

TE12: Evaluation of transform coefficient coding (HHI_TRANSFORM_CODING) with tool breakdown

(JCTVC-C059/m18082)

Vivienne Sze, Madhukar Budagavi

Texas Instruments Inc.

**Joint Collaborative Team on Video Coding (JCT-VC)
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**

3rd Meeting: Guangzhou, CN, 7-15 October, 2010

HHI_TRANSFORM_CODING

- Proposed in JCTVC – A116 (HHI Fraunhofer)
- HHI_TRANSFORM_CODING is a context modeling scheme that aims to improve coding efficiency using three tools:

1. Adaptive scanning order of significance map

- Every diagonal scan can switch direction **depending** on location of significant coefficients

2. Proposed context modeling for significance map

- Context selection **depends** on neighboring coefficients

3. Proposed context modeling for coefficient levels

- Context selection **depends** on number of non-zero coefficient in previous sub-block of the same macroblock

HHI_TRANSFORM_CODING introduces dependencies into Context Modeling which is already difficult to parallelize.

Cross-Verification

- TMuC-0.7 under common conditions (JCTVC-B300)
- Simulation platform is LSF equipped with Intel(R) Xeon(R) CPU X5570@2.93GHz 64 bits Linux machines

Default setting	Alternative setting	Parameters for Alternative setting
HHI_TRANSFORM_CODING On	HHI_TRANSFORM_CODING Off	#define HHI_TRANSFORM_CODING 0

- Coding Efficiency Results (BD-Rate) for High Efficiency Test

Intra	Random Access	Low Delay
1.4	1.3	2.0

- HHI_TRANSFORM_CODING has a 1.3 to 2.0% coding **gain** compared to when it is disabled (Samsung proposal)
- Results have been verified to match those obtained from Samsung

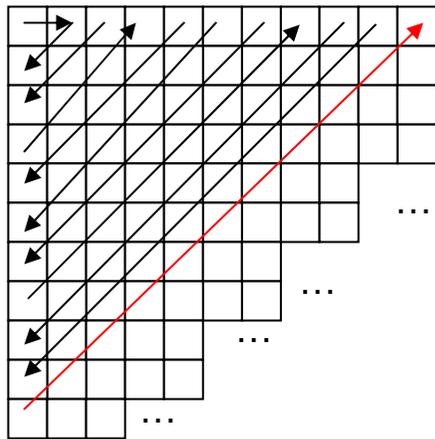
Context Selection For Significance Map

- In Samsung proposal, (i.e. HHI_TRANSFORM_CODING disabled), the context used for significant_coeff_flag in every position is known. Thus, context selection can be done in parallel.

1	1	1	1	1	1
1	1	1	1	1	...
1	1	1	1		
1	1	1			
1	1				
1	:				

Adaptive Scan

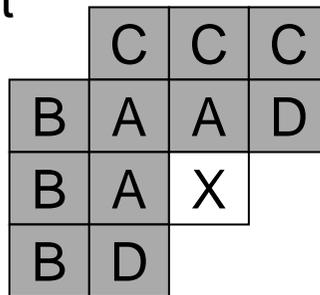
- Direction of the diagonal scan depends on the location of significant coefficients of the previous diagonal scan
 - bottom-left or top-right
- Each diagonal scan can be in any direction; thus scans must be done serially
- *Within* a given scan, the context selection of significant_coeff_flag can still be done in parallel.



1	2	3	4	5	6
2	3	4	5	6	...
3	4	5	6		
4	5	6			
5	6				
6	:				

Proposed context modeling for significance map

- For >8x8 transforms, context of significant_coeff_flag depends on up to **11 neighbors**.
- When used with zig-zag scan or adaptive scan, the context selection for the majority of significant_coeff_flag elements cannot be done in parallel (especially at the decoder)
- Context selection of last_significant_coeff_flag also modified to depend on distance from top-right



Context for significant_coeff_flag in position X depends on surrounding positions highlighted in grey

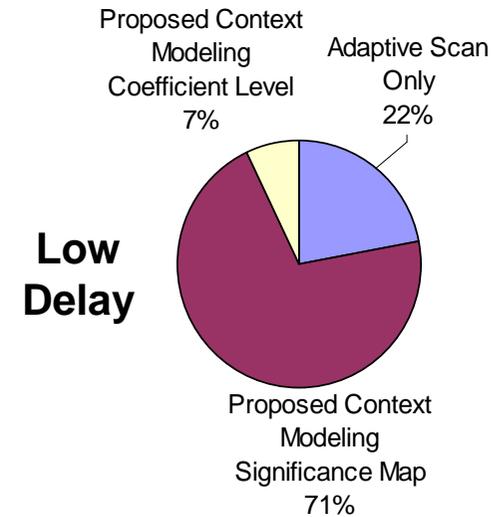
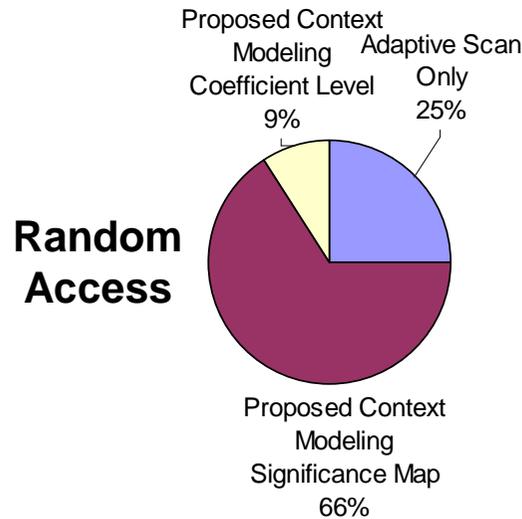
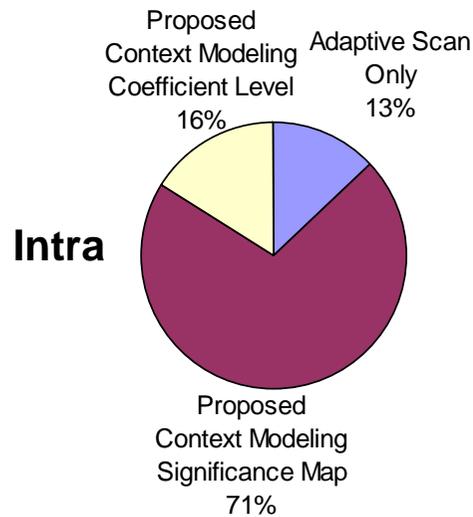
Proposed context modeling for coefficient level

- Different contexts can be used for each 4x4 sub-block to encode `coeff_abs_level_minus_one` depending on the statistics of already coded sub-blocks
 - number of coefficients whose $|\cdot| > 1$ in the previous sub-block within the same macroblock.
- Introduces dependencies between sub-blocks, which may limit parallelism in context modeling.

Due to the increased complexity in context selection, it is important to understand the coding gains of each tool.

Coding Efficiency Breakdown

Tool Under Test (TMuC-0.7.3)	Average Gain (BD-rate)
HHI_TRANSFORM_CODING (all three tools)	1.3 to 2.0 %
Adaptive Scan	0.2 to 0.4%
Proposed context selection significance map	0.9 to 1.4 %
Proposed context selection coefficient level	0.1 to 0.2 %



Majority of the coding efficiency gain due to proposed context modeling for significance map

Conclusions and Recommendations

- Coding gain of HHI_TRANSFORM_CODING was verified to be 1.3 to 2.0%
- Majority of gains (0.9 to 1.4%) due to proposed context selection for significance map. As such, we recommend the following be investigated
 - Breakdown of coding gains between significant_coeff_flag and last_significant_coeff_flag. The former introduces much more complexity than the latter
 - Complexity reduction of approach
- Other two tools have a poor trade-off between coding gain and complexity;
 - Increases complexity in context modeling which is already difficult to parallelize