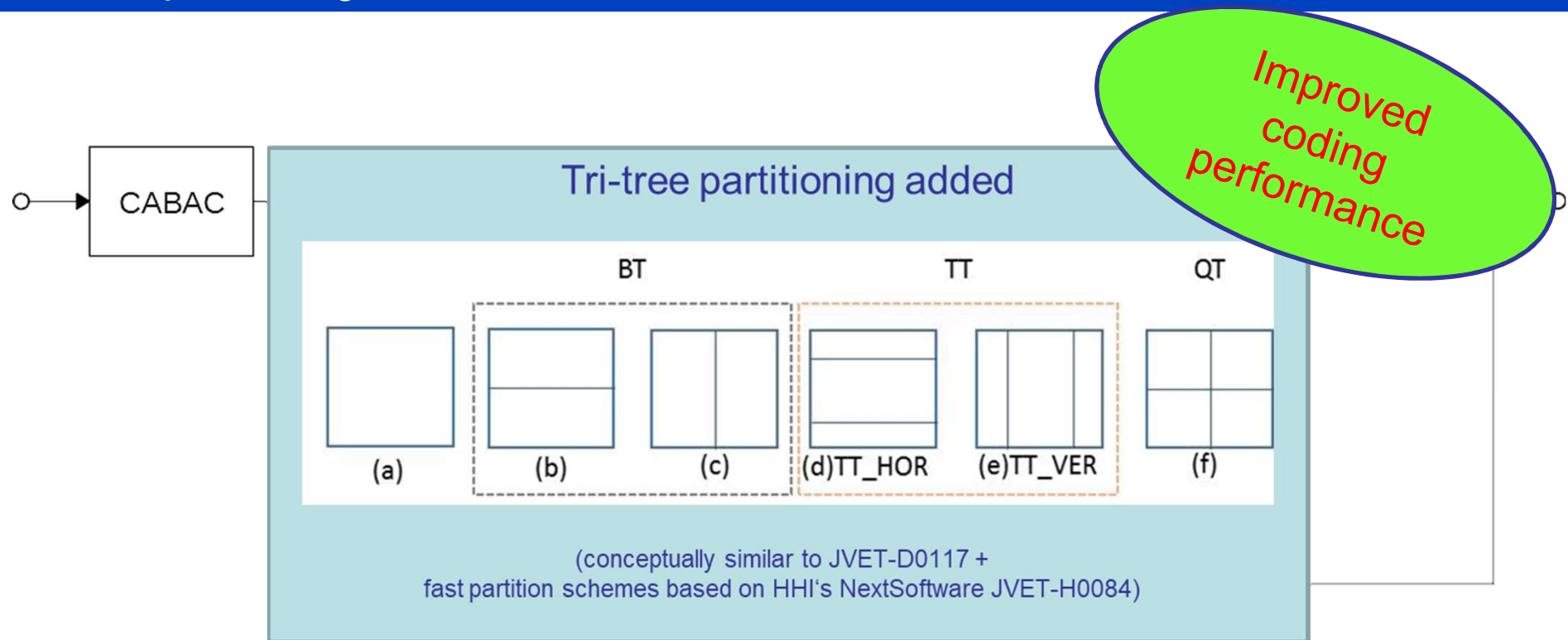
Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - **Detailed explanation of major PEM coding tools**
 - Some more results for information
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

Block partitioning



Block partitioning

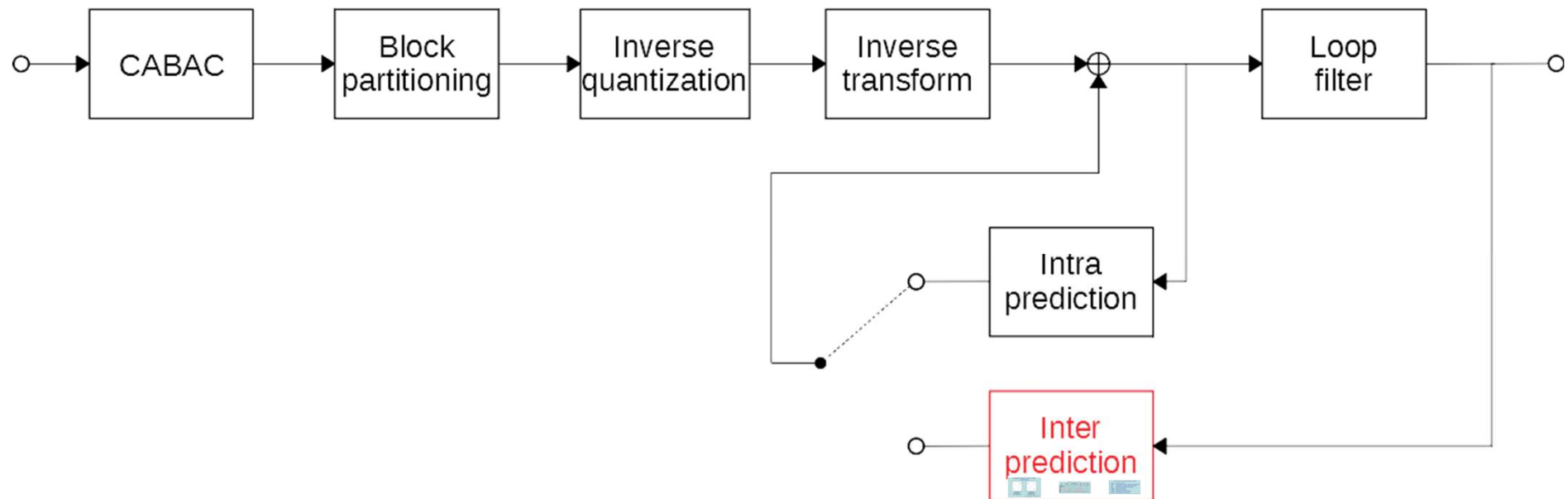


Block partitioning: selection process at the encoder

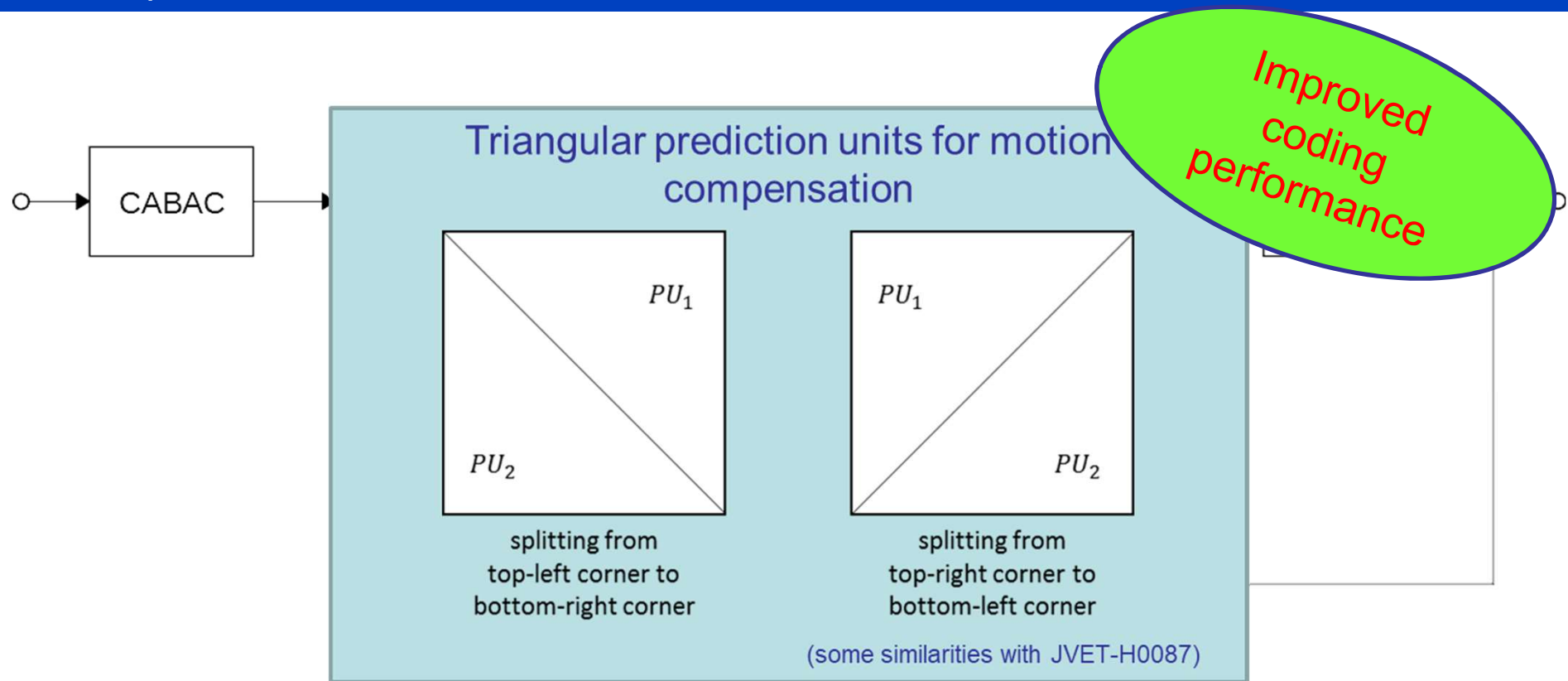
- PEM includes technologies to speed-up the block partitioning selection at the encoder

Condition	Tool-off test					Tool-on test				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
Speed-up by block shape	-0.04%	0.12%	0.28%	106%	100%	0.04%	-0.20%	-0.16%	51%	100%
Speed-up by QT selection result	0.00%	-0.08%	0.27%	109%	100%	0.09%	0.12%	-0.27%	98%	102%
Speed-up by BT selection result	-1.32%	-2.02%	-1.82%	214%	102%	1.24%	2.15%	1.92%	37%	100%
Speed-up by BT/TT selection result	-0.05%	0.02%	0.00%	122%	101%	0.01%	-0.35%	-0.34%	56%	100%
All tools	-1.46%	-2.00%	-1.63%	526%	102%	1.49%	2.05%	1.66%	19%	100%

Inter prediction

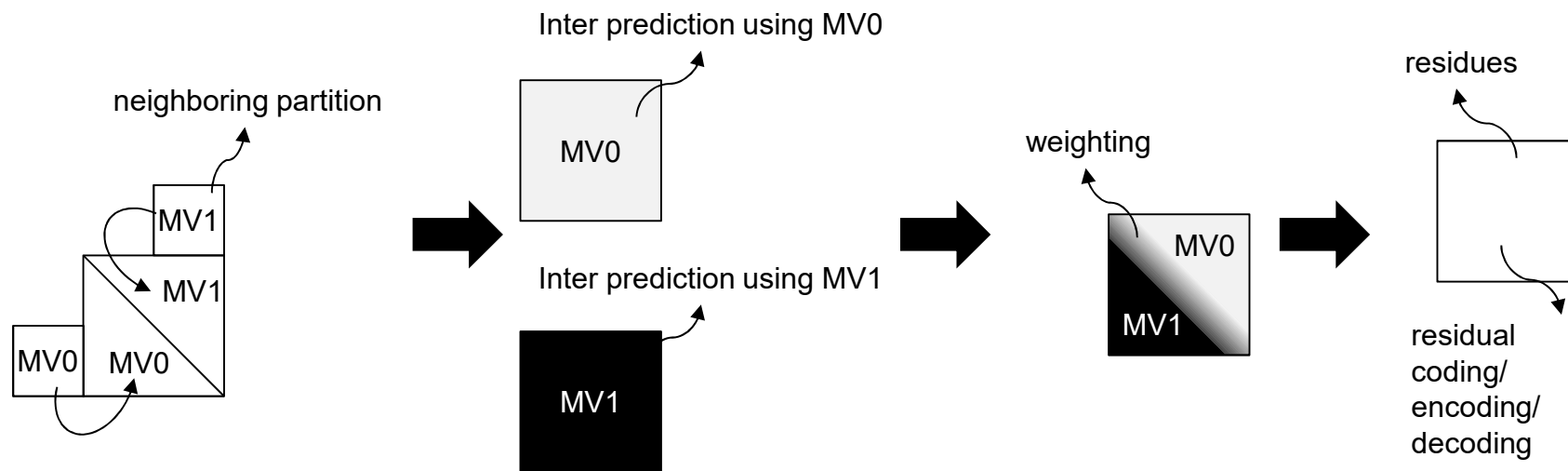
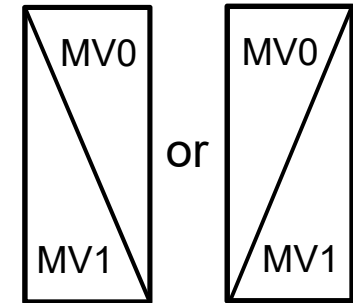


Inter prediction



Triangular prediction units: basic concept

- Splitting a CU into two triangles for prediction
- Only applied to skip and merge CU
- Each triangle has its own motion vector selected from a common MV candidates list



Triangular prediction units: syntax

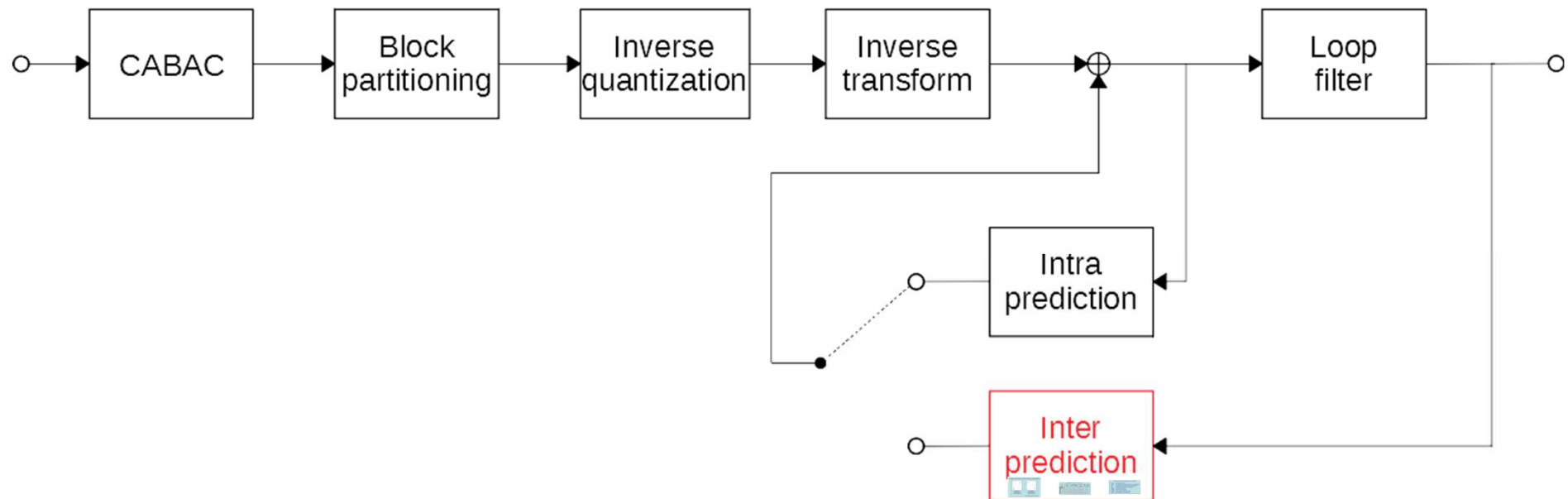
- If skip or merge CU: **triangle_merge_flag**
- If triangle_merge_flag = 1: **triangle_idx**



62 values that indicate both splitting direction and selected motion vectors

Triangle index (EG-1)	Splitting direction	Candidate number in triangle MV candidate list	
		1 st triangle	2 nd triangle
0	top-left to bottom-right	1	0
1	top-right to bottom-left	0	1
2	top-left to bottom-right	1	2
3	top-right to bottom-left	1	0
4	top-right to bottom-left	0	2
5	top-right to bottom-left	1	2
6	top-left to bottom-right	2	0
⋮			

Inter prediction



Inter prediction

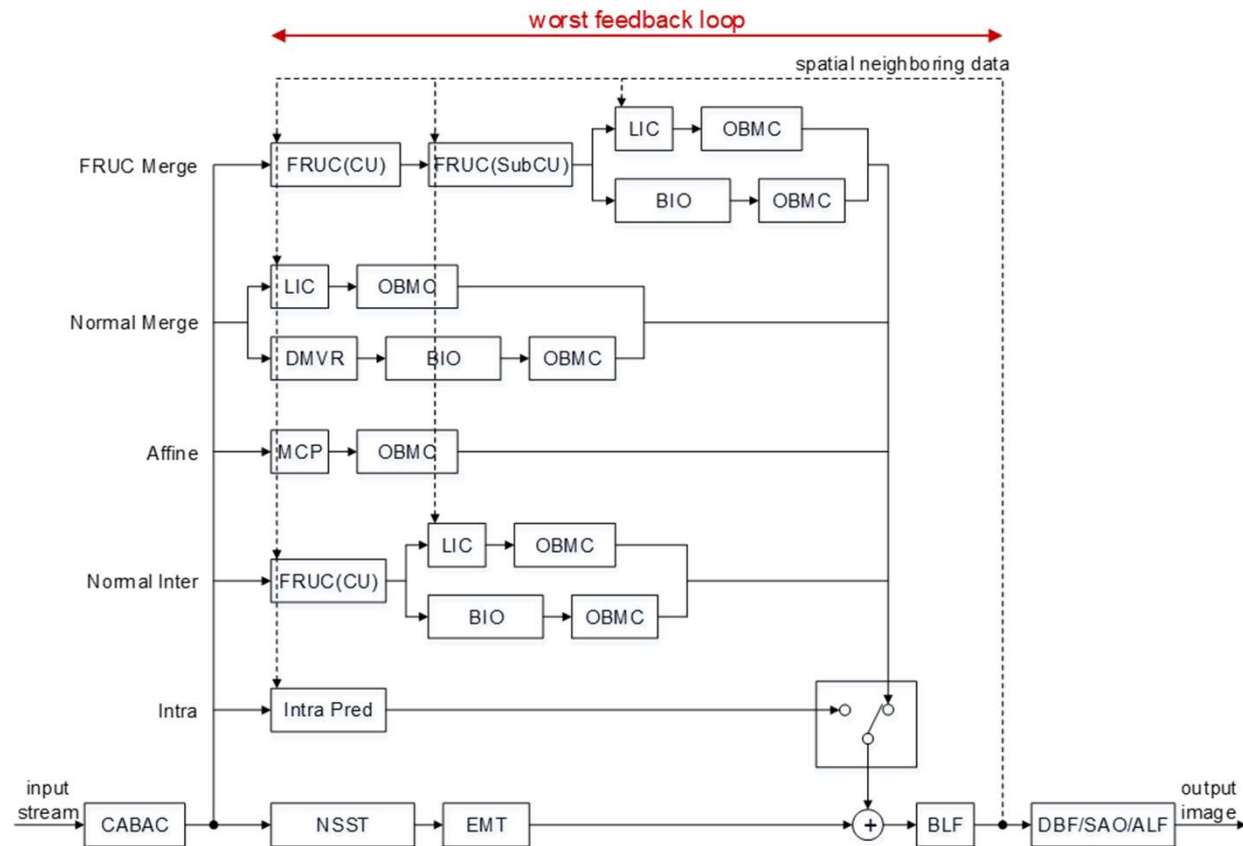
Lower
algorithmic
complexity

Increased capability
for parallel
processing

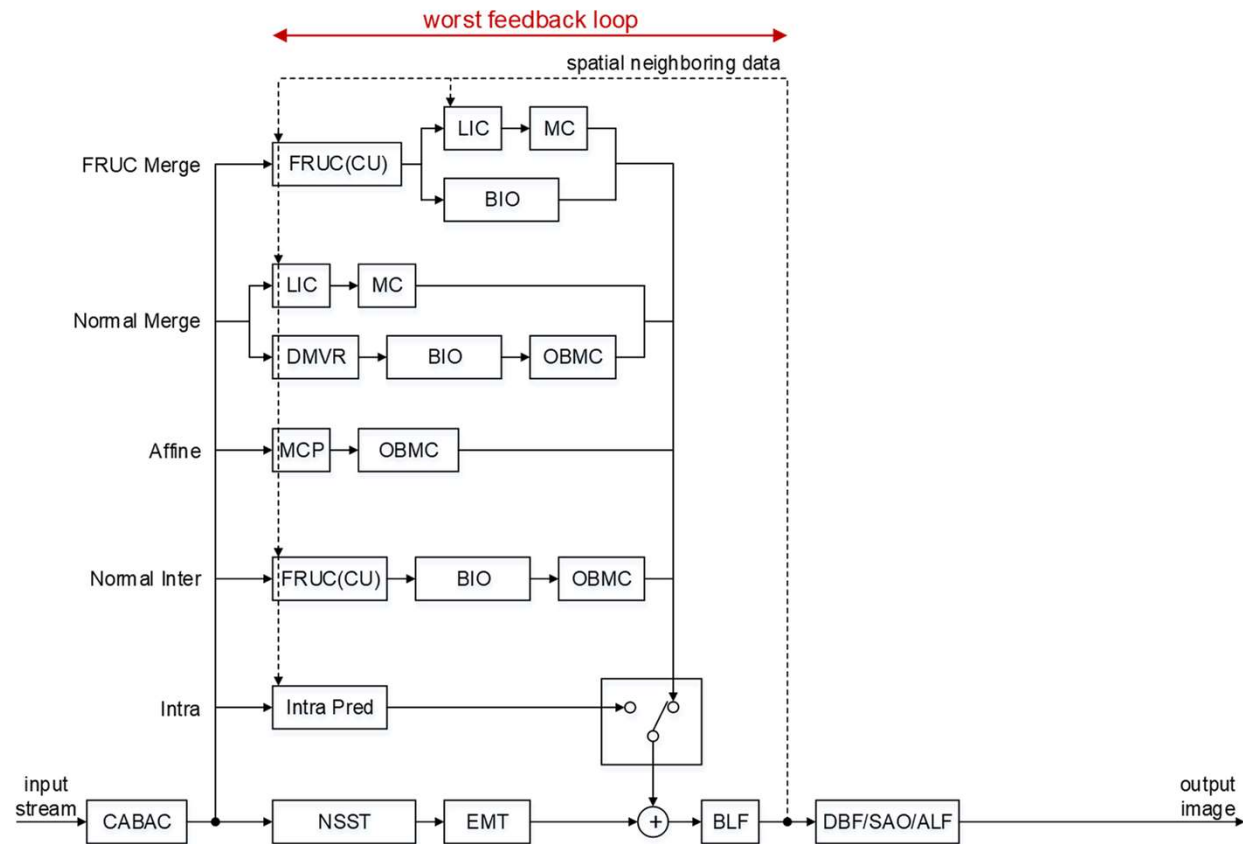
Optimized combinations of inter prediction tools

Tools	Normal Inter		Normal Merge				FRUC Merge		Affine Inter		Affine Merge
			CU		SubCU						
FRUC(CU)	Y						Y				
FRUC(SubCU)							N				
AMVR	Y										
LIC	N		0	1	0	1	0	1			
DMVR			Y								
BIO	Y		Y		N		Y				
OBMC	0	1	Y	N	Y	N	N	N	0	1	Y
EMT	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

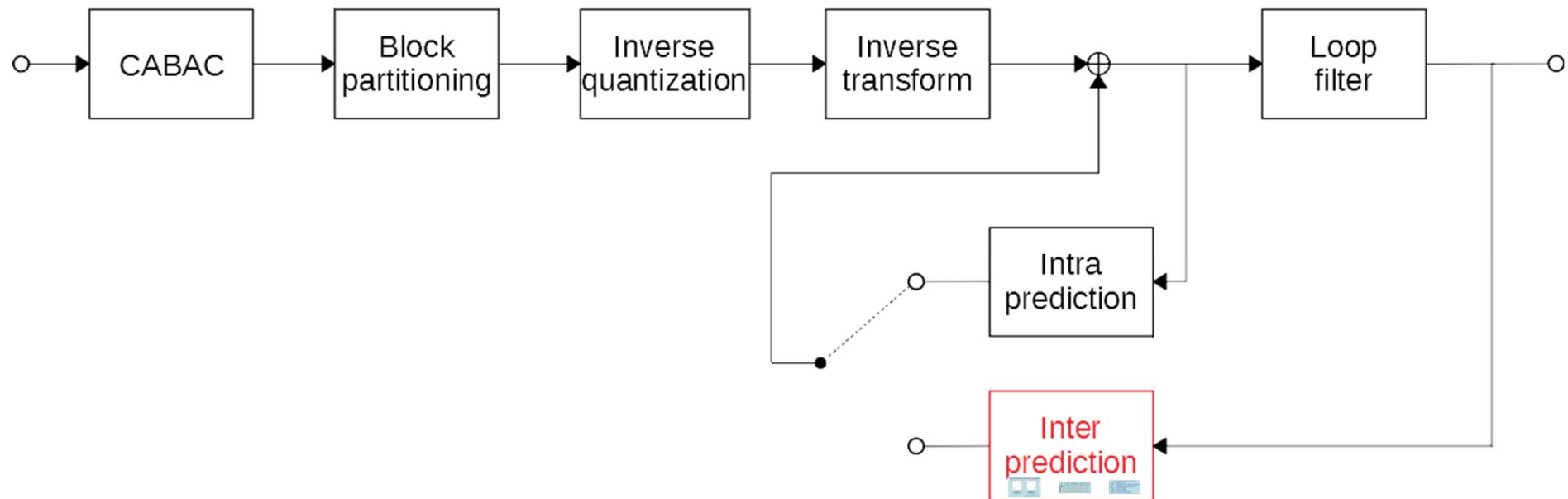
Optimized combinations: process flow of JEM7.0



Optimized combinations: process flow of PEM

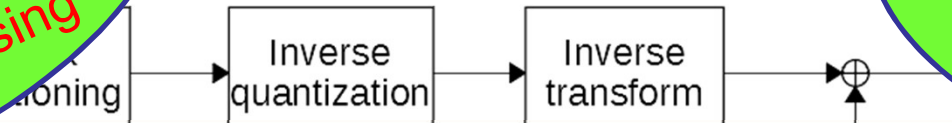


Inter prediction



Inter prediction

Increased
capability for
parallel processing



Lower
algorithmic
complexity

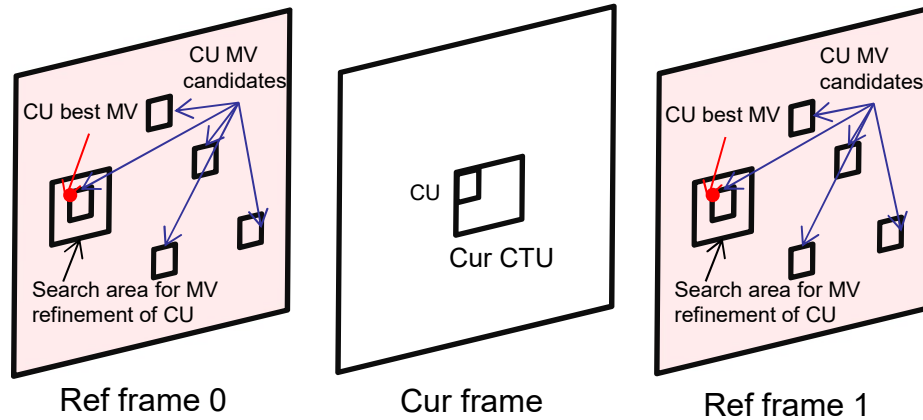
Modifications to inter prediction tools from JEM7.0

- FRUC : bandwidth reduction
- FRUC : MV candidate list and MV refinement simplifications
- BIO : interpolation and gradients filter simplification
- BIO : modifications to optical flow equations
- OBMC : blending simplification
- ATMVP : sub-CU's size restriction
- STMVP : modifications to TMVP derivation
- AMVP : MV candidate list simplification

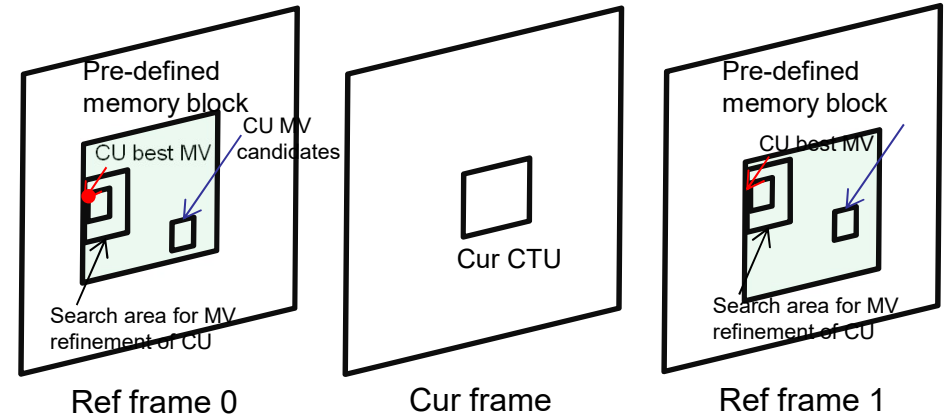
Reduced
memory
bandwidth

FRUC bandwidth reduction: basic concept

JEM7.0



PEM



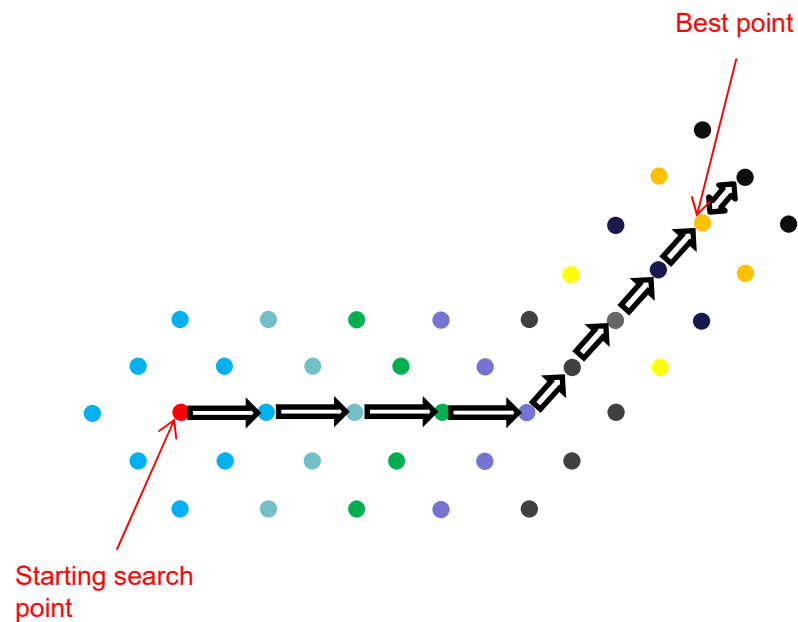
All CUs in a CTU share a pre-defined memory block

FRUC bandwidth per CTU
= $5.6 \times$ bandwidth of motion compensation

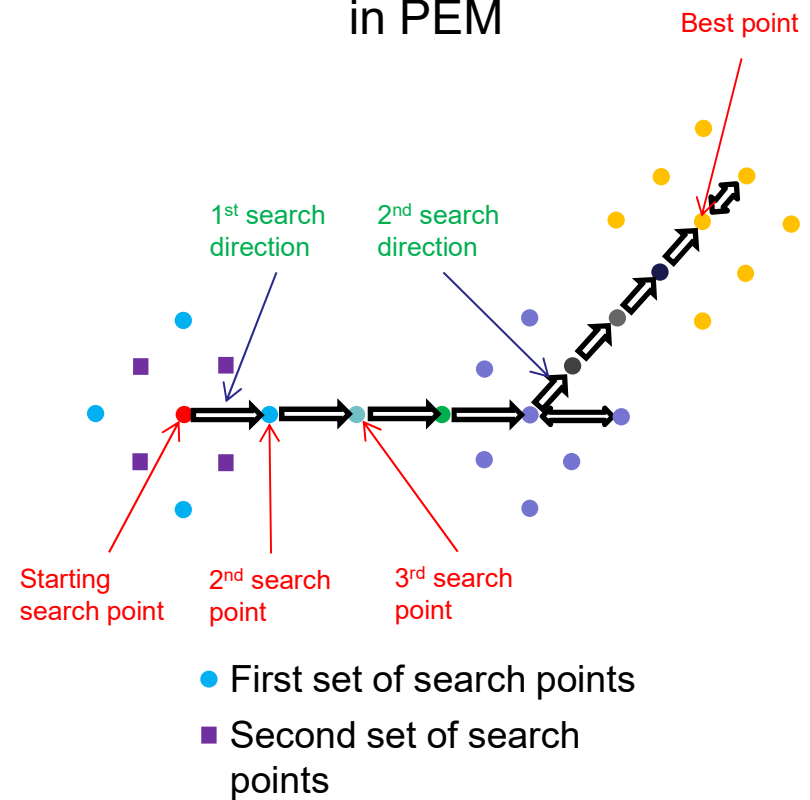
FRUC bandwidth per CTU
= $0.4 \times$ bandwidth of motion compensation

FRUC complexity reduction: simplified diamond search for MV refinement

Diamond search in JEM7.0



Simplified diamond search in PEM

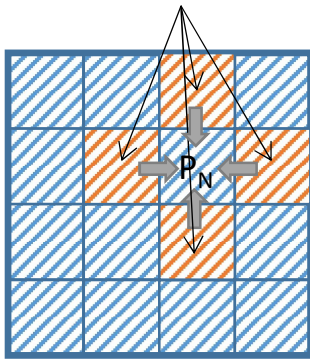


OBMC complexity reduction: simplified blending

CU coded with sub-CU mode
(includes Affine/ATMVP/STMVP)

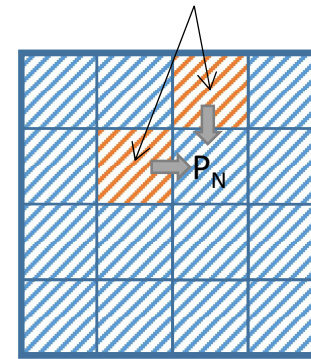
JEM7.0

MVs of four
neighboring sub-blocks
are used in OBMC of P_N

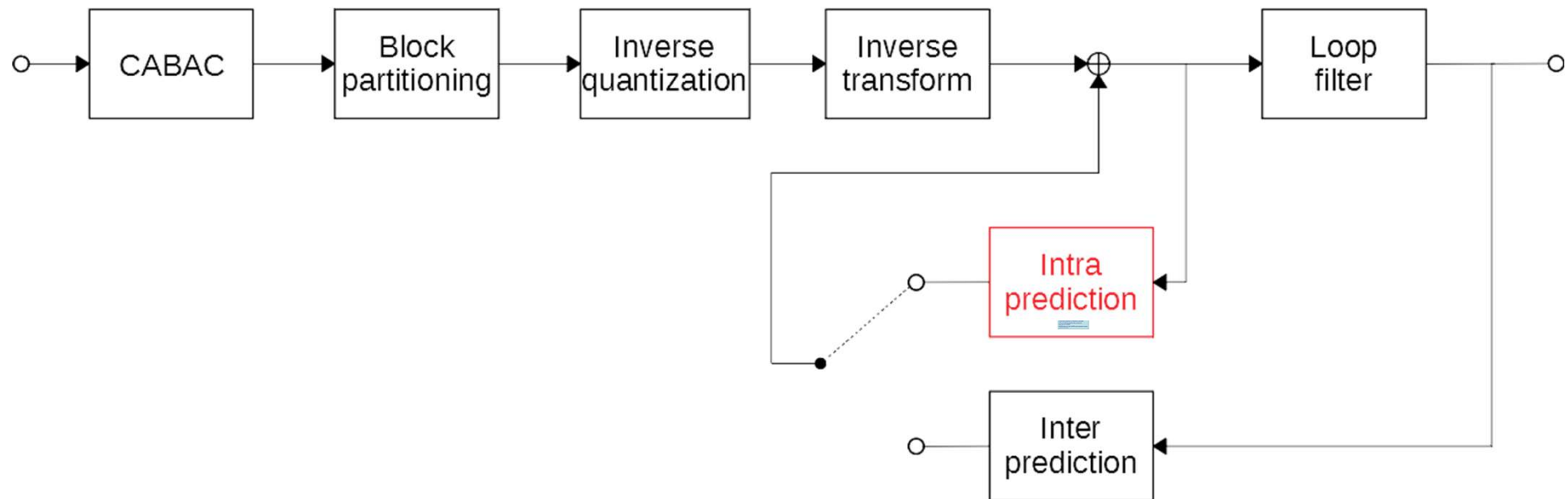


PEM

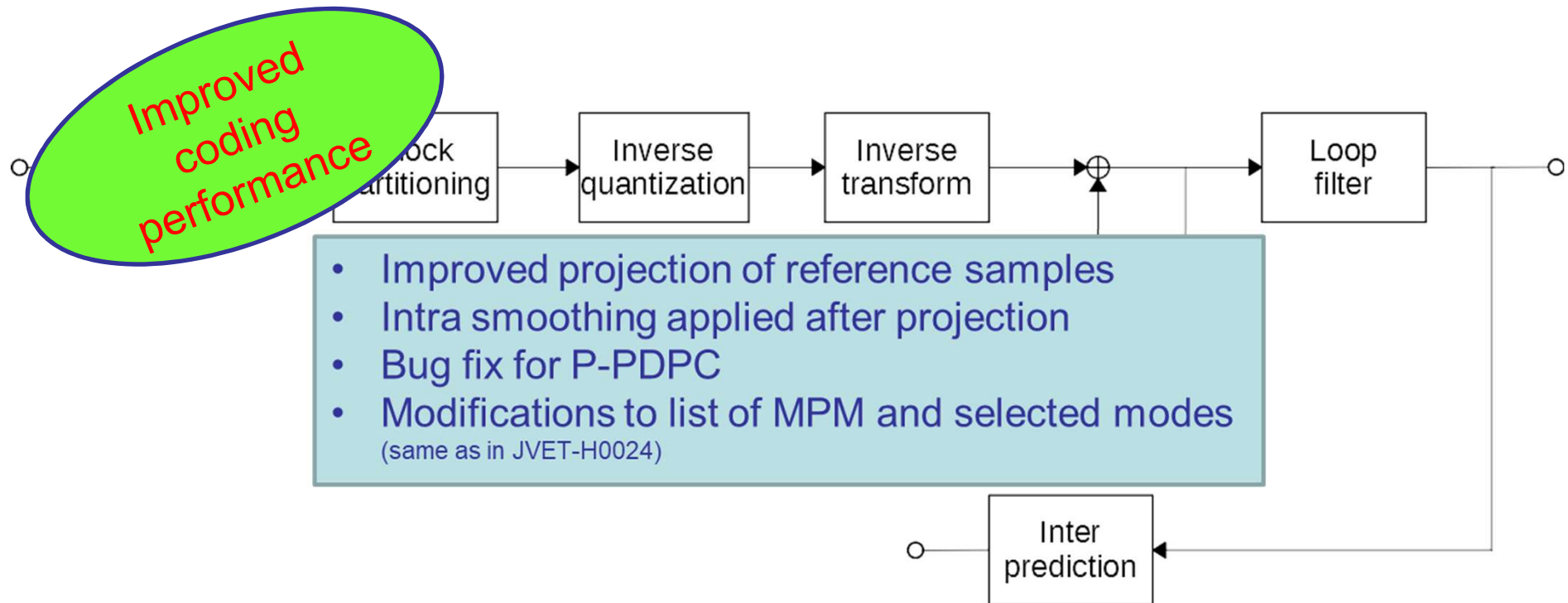
MVs of left and above
neighboring sub-blocks
are used in OBMC of P_N



Intra prediction

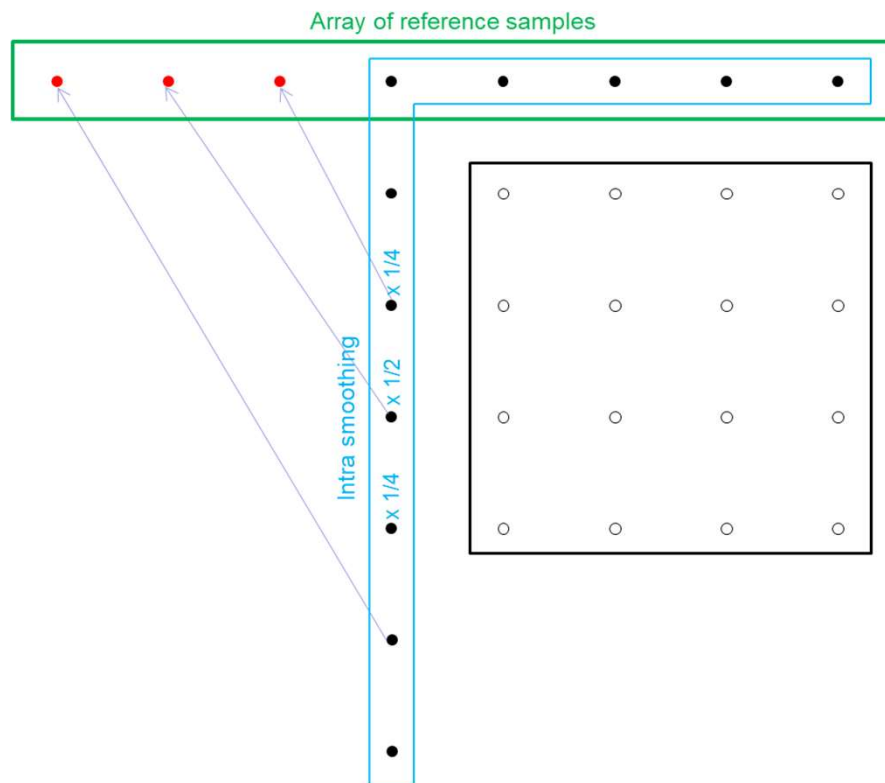


Intra prediction

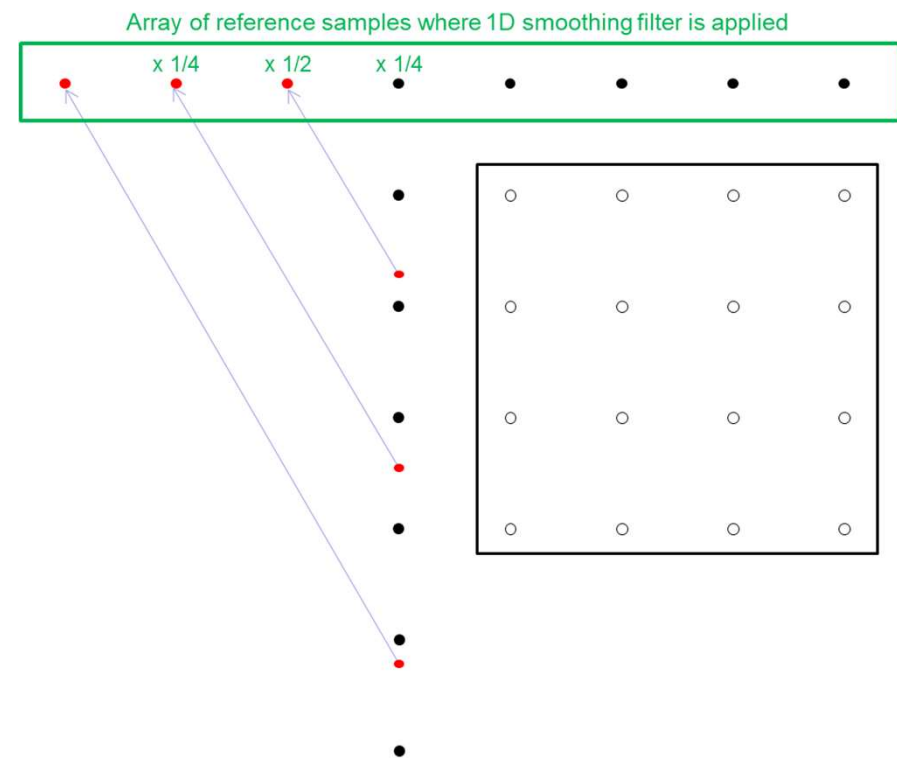


Intra filtering improvements

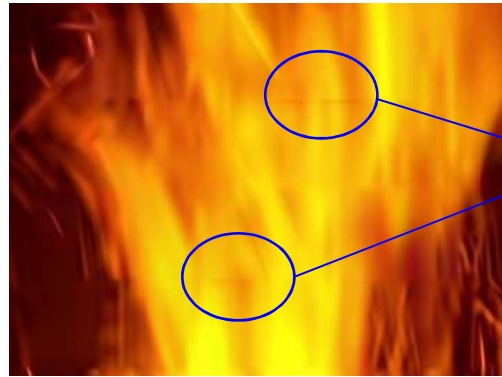
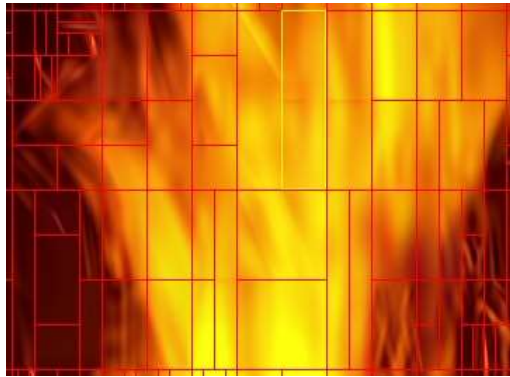
JEM7.0



PEM



PDPC bug fix

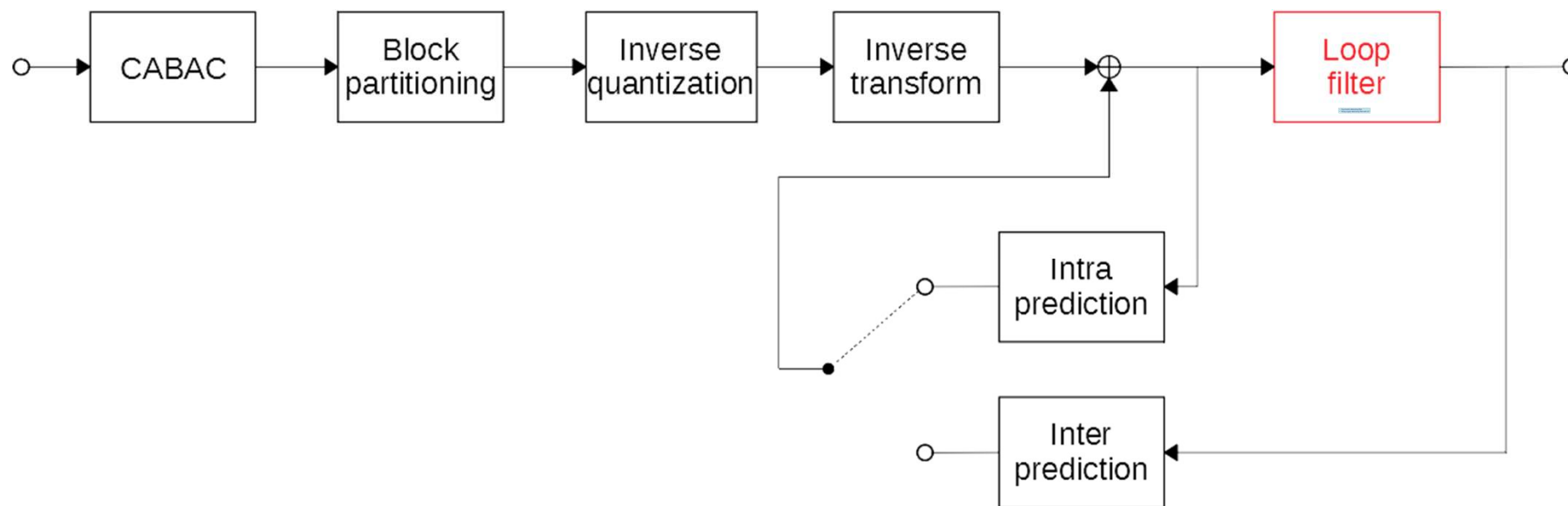


lines appearing at the center
of P-PDPC predicted blocks
in JEM7.0

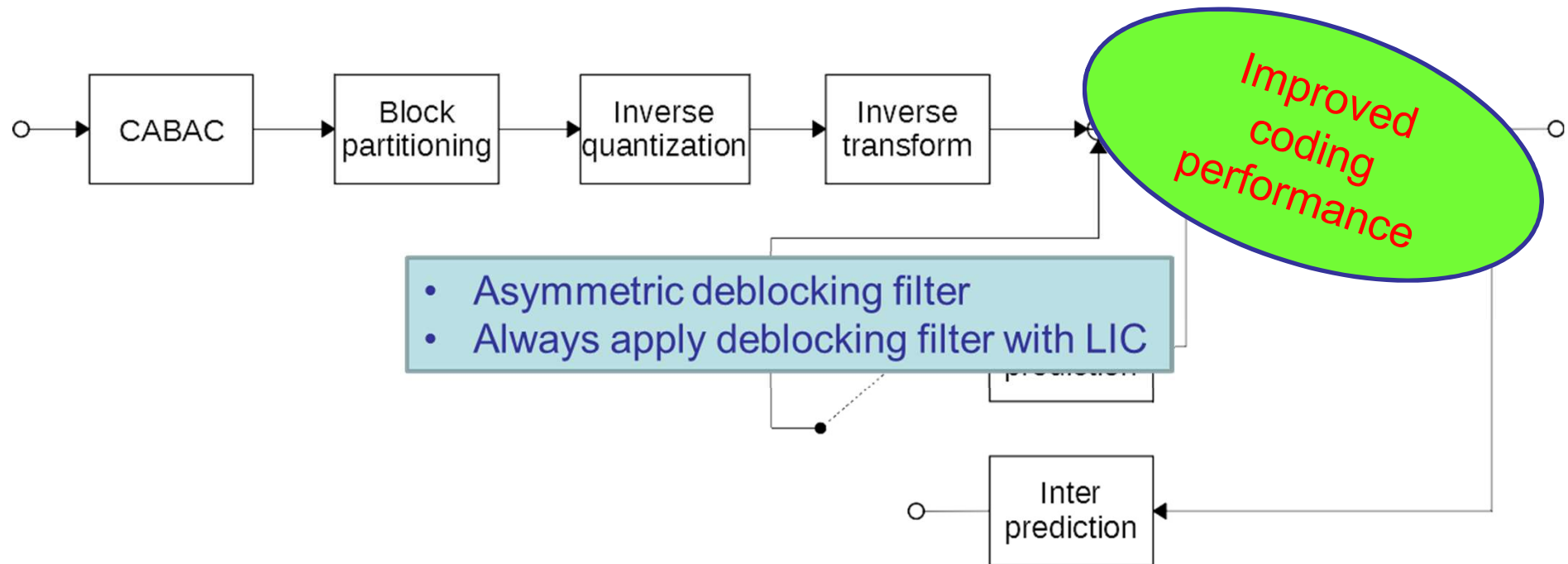
$$\begin{aligned}
 p[x, y] &= \left\{ \left(c_1^{(v)} \gg \lfloor y/d_y \rfloor \right) r[x, -1] - \left(c_2^{(v)} \gg \lfloor y/d_y \rfloor \right) r[-1, -1] + \left(c_1^{(h)} \gg \lfloor x/d_x \rfloor \right) r[-1, y] \right. \\
 &\quad \left. - \left(c_2^{(h)} \gg \lfloor x/d_x \rfloor \right) r[-1, -1] + b[x, y]q[x, y] + 64 \right\} \gg 7
 \end{aligned}$$

Bit shifts not correctly implemented for blocks with width or height bigger than 64
 → coefficients applied to unfiltered reference samples too high in the middle of blocks

Loop filter

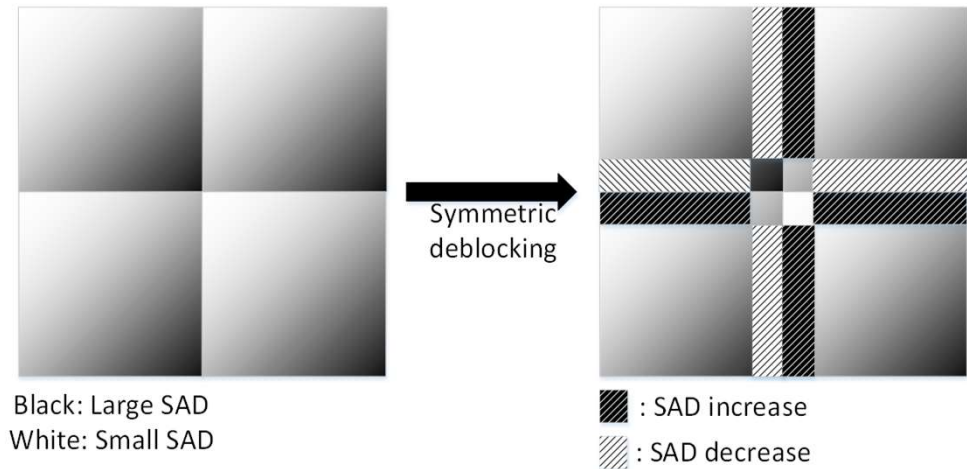


Loop filter

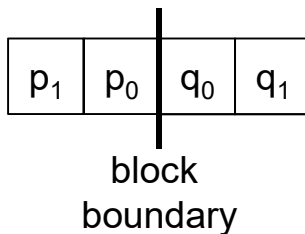


Asymmetric deblocking filter

Deblocking filter has a different effect on Sum of Absolute Differences at different boundaries of a CU



⇒ Asymmetric Deblocking Filter for weak filtering:



$$\Delta = (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$$

JEM7.0

$$p'_0 = p_0 + \Delta$$

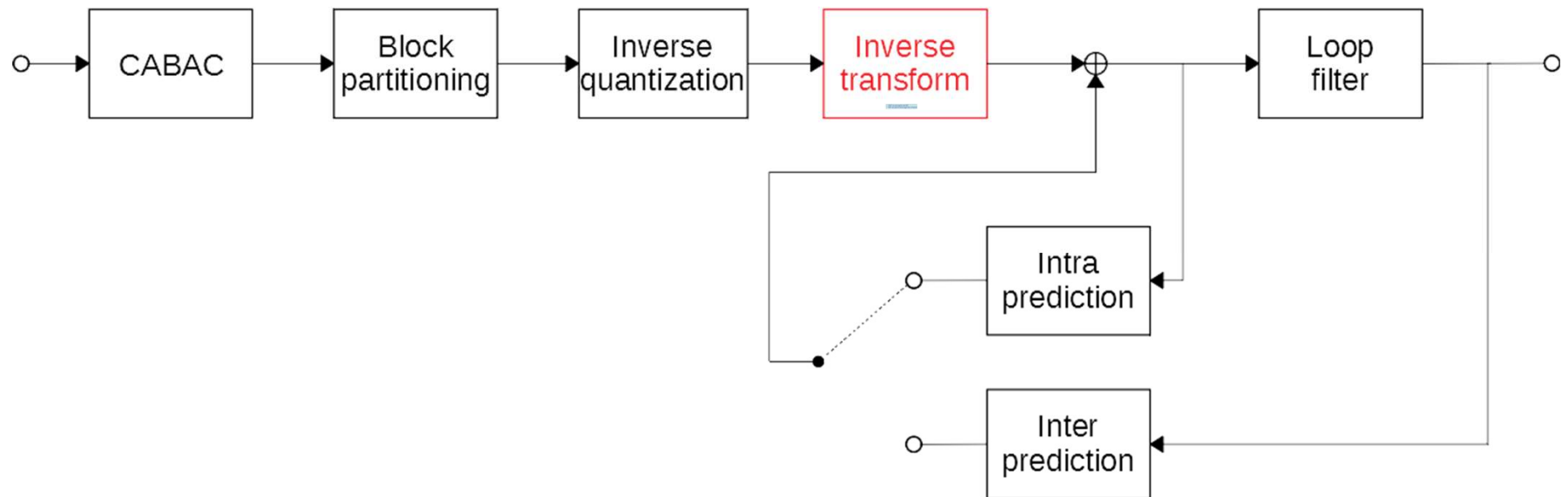
$$q'_0 = q_0 - \Delta$$

PEM

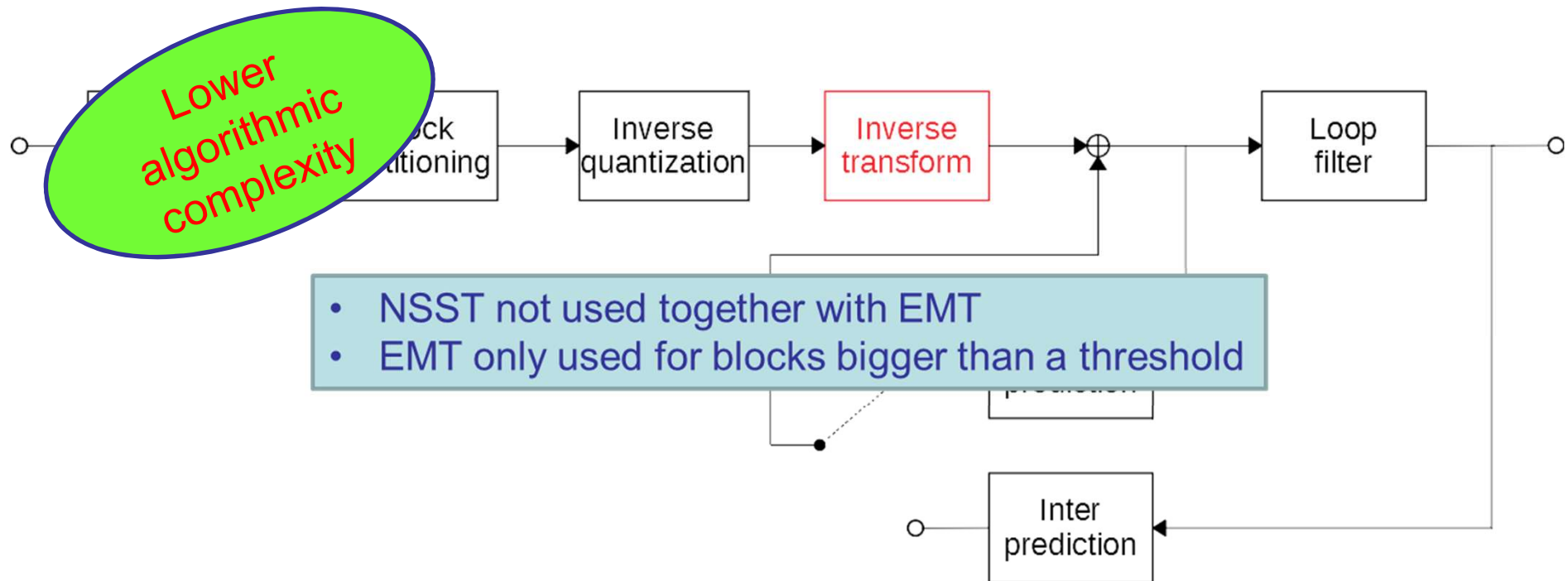
$$p'_0 = p_0 + \Delta \times w_{bottomright}$$

$$q'_0 = q_0 - \Delta \times w_{topleft}$$

Inverse transform



Inverse transform



Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - Detailed explanation of major PEM coding tools
 - **Some more results for information**
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

Lower complexity version of PEM = PEM without tri-tree partitioning

	Constraint Set 1					
	Over JEM7.0(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-0.47%	0.80%	0.20%	0.02	-0.02	0.00
CatRobot1	0.42%	1.34%	1.28%	-0.01	-0.01	-0.02
DaylightRoad2	0.15%	2.28%	1.79%	0.00	-0.03	-0.02
ParkRunning3	-1.10%	-0.10%	-1.06%	0.03	0.00	0.01
Campfire	-0.26%	-0.67%	-0.72%	0.01	0.02	0.01
BQTerrace	0.65%	0.75%	-0.55%	-0.01	-0.01	0.01
RitualDance	-1.01%	0.37%	0.68%	0.05	-0.01	-0.02
MarketPlace	-0.22%	1.63%	1.41%	0.01	-0.02	-0.02
BasketballDrive	-0.53%	1.42%	0.83%	0.01	-0.02	-0.02
Cactus	1.80%	1.31%	1.35%	-0.05	-0.01	-0.02
Average SDR-A	-0.25%	0.73%	0.30%	0.01	-0.01	0.00
Average SDR-B	0.14%	1.10%	0.74%	0.00	-0.01	-0.01
Average all	-0.06%	0.91%	0.52%	0.00	-0.01	-0.01
Enc Time[%]	60%					
Dec Time[%]	55%					

Panasonic's NextSoftware = PEM tools in HHI's NextSoftware

	Constraint Set 1					
	Over JEM7.0(GCC6.2.0, 1 core for decode, dec YUV output)					
	BD-rate(Y)	BD-rate(U)	BD-rate(V)	BD-PSNR(Y)	BD-PSNR(U)	BD-PSNR(V)
FoodMarket4	-2.47%	-2.47%	-3.03%	0.09	0.05	0.08
CatRobot1	-2.01%	-4.85%	-4.05%	0.05	0.05	0.08
DaylightRoad2	-3.00%	-5.61%	-5.33%	0.06	0.07	0.07
ParkRunning3	-2.15%	-2.35%	-4.06%	0.06	0.04	0.05
Campfire	-2.51%	-3.24%	-4.22%	0.07	0.10	0.09
BQTerrace	-1.50%	-2.74%	-2.45%	0.04	0.04	0.04
RitualDance	-2.63%	-4.89%	-5.60%	0.13	0.11	0.15
MarketPlace	-2.56%	-4.29%	-4.61%	0.07	0.05	0.06
BasketballDrive	-2.49%	-4.11%	-4.25%	0.07	0.06	0.10
Cactus	-0.81%	-4.55%	-2.63%	0.03	0.05	0.05
Average SDR-A	-2.43%	-3.71%	-4.14%	0.07	0.06	0.07
Average SDR-B	-2.00%	-4.12%	-3.91%	0.07	0.06	0.08
Average all	-2.21%	-3.91%	-4.02%	0.07	0.06	0.08
Enc Time[%]	98%					
Dec Time[%]	52%					

Outline

- What is PEM?
- Overview of PEM coding tools
- Results
- Detailed explanation of major PEM coding tools
- Some more results for information
- **Conclusion**
- Annex 1: individual results of some major tools
- Annex 2: additional information on PEM coding tools

Conclusion

PEM = optimized JEM7.0

Increased coding efficiency

-2.28%

Significant decoder complexity reduction

56%

Hardware friendly

Coding tools for further JVET's developments

- Based on JVET's previous work
- Optimizations and decreased complexity = easier to test new coding tools
- Consistent behaviour independently from implementation

Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - Detailed explanation of major PEM coding tools
 - Some more results for information
 - Conclusion
-
- **Annex 1: individual results of some major tools**
 - Annex 2: additional information on PEM coding tools

Annex 1: results of tools ON/OFF tests for selected tools (constraint set 1)

Tested tool name	“tool-off” test results					“tool-on” test results				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
TRI_TREE	1.98%	4.15%	4.26%	64%	97%	-1.95%	-4.14%	-4.13%	159%	104%
INTERTOOL_COMB	-0.16%	-0.40%	-0.38%	120%	133%	0.50%	0.99%	0.62%	77%	74%
TRIANGLE	0.40%	0.46%	0.64%	93%	101%	-0.27%	-0.31%	-0.54%	101%	101%
FRUC_LOWBW	0.01%	0.14%	0.26%	100%	99%	0.01%	-0.08%	-0.25%	101%	102%
FRUC_LOWCOMP	-0.03%	-0.06%	0.11%	109%	110%	0.15%	0.11%	0.08%	90%	89%
OBMC_LOWCOMP	-0.12%	0.18%	0.07%	100%	104%	0.27%	0.25%	0.10%	97%	95%
BIO_LOWCOMP	-0.03%	0.06%	0.11%	100%	101%	0.06%	0.04%	0.08%	99%	99%
BIO_IMPROVE	0.18%	0.01%	0.03%	100%	100%	-0.08%	0.03%	0.11%	100%	101%
ATMVP_LOWCOMP	0.00%	-0.01%	0.02%	99%	105%	0.05%	0.11%	-0.02%	98%	95%
SKIP_SIMP	0.10%	0.06%	0.31%	100%	101%	-0.01%	0.18%	-0.19%	98%	100%
INTRA_IMPROVE	0.20%	0.33%	0.47%	98%	100%	-0.12%	0.10%	0.00%	100%	101%
TRANS_LOWCOMP	-0.23%	-0.11%	0.27%	117%	100%	0.32%	-0.05%	0.01%	91%	101%
DBF_IMPROVE	0.15%	0.02%	-0.02%	100%	100%	-0.15%	0.08%	-0.09%	100%	101%
CABAC_IMPROVE	0.38%	-0.14%	0.50%	99%	101%	-0.38%	-0.04%	-0.25%	101%	98%
BIO_CODEREFINE	0.00%	0.00%	0.00%	100%	105%	0.00%	0.00%	0.00%	100%	99%
ALF_CODEREFINE	0.00%	0.00%	0.00%	99%	105%	0.00%	0.00%	0.00%	100%	98%

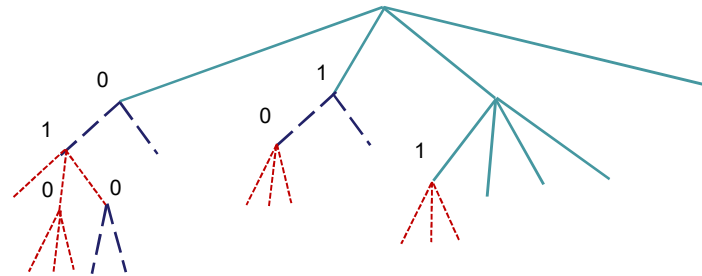
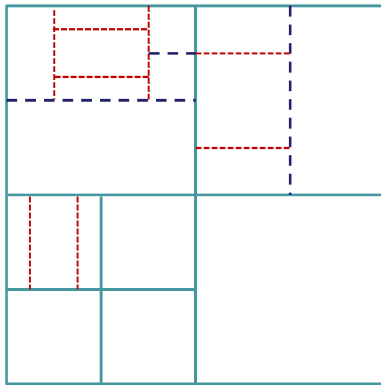
Annex 1: results of tools ON/OFF tests for selected tools (constraint set 2)

Tested tool name	“tool-off” test results					“tool-on” test results				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
TRI_TREE	1.86%	3.72%	4.48%	57%	96%	-1.92%	-4.92%	-4.83%	178%	103%
INTERTOOL_COMB	-0.17%	-0.80%	-0.83%	133%	124%	0.49%	1.42%	1.81%	70%	79%
TRIANGLE	0.64%	1.21%	0.75%	95%	102%	-0.62%	-0.53%	-0.33%	99%	100%
FRUC_LOWBW	-0.08%	0.09%	-0.22%	100%	100%	0.03%	0.19%	0.22%	99%	102%
FRUC_LOWCOMP	-0.10%	0.31%	-0.22%	114%	113%	-0.04%	0.26%	0.54%	85%	88%
OBMC_LOWCOMP	-0.30%	-0.34%	-0.22%	101%	107%	0.32%	0.53%	0.50%	97%	93%
BIO_LOWCOMP	-0.09%	0.06%	0.02%	101%	101%	0.00%	0.39%	0.35%	99%	98%
BIO_IMPROVE	0.00%	0.00%	0.00%	100%	100%	0.00%	0.00%	0.00%	100%	101%
ATMVP_LOWCOMP	-0.14%	-0.15%	0.21%	100%	105%	0.07%	0.10%	0.06%	97%	95%
SKIP_SIMP	-0.22%	0.10%	0.04%	100%	101%	0.08%	0.14%	0.20%	98%	99%
INTRA_IMPROVE	0.03%	-0.14%	-0.46%	99%	100%	-0.12%	0.62%	0.16%	100%	100%
TRANS_LOWCOMP	-0.32%	0.13%	0.01%	109%	101%	0.19%	0.37%	0.27%	96%	100%
DBF_IMPROVE	-0.01%	0.12%	0.04%	100%	100%	-0.22%	0.01%	0.24%	101%	100%
CABAC_IMPROVE	0.36%	-0.37%	0.05%	99%	101%	-0.50%	0.83%	-0.42%	101%	98%
BIO_CODEREFINE	0.00%	0.00%	0.00%	100%	100%	0.00%	0.00%	0.00%	100%	100%
ALF_CODEREFINE	0.00%	0.00%	0.00%	100%	105%	0.00%	0.00%	0.00%	100%	96%

Outline

- What is PEM?
 - Overview of PEM coding tools
 - Results
 - Detailed explanation of major PEM coding tools
 - Some more results for information
 - Conclusion
-
- Annex 1: individual results of some major tools
 - Annex 2: additional information on PEM coding tools

Block partitioning: quad-tree, binary-tree and tri-tree partitioning example



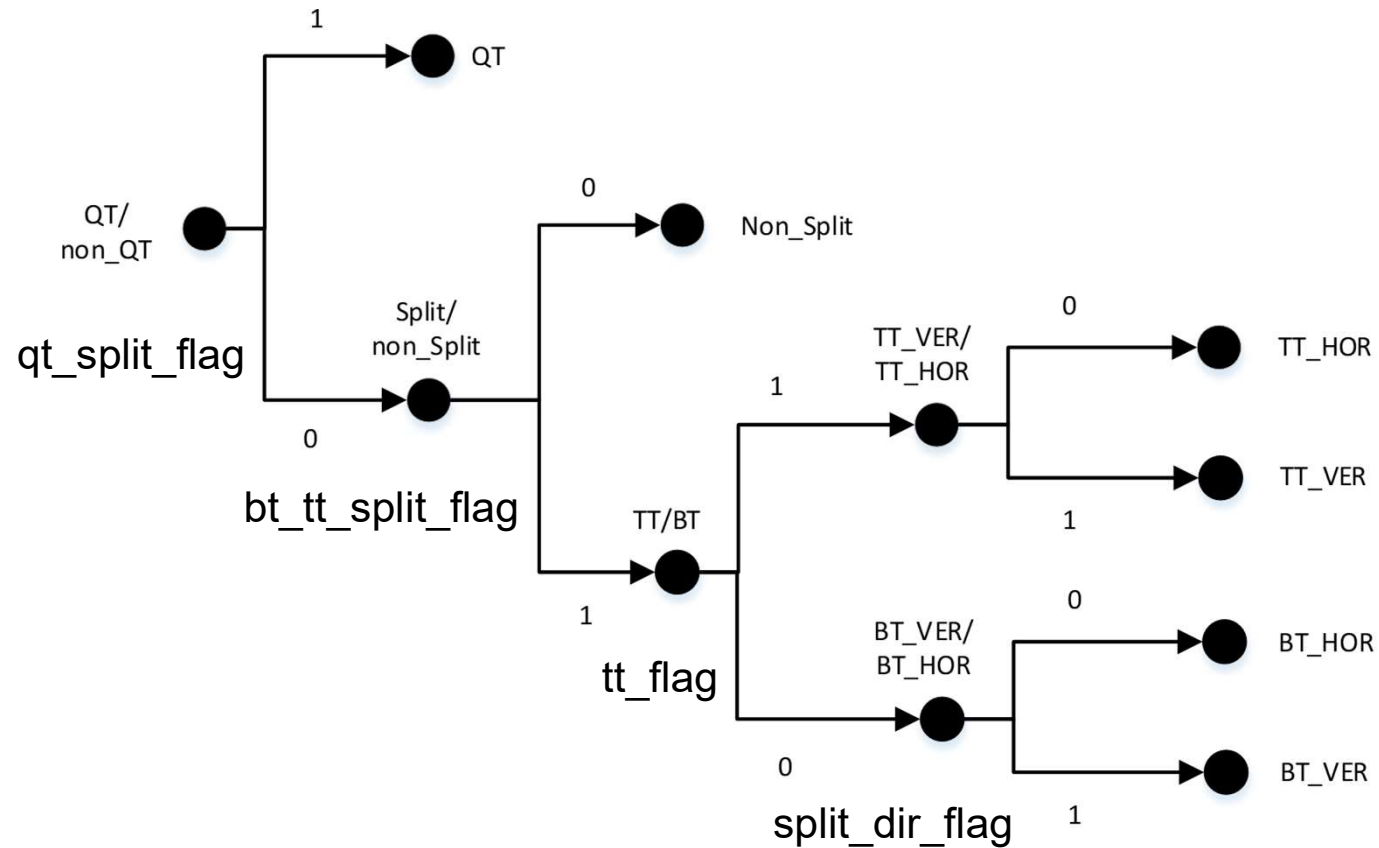
0 → Horizontal Tri-tree

1 → Vertical Tri-tree

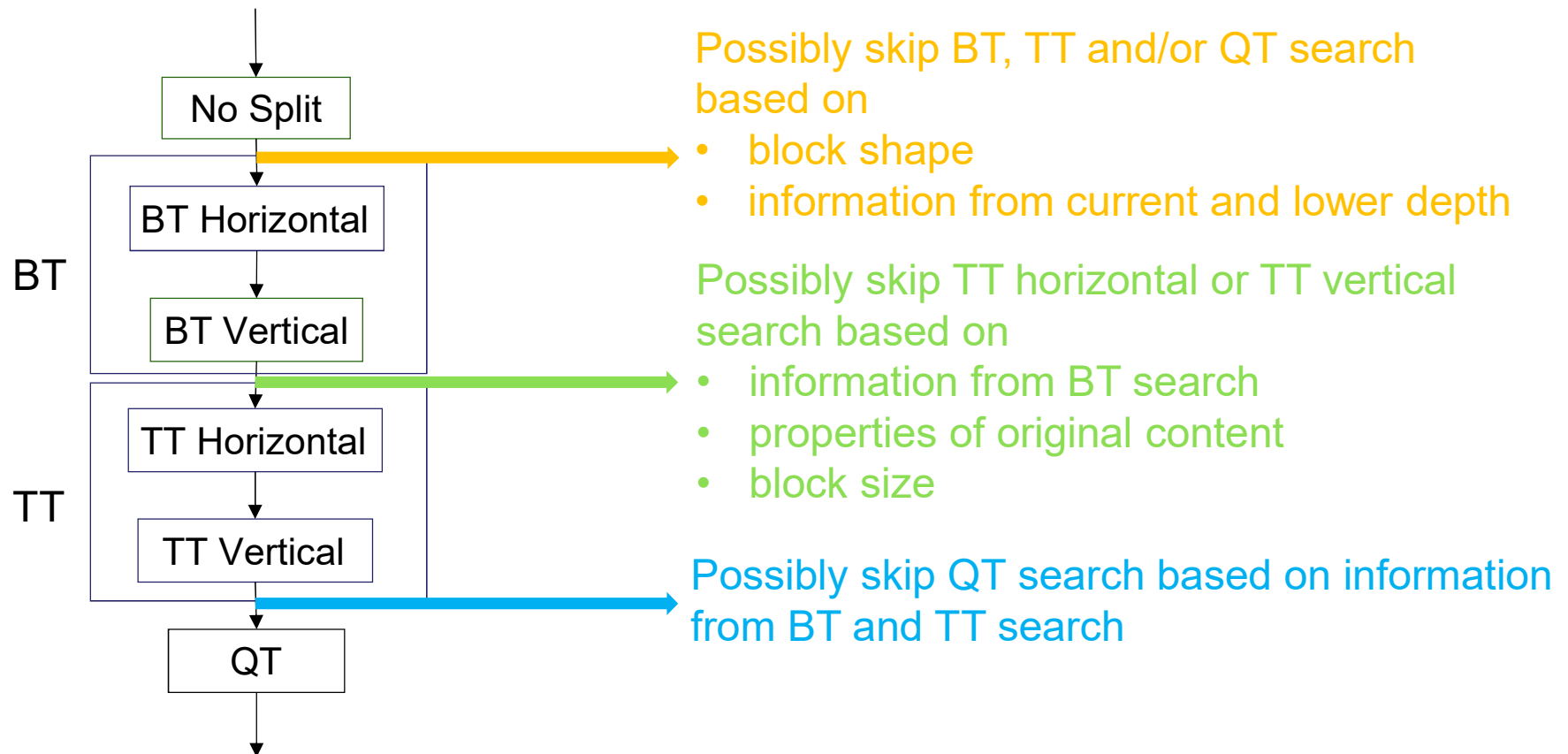
0 → Horizontal BT

1 → Vertical BT

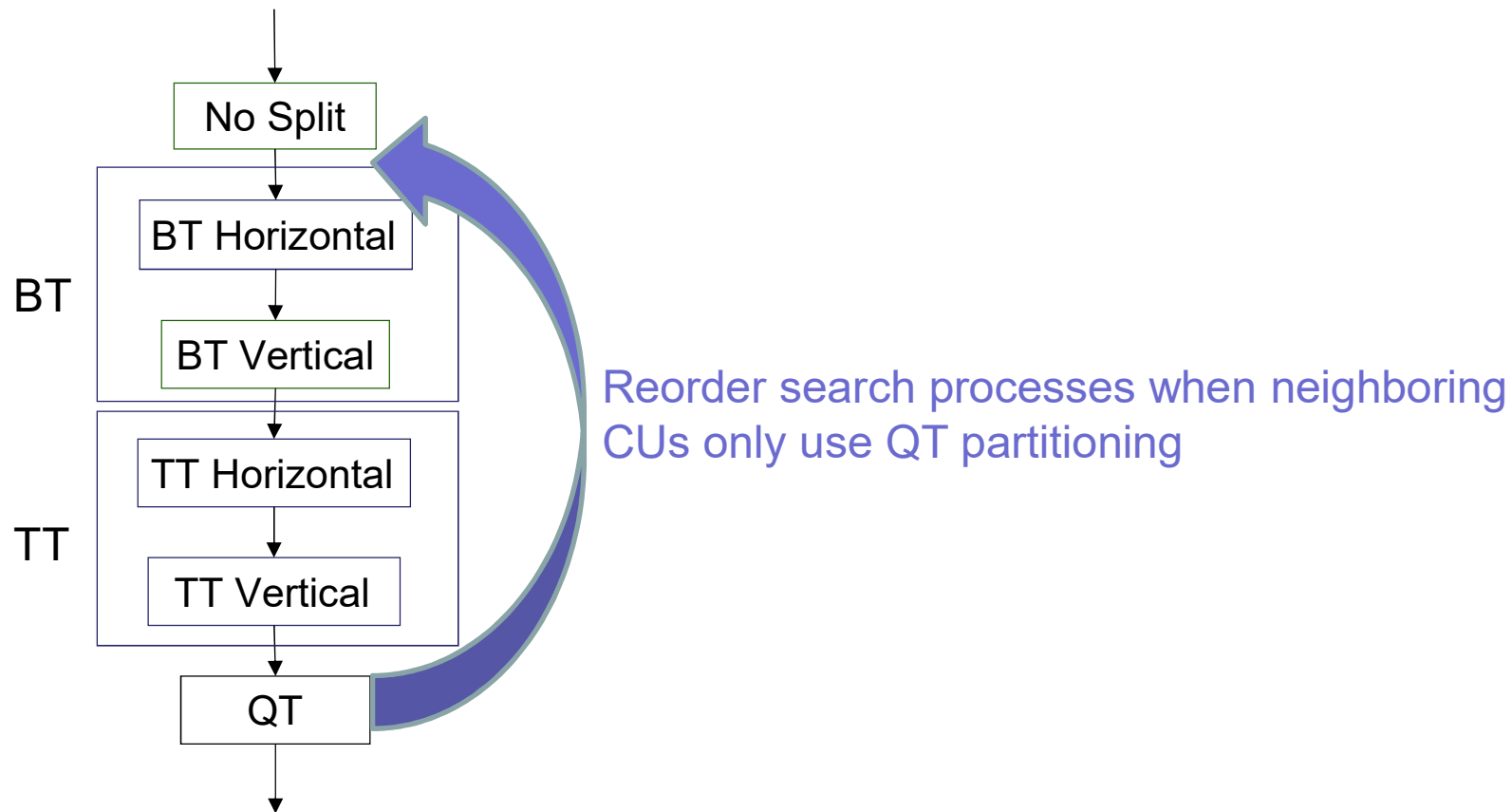
Block partitioning: binary-tree and tri-tree syntax



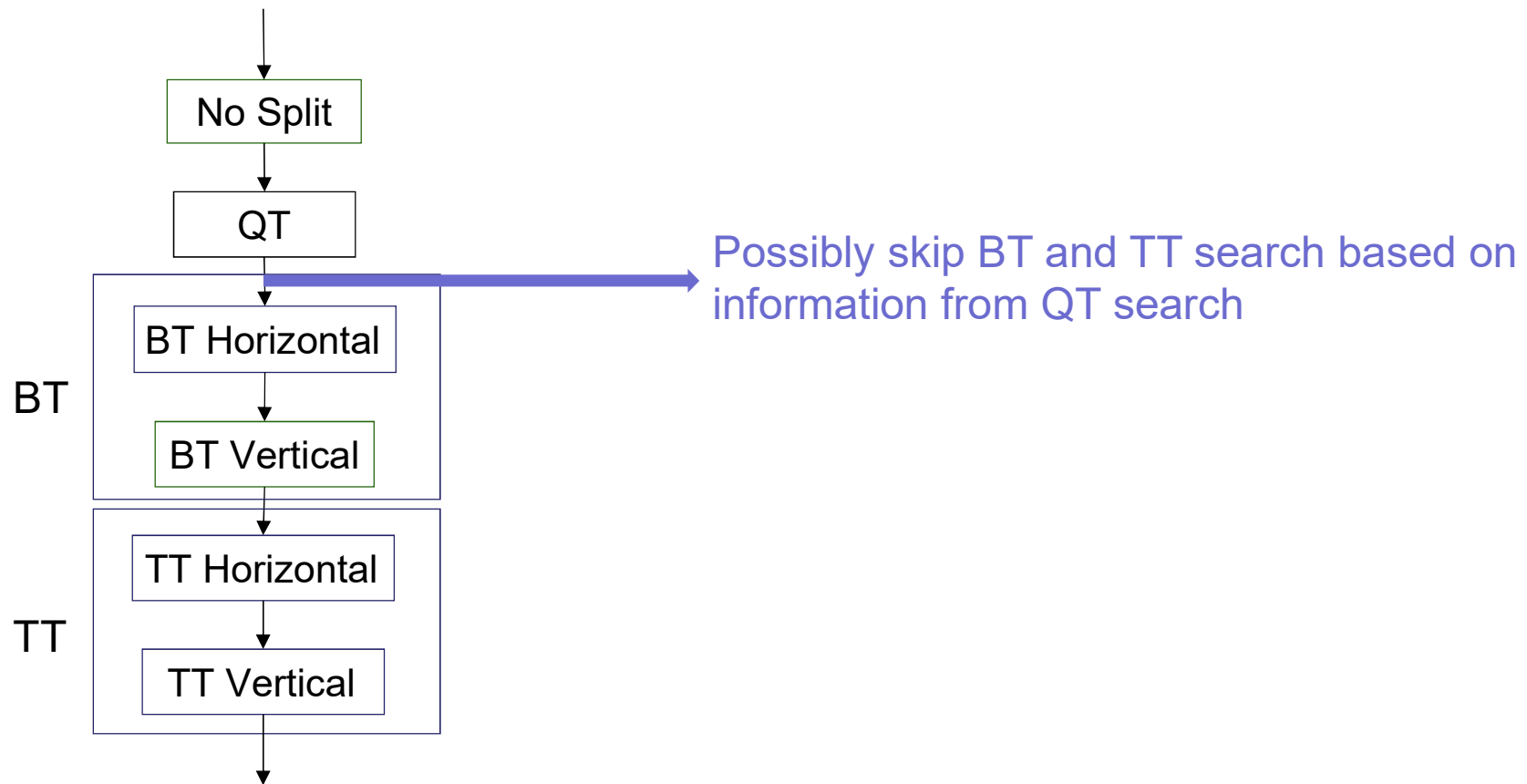
Block partitioning: selection process at the encoder



Block partitioning: selection process at the encoder

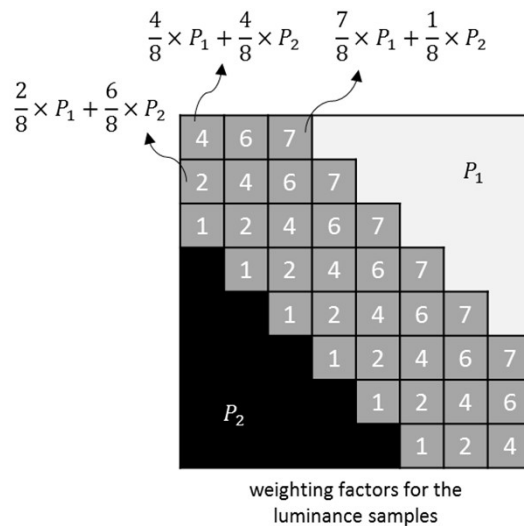


Block partitioning: selection process at the encoder

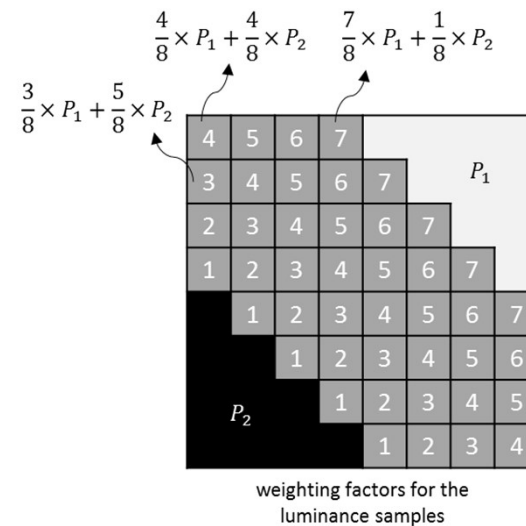
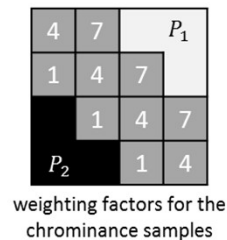


Triangular prediction units: weighting the two triangles

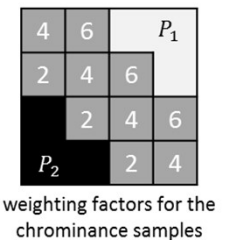
- Two sets of pre-defined weights for each samples for both luma and chroma
- Set of weights applied is based on differences between selected motion vectors



Set 1



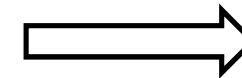
Set 2



Triangular prediction units: encoder selection

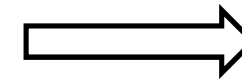
- Only the first X possible values of triangle_idx are checked

If CU of same size and position was previously tested
and mode selected was not the triangular prediction OR
was skip mode



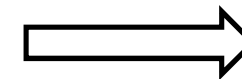
X = 7

If $\frac{CUwidth}{CUheight} = 32$ or $\frac{CUheight}{CUwidth} = 32$



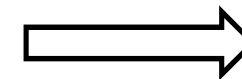
X = 7

If $\frac{CUwidth}{CUheight} = 16$ or $\frac{CUheight}{CUwidth} = 16$



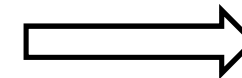
X = 14

If $\frac{CUwidth}{CUheight} = 8$ or $\frac{CUheight}{CUwidth} = 8$



X = 40

Otherwise



X = 62

Optimized combinations of inter prediction tools: detailed tests results

Tool name	Tool-off test					Tool-on test				
	BD-rate			Runtime		BD-rate			Runtime	
	Y	Cb	Cr	Enc	Dec	Y	Cb	Cr	Enc	Dec
FRUC SIMP	-0.02%	-0.21%	-0.17%	104%	122%	0.26%	0.60%	0.48%	94%	80%
LIC SIMP	-0.09%	0.06%	0.21%	113%	100%	0.22%	0.37%	0.18%	87%	101%
BIO SIMP	-0.10%	-0.07%	0.01%	101%	105%	0.05%	0.03%	0.09%	99%	99%
OBMC COMB	0.11%	-0.12%	0.00%	99%	105%	0.16%	0.52%	0.33%	94%	89%
All tools	-0.16%	-0.40%	-0.38%	120%	133%	0.50%	0.99%	0.62%	77%	74%

FRUC bandwidth reduction: finding the pre-defined memory block

Goal for FRUC bandwidth per CTU =
0.4 × worst case bandwidth of motion
compensation in JEM7.0

Number of reference frames	Size N × N of pre- defined memory block
1	374 × 374
2	264 × 264
3	216 × 216
4	187 × 187
5	167 × 167
6	152 × 152
7	141 × 141
8	132 × 132

Position of pre-defined memory block
in each reference frame ref for CTU at
location [i,j]

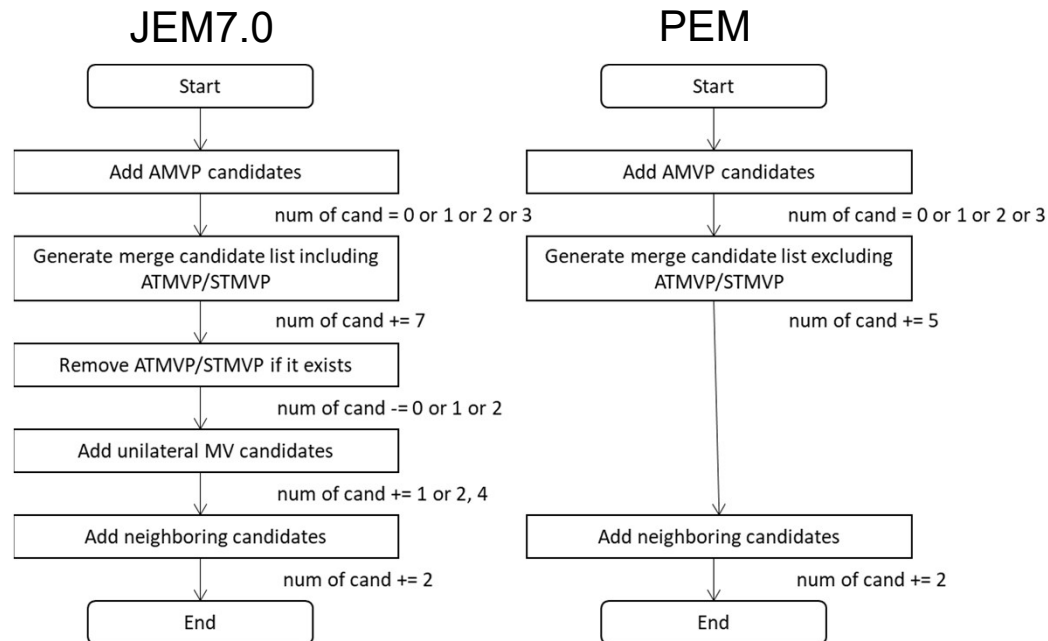
$$\begin{aligned}
 &locationX \\
 &= i - ((N - CTUsize - 1) \gg 1) \\
 &+ \frac{\sum_{CU_MV \in U} (CU_MV_x \times CU_area)}{\sum_{CU_MV \in U} CU_area}
 \end{aligned}$$

$$\begin{aligned}
 &locationY \\
 &= j - ((N - CTUsize - 1) \gg 1) \\
 &+ \frac{\sum_{CU_MV \in U} (CU_MV_y \times CU_area)}{\sum_{CU_MV \in U} CU_area}
 \end{aligned}$$

(U is the set of motion vectors in CTUs left and
above that point to ref)

FRUC complexity reduction: MV candidate list simplifications

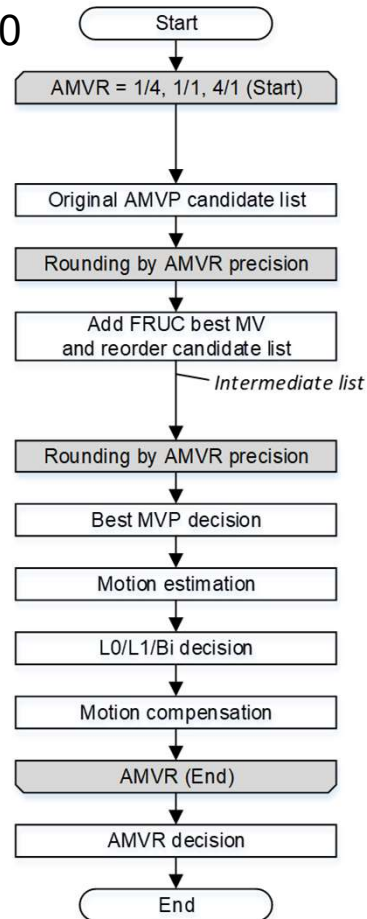
- Refinement of MV can be skipped under certain conditions
- Generation of interpolated MV field for unilateral MV candidate is removed
- ATMVP/STMVP candidates inclusion into MV candidate list is removed



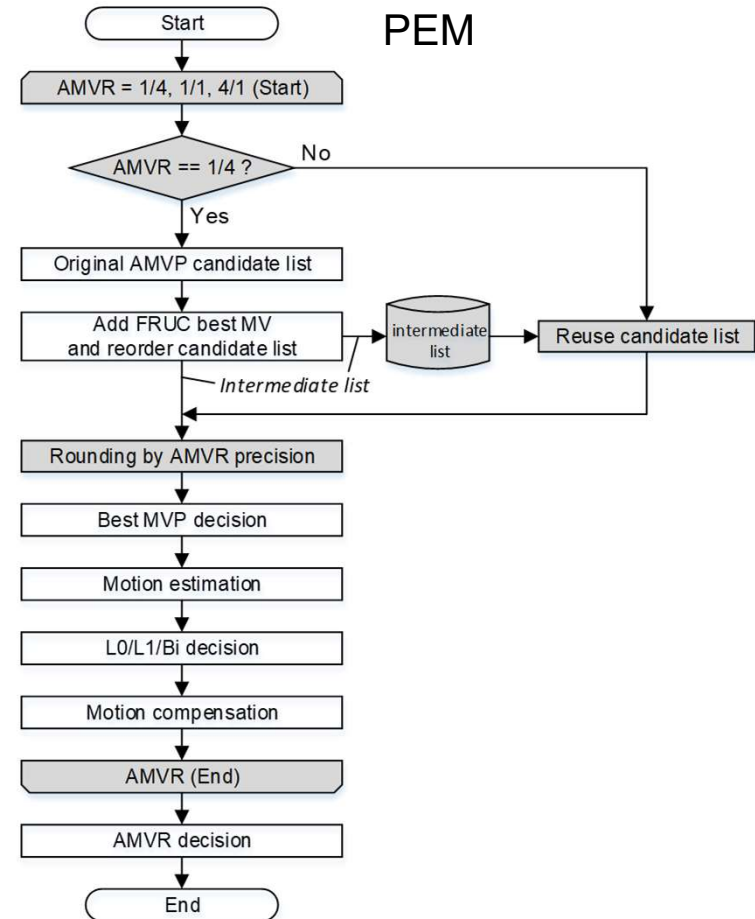
AMVP candidate list simplification

Skip rounding MV
to AMVR
precision before
FRUC process to
avoid applying
FRUC process
too often

JEM7.0

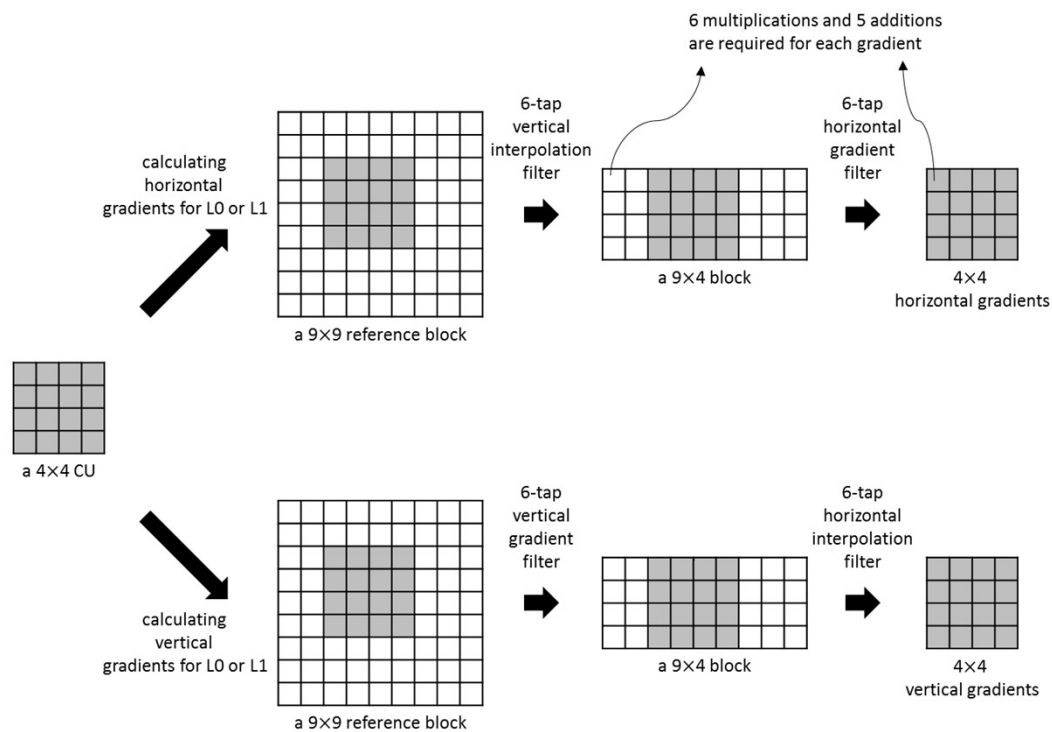


PEM

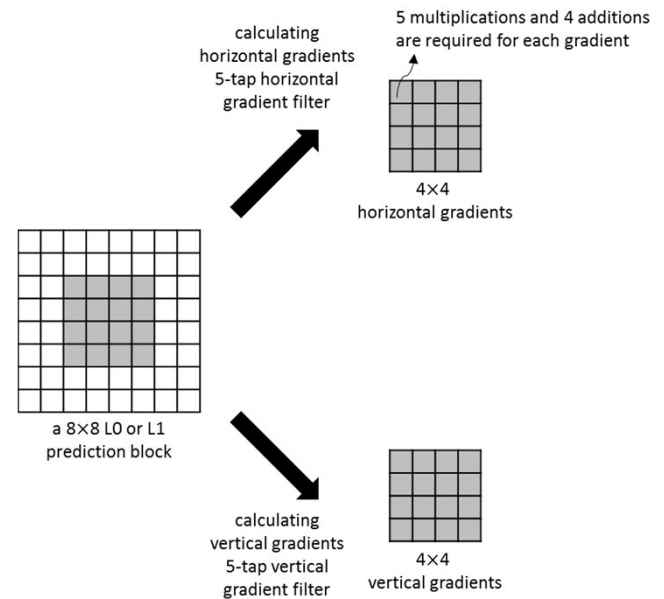


BIO complexity reduction: simplified interpolation and gradient calculation

JEM7.0

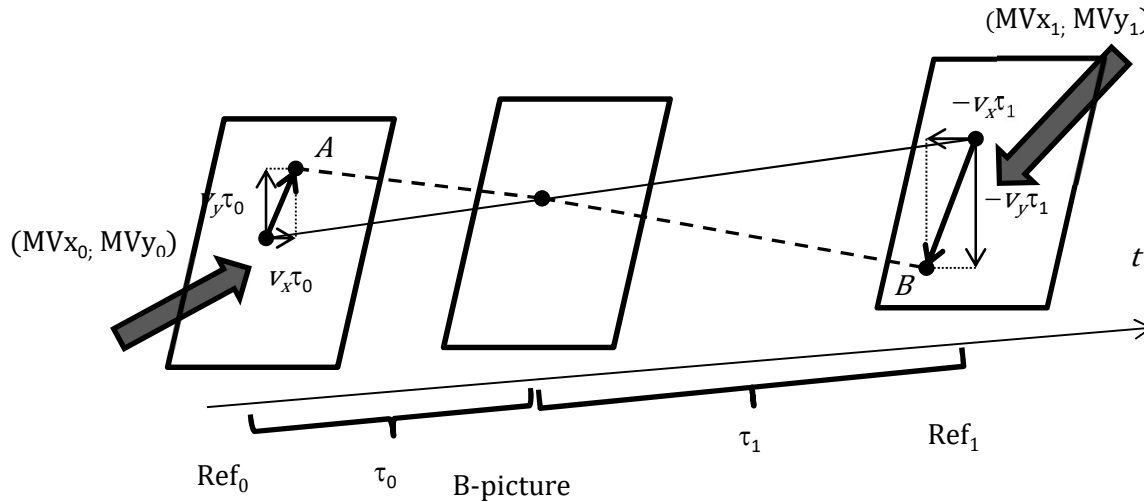


PEM



- BIO gradient filter: [2, -9, 0, 9, -2]
- Prediction block is padded

BIO improvements: Different equations for calculating Vx and Vy



$$v_x[i, j] = \frac{s_3 s_5 - s_2 s_6}{s_1 s_5 - s_2 s_4}$$

$$v_y[i, j] = \frac{s_1 s_6 - s_3 s_4}{s_1 s_5 - s_2 s_4}$$

$$s_2 = s_4 = \sum_{\Omega} \left(\tau_0 I_x^{(0)}[i', j'] + \tau_1 I_x^{(1)}[i', j'] \right) \left(\tau_0 I_y^{(0)}[i', j'] + \tau_1 I_y^{(1)}[i', j'] \right)$$

$$s_3 = - \sum_{\Omega} \left(P^{(0)}[i', j'] - P^{(1)}[i', j'] \right) \left(\tau_0 I_x^{(0)}[i', j'] + \tau_1 I_x^{(1)}[i', j'] \right)$$

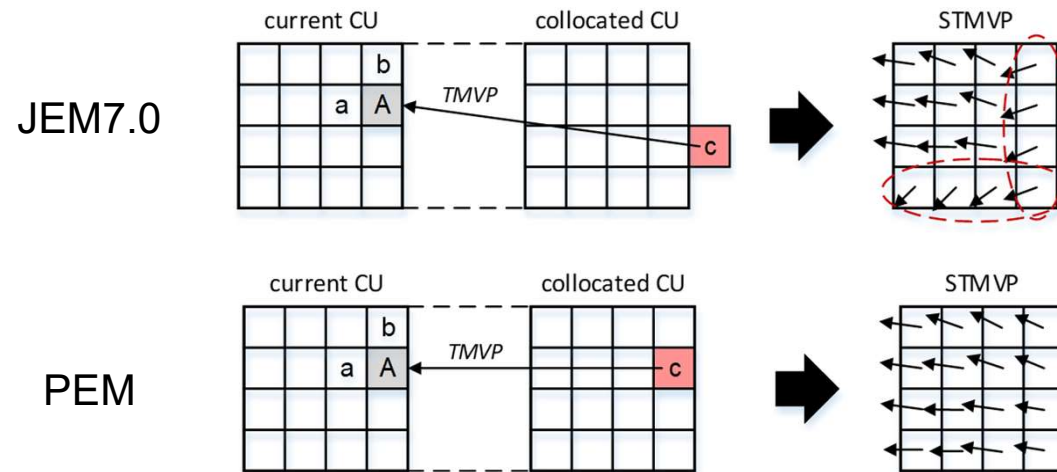
$$s_6 = - \sum_{\Omega} \left(P^{(0)}[i', j'] - P^{(1)}[i', j'] \right) \left(\tau_0 I_y^{(0)}[i', j'] + \tau_1 I_y^{(1)}[i', j'] \right)$$

$$s_1 = \sum_{\Omega} \left(\tau_0 I_x^{(0)}[i', j'] + \tau_1 I_x^{(1)}[i', j'] \right)^2$$

$$s_5 = \sum_{\Omega} \left(\tau_0 I_y^{(0)}[i', j'] + \tau_1 I_y^{(1)}[i', j'] \right)^2$$

Modifications to ATMVP/STMVP

- Different TMVP derivation for calculation of local STMVP



- Restriction on the size of sub-CUs

$$\left. \begin{aligned} SubCU_{width} &= \max(CU_{width} \gg depth, 4) \\ SubCU_{height} &= \max(CU_{height} \gg depth, 4) \end{aligned} \right\} \text{with depth} = 3 \text{ in PEM}$$

Modifications to deblocking filter

Bs decision criteria:

Conditions	Bs
At least one of the blocks uses Intra prediction	2
At least one of the blocks has non-zero coded residual coefficient	1
Absolute difference between corresponding spatial motion vector components of the two blocks are ≥ 1 in units of integer pixels	1
Motion-compensated prediction for the two blocks refers to different reference pictures or the number of motion vectors is different for the two blocks	1
At least one of the blocks uses LIC	1
Otherwise	0

Always apply deblocking filter when LIC is used

NSST and EMT constraints

EMT transforms for blocks with width ≤ 64 and height ≤ 64

Horizontal primary
transform

followed by

Vertical primary
transform

may be followed by

Secondary
transform

