

JVET-P0382

NON-CE1: SIMPLIFIED MOTION COMPENSATION IN RPR

Kai Zhang, Li Zhang, Hongbin Liu, Zhipin Deng, Jizheng Xu, Na Zhang



Background of RPR

- The reference sample position (x_r, y_r) is derived based on the current sample position (x_c, y_c) and a scaling ratio between the $W_r \times H_r$ conformance window in the reference picture and the $W_c \times H_c$ conformance window in the current picture: $x_r = x_c \times W_r / W_c$ and $y_r = y_c \times H_r / H_c$.
- In the cases when the scaling ratio is larger than one, a higher bandwidth and more computations are required for a $M \times N$ block.
 - *Ratio = 1*
 - Bandwidth: $(M + 7) \times (N + 7)$ integer samples
 - Multiplications: $8 \times (M \times (N + 7) + M \times N)$
 - *Ratio = 2*
 - Bandwidth: $(2M + 7) \times (2N + 7)$ integer samples
 - Multiplications: $8 \times (M \times (2N + 7) + M \times N)$

Proposed

- Apply the 6-tap interpolation filters used in the affine motion compensation on a luma $M \times N$ block satisfying certain conditions
 - *Test #1: The luma block is bi-predicted.*
 - *Test #2: The luma block is bi-predicted and $(M \times N \leq 64 \mid \mid M == 4 \mid \mid N == 4)$*
- With the both the two tests when the scaling ratio is equal to 2:
 - *The required bandwidth in the worst case is decreased to 82%*
 - *The required multiplications in the worst case are decreased to 67%*

Experimental results on CE1-anchor (Test 1)

■ Ratio = 1.5

	PSNR1					PSNR2				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	-0.01%	0.13%	-0.02%	100%	100%	-0.01%	0.15%	0.03%	100%	100%
Class C	-0.05%	-0.16%	-0.21%	99%	98%	-0.03%	-0.20%	-0.18%	99%	98%
Class E	0.04%	0.59%	0.67%	100%	97%	0.06%	0.55%	0.55%	100%	97%
Overall	-0.01%	0.15%	0.09%	100%	99%	0.00%	0.13%	0.09%	100%	99%
Class D	-0.01%	0.02%	0.27%	101%	100%	0.00%	0.13%	0.29%	101%	100%
Class F	-0.20%	-0.28%	-0.19%	100%	98%	-0.12%	-0.12%	-0.14%	100%	98%

■ Ratio = 2.0

	PSNR1					PSNR2				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	0.09%	0.27%	0.16%	99%	100%	0.02%	0.22%	0.14%	99%	100%
Class C	-0.01%	-0.01%	-0.12%	98%	97%	-0.01%	-0.01%	0.01%	98%	97%
Class E	0.32%	0.10%	0.13%	100%	96%	0.22%	0.14%	0.01%	100%	96%
Overall	0.11%	0.13%	0.06%	99%	98%	0.06%	0.12%	0.06%	99%	98%
Class D	-0.01%	0.14%	-0.21%	100%	96%	-0.04%	0.06%	0.01%	100%	96%
Class F	-0.12%	-0.23%	0.30%	100%	99%	-0.08%	-0.30%	0.30%	100%	99%

Experimental results on CE1-anchor (Test 2)

■ Ratio = 1.5

	PSNR1					PSNR2				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	0.02%	0.25%	0.00%	101%	99%	0.02%	0.27%	-0.01%	101%	99%
Class C	0.00%	0.00%	-0.04%	99%	98%	-0.01%	0.00%	-0.03%	99%	98%
Class E	-0.04%	0.31%	0.55%	100%	99%	-0.03%	0.31%	0.51%	100%	99%
Overall	0.00%	0.18%	0.12%	100%	99%	0.00%	0.19%	0.11%	100%	99%
Class D	-0.02%	0.49%	-0.12%	99%	98%	-0.03%	0.59%	-0.26%	99%	98%
Class F	-0.73%	-0.73%	-0.67%	100%	100%	-0.61%	-0.63%	-0.53%	100%	100%

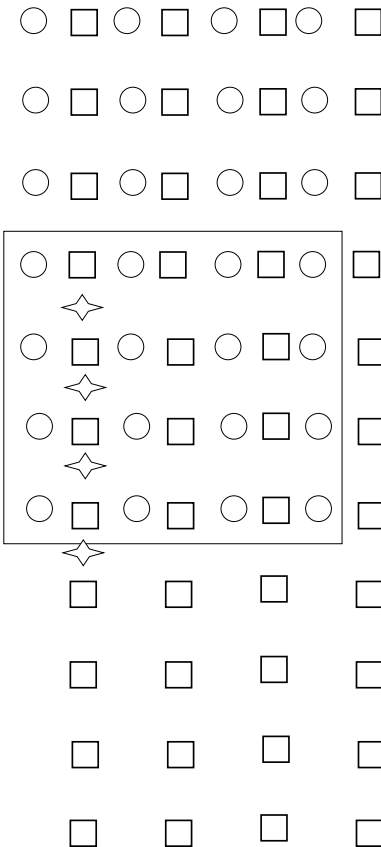
■ Ratio = 2.0

	PSNR1					PSNR2				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	0.01%	0.30%	0.24%	100%	99%	-0.02%	0.28%	0.19%	100%	99%
Class C	-0.10%	0.07%	-0.07%	97%	97%	-0.03%	0.02%	0.05%	97%	97%
Class E	0.11%	-0.07%	0.13%	100%	99%	0.07%	-0.10%	-0.01%	100%	99%
Overall	0.00%	0.13%	0.11%	99%	99%	0.00%	0.10%	0.09%	99%	99%
Class D	0.04%	-0.62%	-0.25%	99%	97%	0.03%	-0.39%	-0.19%	99%	97%
Class F	-0.01%	-0.05%	0.09%	100%	97%	-0.05%	-0.19%	0.12%	100%	97%

Conclusions

- When the scaling ratio equal to 2
 - *The required bandwidth of motion compensation is reduced to 82% in the worst case*
 - *The required number of multiplications is reduced to 67% in the worst case*
- Thanks Alibaba for cross-checking our proposal ([JVET-P0661](#))

Interpolation Process with ratio = 1



Interpolation Process with ratio = 2

