AHG8: Independent Rice Parameter Derivation for high bit depth and high bit rate extensions

JVET-X0129

Yue Yu, Haoping Yu, Zhihuang Xie, Fan Wang, Luhang Xu, Dong Wang

History-based Rice parameter derivation

- A history-based Rice parameter derivation method JVET-V0106 was adopted for VVC v2;
- A HistValue is calculated to update locSumAbs to derive new Rice parameter;
- A counter/color, StatCoeff[3] is utilized and may be updated once per TU from the first abs_remainder[] or dec_abs_level[]

  StatCoeff[i] = (StatCoeff[i] +Floor(Log2(abs_remainder[]) + 2) >> 1

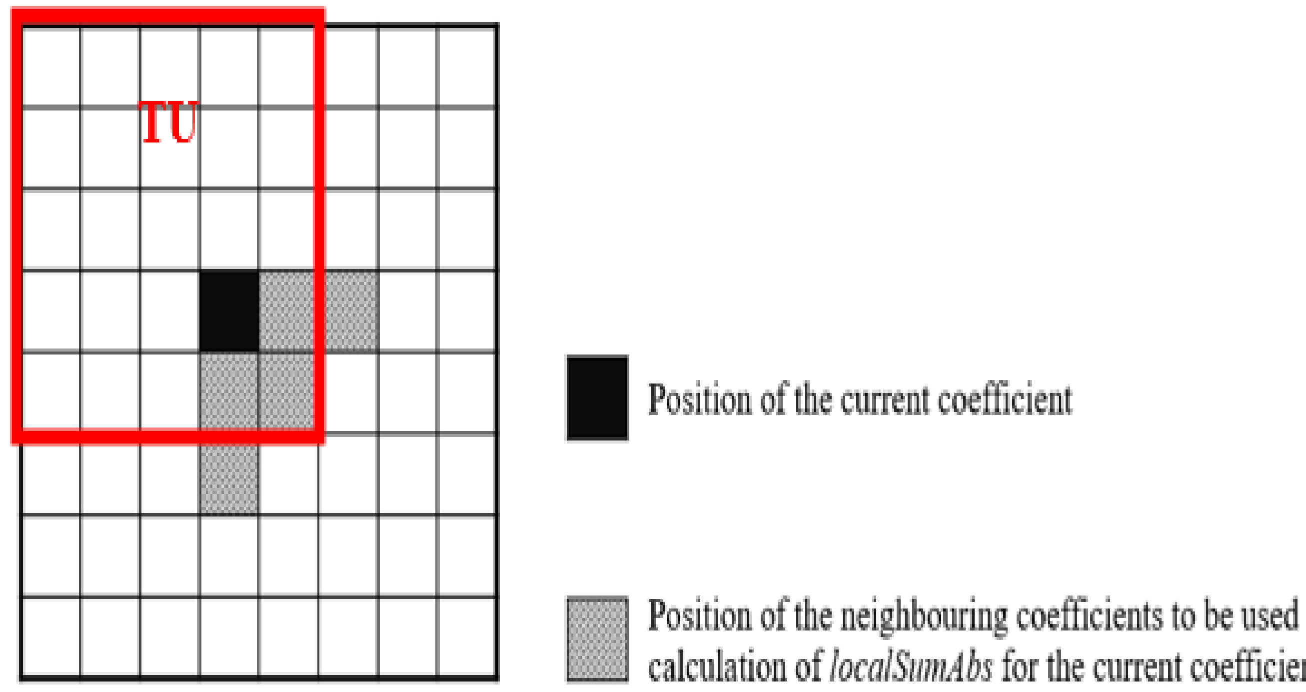
or

  StatCoeff[i] = (StatCoeff[i] +Floor(Log2(abs_remainder[])) >> 1

Prior to TU coding, HistValue = 1 << StatCoeff[i]

and StatCoeff[i] is initialed as 2* Floor(Log2(BitDepth - 10))

- There is TU dependency to update StatCoeff[].




```
locSumAbs = 0
if( xC < ( 1 << log2TbWidth ) − 1 ) {
    locSumAbs += AbsLevel[ xC + 1 ][ yC ]
    if( xC < ( 1 << log2TbWidth ) − 2 )
        locSumAbs += AbsLevel[ xC + 2 ][ yC ]
    else
        locSumAbs += HistValue
    if( yC < ( 1 << log2TbHeight ) − 1 )
        locSumAbs += AbsLevel[ xC + 1 ][ yC + 1 ]
    else
        locSumAbs += HistValue
} else
    locSumAbs += 2 * HistValue
if( yC < ( 1 << log2TbHeight ) − 1 ) {
    locSumAbs += AbsLevel[ xC ][ yC + 1 ]
    if( yC < ( 1 << log2TbHeight ) − 2 )
        locSumAbs += AbsLevel[ xC ][ yC + 2 ]
    else
        locSumAbs += HistValue
} else
    locSumAbs += HistValue
```

The Proposed Method

- An initial value of StatCoeff_init[] is specified at the slice header.

[illegible]	[illegible]
if(sps_reverse_last_sig_coeff_enabled_flag)	
sh_reverse_last_sig_coeff_flag	u(1)
if(sps_persistent_rice_adaptation_enabled_flag){	
for(i = 0; i < 3; i++)	
statCoeff_init[i]	u(3)
}	
if(pps_slice_header_extension_present_flag) {	
sh_slice_header_extension_length	ue(v)

- Prior to the coding of each TU, HistValue = 1<<StatCoeff_init is used to derive the Rice parameter for coding the abs_remainder[] of the last significant if it does exist.
- Once last significant coefficient is coded, HistValue is updated with the actual level.

lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TbWidth + log2TbHeight − (log2SbW + log2SbH))) − 1	
HistValue = sps_persistent_rice_adaptation_enabled_flag ? (1 << StatCoeff_init[cIdx]) : 0	
updateHist = sps_persistent_rice_adaptation_enabled_flag ? 1 : 0	
…	
for(n = firstPosMode0; n > firstPosMode1; n− −) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(abs_level_gtx_flag[n][1]) {	
abs_remainder[n]	ae(v)
~~if(updateHist && abs_remainder[n] > 0) {~~	
~~StatCoeff[cIdx] = (StatCoeff[cIdx] + Floor(Log2(abs_remainder[n])) + 2) >> 1~~	
~~updateHist = 0~~	
~~}~~	
}	
AbsLevel[xC][yC] = AbsLevelPass1[xC][yC] +2 * abs_remainder[n]	
if(updateHist){	
HistValue = AbsLevel[xC][yC]	
updateHist = 0	
}	
}	
for(n = firstPosMode1; n >= 0; n− −) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(sb_coded_flag[xS][yS]) {	
dec_abs_level[n]	ae(v)
~~if(updateHist && dec_abs_level[n] > 0) {~~	
~~StatCoeff[cIdx] = (StatCoeff[cIdx] + Floor(Log2(dec_abs_level[n]))) >> 1~~	
~~updateHist = 0~~	
~~}~~	
}	
if(AbsLevel[xC][yC] > 0) {	

Simulation Results

- Simulation conditions follow the CTC for high bit-depth coding;
- Over VTM-14.0, low QP, lossy, statCoeff_init[idx] = Clip(1, 8, (int) ((19 - QP) / 6))-1

HDR PQ	AI							
	Over VTM14.0							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	#VALUE!	#VALUE!	#VALUE!	0.07%	0.04%	0.03%	99%	100%
PQ422	#VALUE!	#VALUE!	#VALUE!	0.09%	0.04%	0.04%	99%	100%
Overall	#VALUE!	#VALUE!	#VALUE!	0.08%	0.04%	0.04%	99%	100%
	LDB							
	Over VTM14.0							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	#VALUE!	#VALUE!	#VALUE!	0.00%	-0.02%	0.02%	100%	99%
PQ422	#VALUE!	#VALUE!	#VALUE!	0.02%	0.02%	0.01%	100%	99%
Overall	#VALUE!	#VALUE!	#VALUE!	0.01%	0.00%	0.02%	100%	99%
	RA							
	Over VTM14.0							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	0.02%	0.02%	0.02%	0.02%	0.02%	0.02%	99%	100%
PQ422	0.03%	0.03%	0.01%	0.03%	0.03%	0.01%	100%	100%
Overall	0.02%	0.03%	0.01%	0.02%	0.03%	0.01%	100%	100%
Overall PQ	#VALUE!	#VALUE!	#VALUE!	0.04%	0.02%	0.02%	100%	100%

HDR HLG	AI				
	Over VTM14.0				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	0.09%	0.04%	0.04%	99%	100%
HLG422	0.09%	0.04%	0.03%	99%	100%
Overall	0.09%	0.04%	0.03%	99%	100%
	LDB				
	Over VTM14.0				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	0.00%	-0.01%	-0.01%	100%	100%
HLG422	0.02%	0.00%	0.02%	99%	100%
Overall	0.01%	0.00%	0.01%	100%	100%
	RA				
	Over VTM14.0				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	0.01%	0.01%	0.01%	100%	100%
HLG422	0.02%	0.02%	0.03%	100%	99%
Overall	0.02%	0.02%	0.02%	100%	100%
Overall HLG	0.04%	0.02%	0.02%	99%	100%

SVT RGB	AI				
	Over VTM14.0				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	0.21%	0.16%	0.16%	98%	98%
SVT12	0.09%	0.03%	0.03%	100%	100%
Overall	0.15%	0.10%	0.10%	99%	99%
	LDB				
	Over VTM14.0				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	0.11%	0.11%	0.11%	100%	100%
SVT12	0.03%	0.03%	0.03%	99%	100%
Overall	0.07%	0.07%	0.07%	99%	100%
	RA				
	Over VTM14.0				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	0.10%	0.11%	0.11%	99%	100%
SVT12	0.03%	0.02%	0.02%	99%	100%
Overall	0.06%	0.06%	0.06%	99%	100%
Overall RGB	0.09%	0.08%	0.08%	99%	100%

- # Over VTM-14.0, lossless

PQ	All Intra			Low delay B			Random Access		
	ratio		bit-rate savings	ratio		bit-rate savings	ratio		bit-rate savings
	VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012	
PQ444	2.6	2.6	0.59%	3.1	3.1	0.46%	3.1	3.1	0.44%
PQ422	2.4	2.4	0.50%	2.9	2.9	0.49%	2.9	2.9	0.46%
Overall	**2.5**	**2.5**	**0.55%**	**3.0**	**3.0**	**0.48%**	**3.0**	**3.0**	**0.45%**
Enc Time[%]	96%			105%			94%		
Dec Time[%]	82%			104%			84%		

HLG	All Intra			Low delay B			Random Access		
	ratio		bit-rate savings	ratio		bit-rate savings	ratio		bit-rate savings
	VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012	
HLG444	1.8	1.8	0.60%	2.0	2.0	0.48%	2.0	2.0	0.47%
HLG422	1.7	1.7	0.45%	1.9	1.9	0.54%	1.9	1.9	0.52%
Overall	**1.7**	**1.7**	**0.53%**	**2.0**	**1.9**	**0.51%**	**2.0**	**1.9**	**0.50%**
Enc Time[%]	97%			105%			94%		
Dec Time[%]	78%			99%			82%		

SVT	All Intra			Low delay B			Random Access		
	ratio		bit-rate savings	ratio		bit-rate savings	ratio		bit-rate savings
	VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012		VTM-14.0	/TM-14.0+X012	
SVT16	1.2	1.2	0.84%	1.2	1.2	0.42%	1.2	1.2	0.41%
SVT12	1.3	1.3	0.41%	1.3	1.3	0.20%	1.3	1.3	0.19%
Overall	**1.2**	**1.2**	**0.63%**	**1.3**	**1.3**	**0.31%**	**1.3**	**1.3**	**0.30%**
Enc Time[%]	94%			93%			96%		
Dec Time[%]	81%			77%			88%		

Over VTM-14.0 without PersistentRiceAdaption (new anchor)

HDR PQ	AI							
	Over VTM14.0 - PersistentRiceAdapt							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	#VALUE!	#VALUE!	#VALUE!	-0.12%	-0.07%	-0.09%	105%	104%
PQ422	#VALUE!	#VALUE!	#VALUE!	-0.13%	-0.09%	-0.10%	100%	98%
Overall	#VALUE!	#VALUE!	#VALUE!	-0.13%	-0.08%	-0.10%	102%	101%

	LDB							
	Over VTM14.0 - PersistentRiceAdapt							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	#VALUE!	#VALUE!	#VALUE!	-0.06%	-0.07%	-0.05%	105%	117%
PQ422	#VALUE!	#VALUE!	#VALUE!	-0.07%	-0.05%	-0.09%	103%	111%
Overall	#VALUE!	#VALUE!	#VALUE!	-0.06%	-0.06%	-0.07%	104%	114%

	RA							
	Over VTM14.0 - PersistentRiceAdapt							
	wPsnrY	wPsnrU	wPsnrV	psnrY	psnrU	psnrV	EncT	DecT
PQ444	-0.05%	-0.04%	-0.05%	-0.06%	-0.04%	-0.05%	102%	103%
PQ422	-0.06%	-0.04%	-0.08%	-0.06%	-0.04%	-0.08%	102%	104%
Overall	-0.06%	-0.04%	-0.06%	-0.06%	-0.04%	-0.06%	102%	103%

Overall PQ	#VALUE!	#VALUE!	#VALUE!	-0.08%	-0.06%	-0.08%	103%	106%

HDR HLG	AI				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	-0.26%	-0.13%	-0.13%	101%	101%
HLG422	-0.30%	-0.18%	-0.19%	99%	101%
Overall	-0.28%	-0.15%	-0.16%	100%	101%

	LDB				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	-0.04%	-0.04%	-0.05%	110%	117%
HLG422	-0.03%	-0.03%	-0.04%	108%	118%
Overall	-0.04%	-0.03%	-0.04%	109%	117%

	RA				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrY	psnrU	psnrV	EncT	DecT
HLG444	-0.04%	-0.02%	-0.03%	106%	116%
HLG422	-0.04%	-0.04%	-0.03%	106%	119%
Overall	-0.04%	-0.03%	-0.03%	106%	118%

Overall HLG	-0.12%	-0.07%	-0.08%	105%	112%

SVT RGB	AI				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	-1.48%	-1.21%	-1.19%	109%	128%
SVT12	-0.71%	-0.43%	-0.43%	105%	106%
Overall	-1.09%	-0.82%	-0.81%	107%	117%

	LDB				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	-0.97%	-1.03%	-1.03%	129%	129%
SVT12	-0.32%	-0.31%	-0.31%	120%	130%
Overall	-0.65%	-0.67%	-0.67%	124%	129%

	RA				
	Over VTM14.0 - PersistentRiceAdapt				
	psnrG	psnrB	psnrR	EncT	DecT
SVT16	-0.96%	-0.93%	-0.93%	131%	130%
SVT12	-0.31%	-0.27%	-0.27%	119%	129%
Overall	-0.63%	-0.60%	-0.60%	125%	129%

Overall RGB	-0.79%	-0.70%	-0.69%	119%	125%

Conclusion

- Proposes an independent Rice parameter derivation method, where the current TU dependent StatCoeff is totally removed.
- Minor loss over VTM-14.0.
- Gain over VTM-14.0 without original TU dependent persistent Rice adaption.
- Suggest adopting this independent Rice parameter derivation.

Thank Qualcomm for crosschecking!

Thank you