m49627:

[G-PCC] EE13.3 related proposal on lossless attributes coding using Integer Haar Transform harmonized with RAHT

Panasonic

Introduction

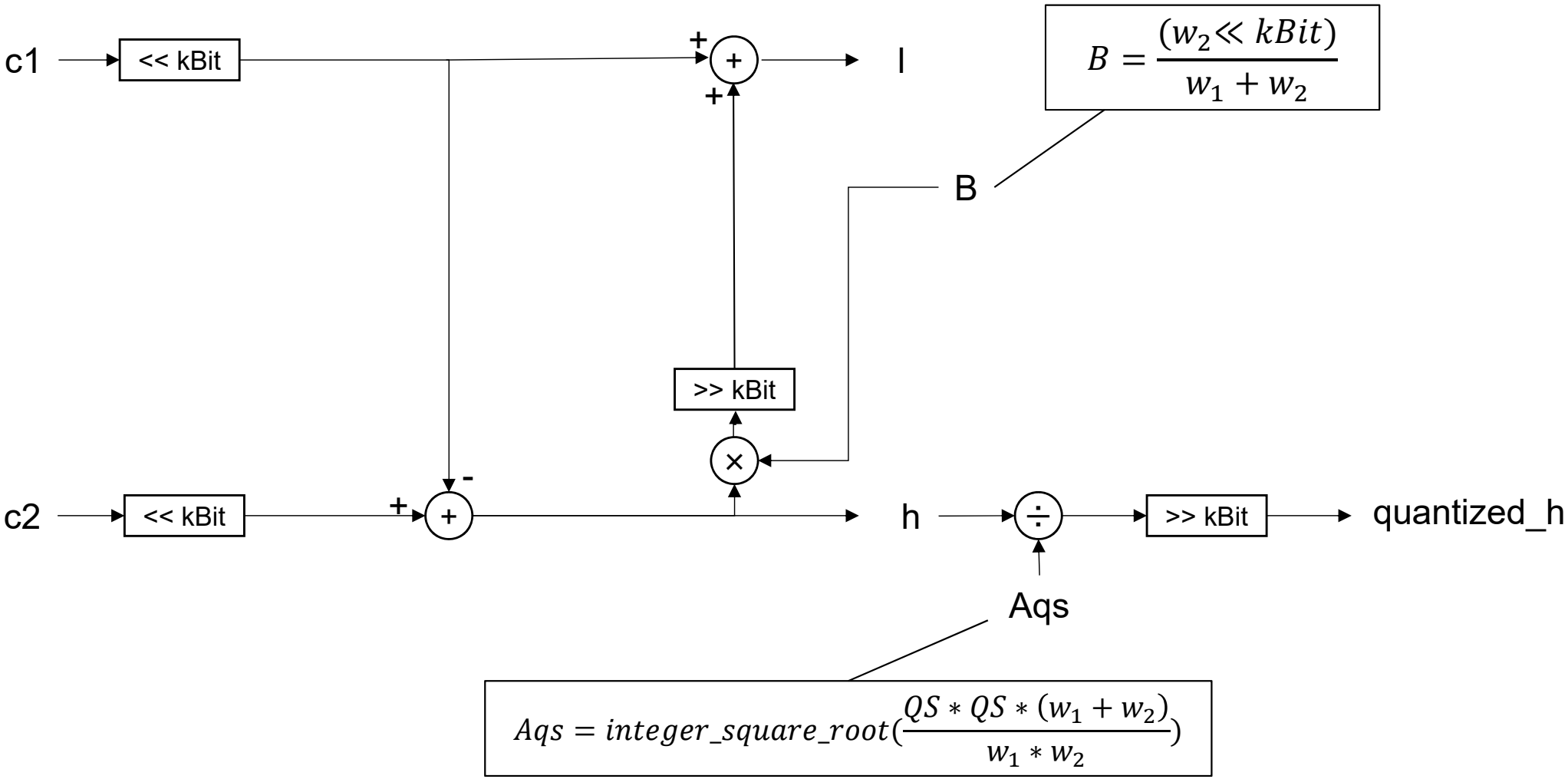
- RAHT is an efficient and effective method for Attributes coding of 3D point cloud
- Although Fixed Point RAHT has been introduced to facilitate hardware implementation, Lossless RAHT cannot be achieved by it
- In this proposal, Integer Haar Transform is proposed to realize lossless attributes coding in G-PCC
- It can be integrated to Fixed Point RAHT with minimal modifications

Fixed Point RAHT

Forward transform

$$h = c_2 - c_1$$

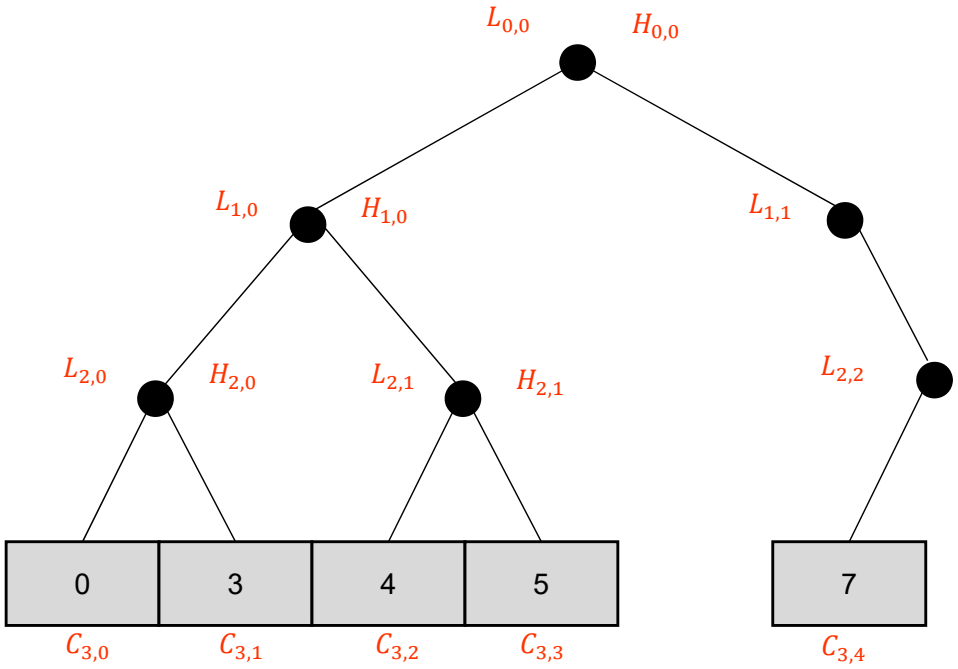
$$l = c_1 + ((B * h + kHalf) \gg kBit)$$



lossless compression cannot be realized due to square-root and division operations

Integer Haar Transform

- RAHT transform process includes square root operators and division. The information is lost and not possible to realize the lossless compression.
  → We propose Integer Haar transform to obtain lossless compression



Unnormalized Haar Transform

$$\begin{bmatrix} L_{l,m} \\ H_{l,m} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -1 & 1 \end{bmatrix} \begin{bmatrix} C_{l+1,2m} \\ C_{l+1,2m+1} \end{bmatrix}$$

Lowpass sub-band at level l:

$$L_{l,m} = (C_{l+1,2m} + C_{l+1,2m+1})/2$$

Highpass sub-band at level l:

$$H_{l,m} = C_{l+1,2m+1} - C_{l+1,2m}$$

The highpass sub-band will be entropy encoded. The lowpass sub-band will be passed to the next level $C_{l,m} = L_{l,m}$

The hierarchical transform of RAHT is preserved and the transform coefficients are fixed.

Integer Haar Transform (cont.)

The unnormalized Haar Transform can be rewritten as

$$H_{l,m} = C_{l+1,2m+1} - C_{l+1,2m}$$

$$L_{l,m} = \frac{C_{l+1,2m} + C_{l+1,2m+1}}{2} = C_{l+1,2m} + \frac{H_{l,m}}{2}$$

The Integer Transform can be achieved by:

Integer Haar Transform

$$H_{l,m} = C_{l+1,2m+1} - C_{l+1,2m}$$

$$L_{l,m} = C_{l+1,2m} + \lfloor H_{l,m}/2 \rfloor$$

Integer Inverse Haar Transform

$$C_{l+1,2m} = L_{l,m} - \lfloor H_{l,m}/2 \rfloor$$

$$C_{l+1,2m+1} = H_{l,m} + C_{l+1,2m}$$

where $\lfloor \quad \rfloor$ is *floor* operation.

Efficient Implementation:

Define $C_{i,j}$ as an integer type.
Therefore, $H_{i,j}$ and $L_{i,j}$ are also integers.

The efficient implementation of the transform can be realized by

$$H_{l,m} = C_{l+1,2m+1} - C_{l+1,2m}$$

$$L_{l,m} = C_{l+1,2m} + (H_{l,m} \gg 1)$$

and

$$C_{l+1,2m} = L_{l,m} - (H_{l,m} \gg 1)$$

$$C_{l+1,2m+1} = H_{l,m} + C_{l+1,2m}$$

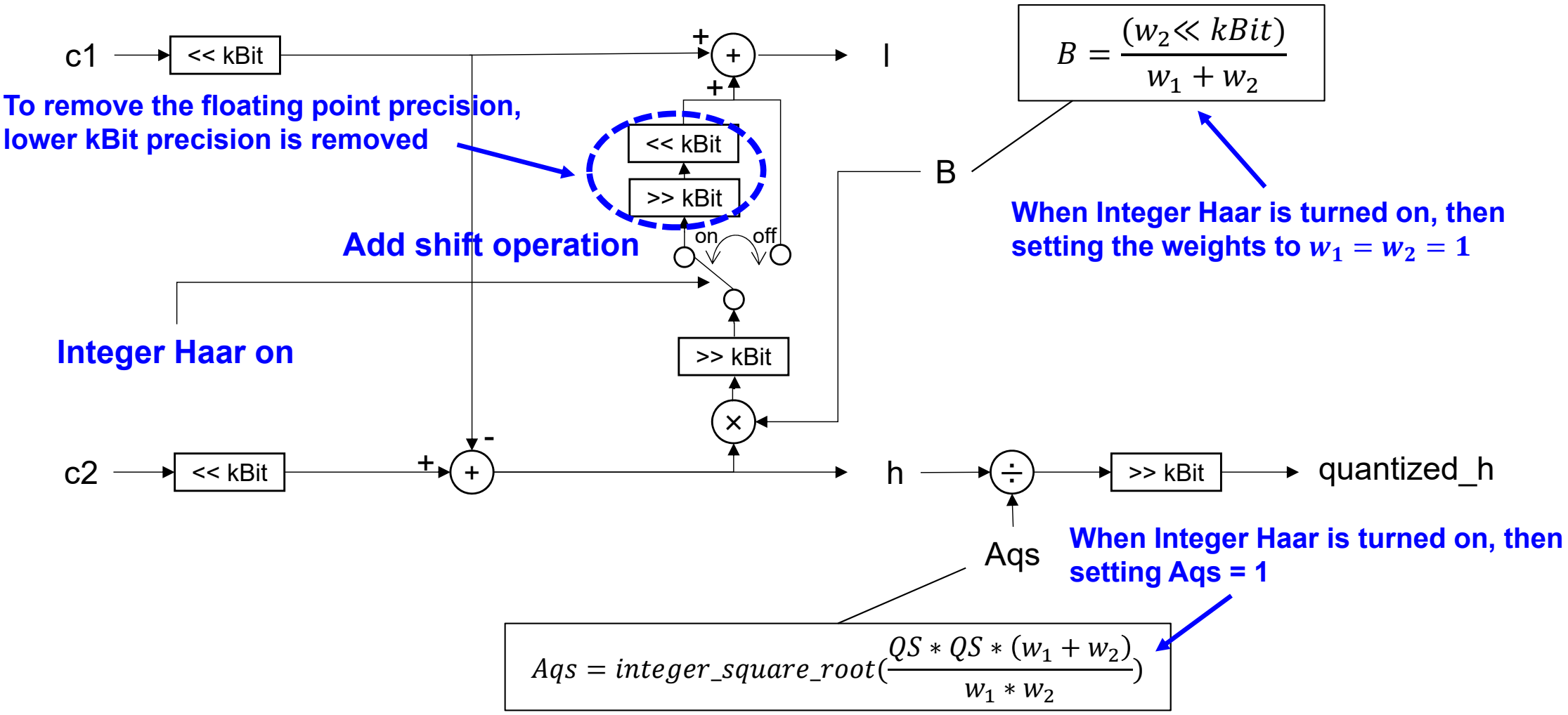
Notice that only 1 addition, 1 subtraction and 1 shift-left operations needed for the transform.

Integer Haar Transform on top of Fixed Point RAHT

Forward transform

$$h = c_2 - c_1$$

$$l = c_1 + (h \gg 1)$$



The modification is minimal

Test Results

- The proposed method is implemented in TMC13v6
- Test on CW setting in CTC is conducted and the results are shown below

Objective Results (CW) : Proposed method Compared with Lossless Predicting Transform

CW_ai	lossless geometry, lossless attributes [all intra] bpip ratio [%]			
	Geometry	Colour	Reflectance	Total
Cat1-A average	100.0%	112.6%		109.0%
Cat1-B average	100.0%	#VALUE!		100.0%
Cat3-fused average	100.0%	110.1%	104.8%	104.4%
Cat3-frame average	100.0%		106.3%	100.9%
Overall average	100.0%	#VALUE!	105.7%	103.2%
Avg. Enc Time [%]	71%			
Avg. Dec Time [%]	64%			

Overall average of 3.2% loss with 30% speed advantage

Additional Test Results

- Evaluation with similar coding time was required
- To do it, SearchRange of Predicting Transform was set to 2

Objective Results (CW) : Proposed method Compared with Predicting Transform + SearchRange=2

CW_ai	lossless geometry, lossless attributes [all intra] bpip ratio [%]			
	Geometry	Colour	Reflectance	Total
Cat1-A average	100.0%	102.9%		102.1%
Cat1-B average	100.0%	#VALUE!		100.0%
Cat3-fused average	100.0%	101.9%	98.7%	100.6%
Cat3-frame average	100.0%		100.3%	100.1%
Overall average	100.0%	#VALUE!	99.7%	100.7%
Avg. Enc Time [%]	101%			
Avg. Dec Time [%]	97%			

Some loss for color while some gain for reflectance

Conclusion

- We proposed lossless attributes coding using Integer Haar Transform
- It can be easily integrated with fixed point RAHT
- It achieves about 30% speed-up with 3.2% coding loss compared to lossless Predicting Transform
- We recommend the proposed approach be evaluated in next CE activity