SHARP

Be Original.

Non-CE9: A speed up method of BDOF JVET-O0211

SHARP CORPORATION

Takeshi Chujoh and Tomohiro Ikai

Summary

SHARP
Be Original.

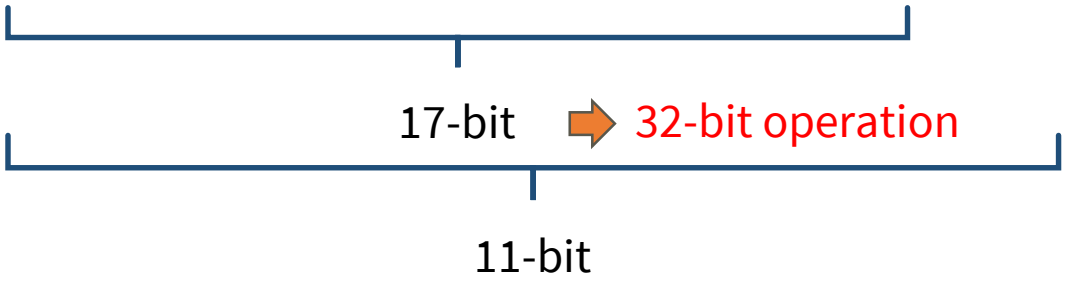
- On the current specification of deriving the gradient values for BDOF, 32-bit operation for each pixel is requested.
- In this contribution, by changing the order of subtraction and shift, 16-bit operations is realized.
  - When SIMD is used, the number of parallel operations become double and the speed of the processing is upped.
- As experimental results, the 1% encoding time has been improved and there was small gain on average.
- This is identical with JVET-O0570(Bytedance) test 2.

Proposal

SHARP
Be Original.

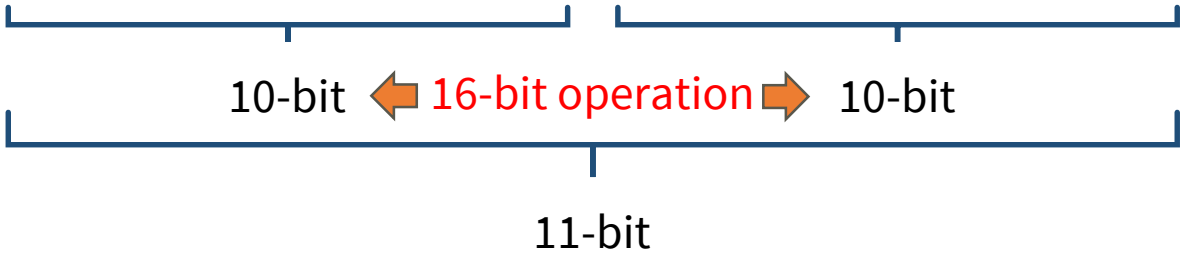
- The dynamic range of variable “predSamples” is 16-bit and the value of variable “shif1” is equal to 6 on CTC.

gradientHL0[x][y] = (predSamplesL0[hx + 1][vy] − predSampleL0[hx − 1][vy]) >> shift1 (8-769)



- Change the order of subtraction and right shift

gradientHL0[x][y] = (predSamplesL0[hx + 1][vy]>>shift1) − (predSampleL0[hx − 1][vy]>> shift1) (8-769)



- In case of 128-bit register, it can be improved from 4-parallel to 8-parallel.

SHARP
Be Original.

- The derivation of gradient value of PROF is the same as that of current BDOF.

- However, it is implemented with 16-bit operation.

```
__m128i mmGradVer = _mm_sra_epi16(_mm_sub_epi16(mmPixBottom, mmPixTop),
                                  _mm_cvtsi32_si128(IDCC_PROF_GRAD_SHIFT));
__m128i mmGradHor = _mm_sra_epi16(_mm_sub_epi16(mmPixRight, mmPixLeft),
                                  _mm_cvtsi32_si128(IDCC_PROF_GRAD_SHIFT));
```

- Since the range of interpolated sample value is 16-bit, it is necessary to guarantee not to cause any overflow.

Changes of working draft

SHARP
Be Original.

- Change 4 lines.

8.5.6.4 Bidirectional optical flow prediction process

From

The variables gradientHL0[x][y], gradientVL0[x][y], gradientHL1[x][y] and gradientVL1[x][y] are derived as follows:

$gradientHL0[x][y] = (predSamplesL0[h_x + 1][v_y] - predSampleL0[h_x - 1][v_y]) >> shift1$ (8-769)
$gradientVL0[x][y] = (predSampleL0[h_x][v_y + 1] - predSampleL0[h_x][v_y - 1]) >> shift1$ (8-770)
$gradientHL1[x][y] = (predSamplesL1[h_x + 1][v_y] - predSampleL1[h_x - 1][v_y]) >> shift1$ (8-771)
$gradientVL1[x][y] = (predSampleL1[h_x][v_y + 1] - predSampleL1[h_x][v_y - 1]) >> shift1$ (8-772)

to

The variables gradientHL0[x][y], gradientVL0[x][y], gradientHL1[x][y] and gradientVL1[x][y] are derived as follows:

$gradientHL0[x][y] = (predSamplesL0[h_x + 1][v_y] >> shift1) - (predSampleL0[h_x - 1][v_y] >> shift1)$ (8-769)
$gradientVL0[x][y] = (predSampleL0[h_x][v_y + 1] >> shift1) - (predSampleL0[h_x][v_y - 1] >> shift1)$ (8-770)
$gradientHL1[x][y] = (predSamplesL1[h_x + 1][v_y] >> shift1) - (predSampleL1[h_x - 1][v_y] >> shift1)$ (8-771)
$gradientVL1[x][y] = (predSampleL1[h_x][v_y + 1] >> shift1) - (predSampleL1[h_x][v_y - 1] >> shift1)$ (8-772)

Experimental results (CTC)

SHARP
Be Original.

- The 1% encoding time has been improved.
- There was a little gain on average.

	Random Access Main 10				
	Over VTM-5.0				
	Y	U	V	EncT	DecT
Class A1	0.00%	-0.20%	-0.04%	99%	100%
Class A2	-0.01%	-0.05%	-0.02%	100%	100%
Class B	0.00%	0.02%	0.02%	99%	100%
Class C	-0.02%	-0.02%	-0.07%	99%	100%
Class E					
Overall	-0.01%	-0.05%	-0.02%	99%	100%
Class D	0.00%	0.00%	0.15%	99%	99%
Class F	0.00%	0.00%	-0.01%	99%	100%

Experimental results (Cross-check: JVET-O0982)

SHARP
Be Original.

- The coding efficiency results were identical to ours.
- The 1% decoding time has been improved.

	Random access Main10				
	Over VTM-5.0				
	Y	U	V	EncT	DecT
Class A1	0.00%	-0.20%	-0.04%	100%	100%
Class A2	-0.01%	-0.05%	-0.02%	99%	99%
Class B	0.00%	0.02%	0.02%	100%	100%
Class C	-0.02%	-0.02%	-0.07%	100%	99%
Class E					
Overall	-0.01%	-0.05%	-0.02%	100%	99%
Class D	0.00%	0.00%	0.15%	100%	97%
Class F	0.00%	0.00%	-0.01%	100%	99%

Conclusion

SHARP
Be Original.

- In this contribution, we proposed a speed up method for BDOF.
- As an experimental result the encoding times became 1% faster and the average gains of coding efficiency were -0.01%/-0.05%/-0.02% for Y/Cb/Cr respectively.
- Recommend to fix this change to WD and VTM software.
- Thank Unno-san (KDDI, JVET-O0982) for cross-checking.

SHARP

Be Original.

