Video Coding Technology Proposal by Texas Instruments Inc. (JCTVC-A101.doc)

Madhukar Budagavi *, Vivienne Sze **, Mehmet Umut Demircin *

Salih Dikbas *, Minhua Zhou *, Anantha P. Chandrakasan **

*** Texas Instruments Inc.**

**** Massachusetts Institute of Technology**

Joint Collaborative Team on Video Coding (JCT-VC)
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11
1st Meeting: Dresden, DE, 15-23 April, 2010

Overview - I

- Memory access bandwidth and video throughput are key limiting factors in video codec chip design
- Therefore in addition to coding efficiency improvements, memory access bandwidth reduction and enhanced parallelism to increase throughput are important for new video coding standard
- Three video coding tools proposed:
  - Orthogonal Mode Dependent Directional Transform (OMDDT)
    (**Low Implementation Complexity:** Simplified spatial transform)
  - Massively Parallel CABAC (MP-CABAC)
    (**High Throughput Operation:** Entropy coding parallelization)
  - Compressed Reference Frame Buffer (CRFB)
    (**Low Power Operation:** Memory access bandwidth reduction)

Overview - II

- Codec submission built on top of KTA2.6r1.
- Other tools enabled:
  - Extended macroblock sizes (64x64), Enhanced AIF, Motion vector competition, Mode dependent directional transform, High precision H.264 filter, Adaptive loop filter
    - `UseAdaptiveFilter        = 5  # Enhanced AIF`
    - `MVCompetition            = 1  # Enabled with default parameters`
    - `UseIntraMDDT             = 1  # Use MDDT for intra blocks`
    - `UseHPFilter              = 1  # Use High Precision H.264 filter`
    - `UseAdaptiveLoopFilter    = 1  # Use adaptive loop filtering`
    - `UseExtMB                 = 2  # Use extended block size (64x64)`
- VCEG-AJ10 common conditions used in encoder

Orthogonal Mode Dependent Directional Transform (OMDDT)

- **Objective:** Simplify Mode Dependent Directional Transform of KTA2.6r1
- MDDT uses one transform per Intra prediction direction
  - Transform matrices designed using KLT
  - Transform given by $Y = B_i X A_i^T$, i = Intra prediction direction
  - Uses 2 transform matrices per Intra prediction direction
    - Total of 44 transform matrices (18 for 4x4, 18 for 8x8, 8 for 16x16)
- Desirable to reduce MDDT complexity for reducing implementation cost, while preserving coding efficiency

OMDDT Proposal

- Orthogonal MDDT (OMDDT)
  - Transform given by $Y = S_i X S_i^T$, i = Intra prediction direction
  - Transform designed using KLT using autocorrelation matrix of X
- Uses one transform matrix per Intra prediction direction versus two transform matrices used by original MDDT
  - Need to store only half the number transform matrices compared to MDDT
- On some hardwired architectures, OMDDT uses about half the area of the original MDDT since hardwired matrix multiplication logic of the first matrix multiplication can be reused for the second matrix multiplication in the transform
  - $XS_i^T = (S_i X^T)^T$

OMDDT Results

- Memory savings

	Total storage for transform matrices (bytes)	
	MDDT	OMDDT
16x16	2*4*16*16 = 2048	4*16*16 = 1024
8x8	2*9*8*8 = 1152	9*8*8 = 576
4x4	2*9*4*4 = 288	9*4*4 = 144
Total	3488	1744

- Rate-distortion performance of individual tool compared to KTA2.6r1

	Proposal_HierB - KTA_HierB					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class A	-0.01	-0.01	-0.01	0.23	0.37	0.12
Class B	-0.01	0.00	-0.01	0.27	0.44	0.18
Class C	-0.01	0.00	-0.02	0.18	0.44	-0.04
Class D	0.00	0.00	-0.01	0.07	0.16	0.00
Overall	-0.01	0.00	-0.02	0.19	0.44	-0.04

	Proposal_IPPP - KTA_IPPP					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	0.00	0.00	-0.01	0.07	0.25	-0.05
Class C	0.00	0.01	-0.01	0.01	0.14	-0.19
Class D	0.00	0.01	-0.01	0.09	0.41	-0.11
Class E	-0.01	0.00	-0.01	0.33	0.47	0.09
Overall	0.00	0.01	-0.01	0.11	0.47	-0.19

Massively Parallel CABAC (MP-CABAC) - I

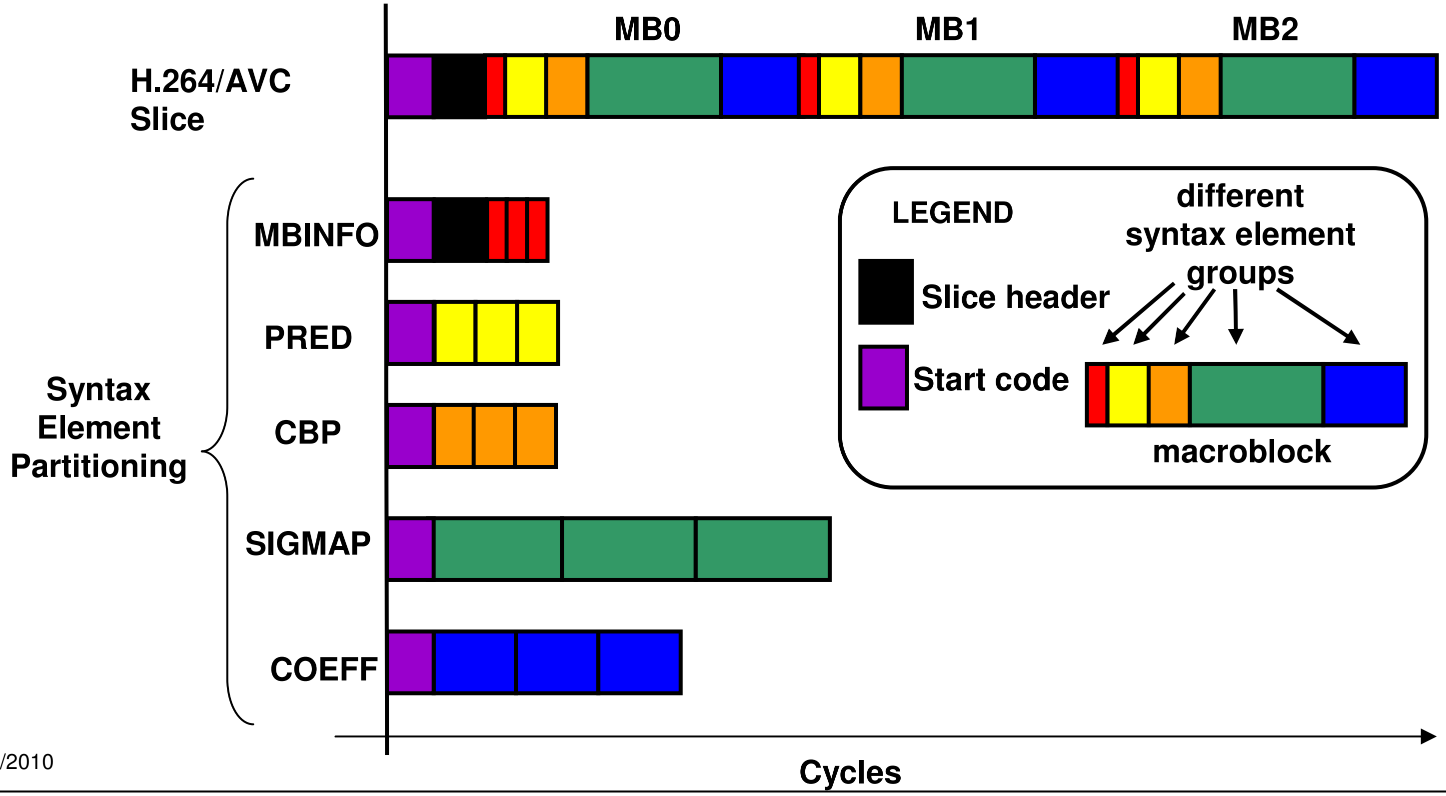
- CABAC key bottleneck in AVC
- **Objective:** Increase throughput of CABAC entropy coding with parallel processing.
  - Increased performance
    - High frame rate/resolution
    - Low latency
  - Reduced power consumption (Low Voltage)
- **Design Goals:**
  - Improve trade-off between
    - Coding Penalty vs. Effective Throughput
    - Area Cost vs. Effective Throughput
  - Low complexity hardware/software implementation

Massively Parallel CABAC (MP-CABAC) - II

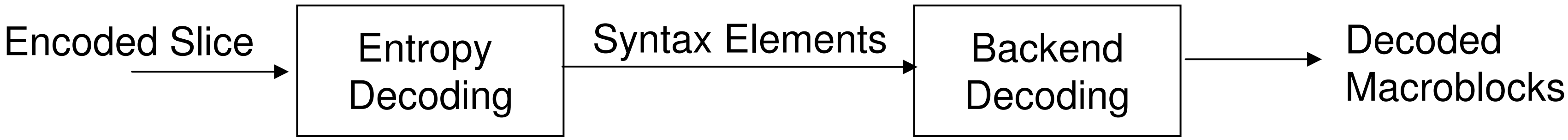
- Combine **three levels of parallelism**
  1) Binary symbol (bin) [COM16-C334]
  2) Syntax Element
  3) Macroblock/Slice
- Adopted into KTA (KTA2.7)

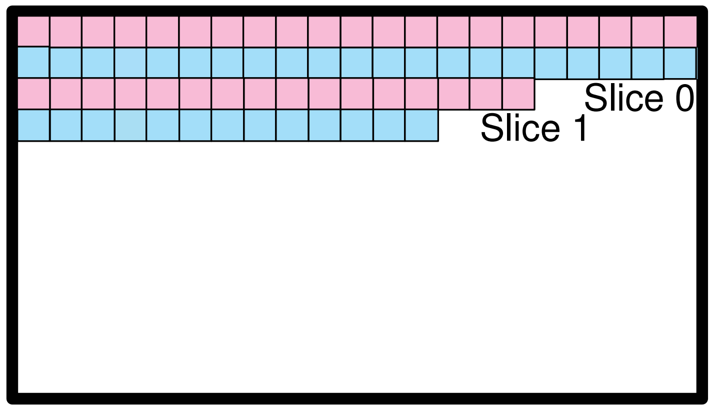
Syntax Element Partitions

- Place syntax elements in different partitions (demarcated with start code)
- Process partitions in parallel
- Allocation of syntax elements to partitions based on distribution (balance workload)



Interleaved Entropy Slices (IES)

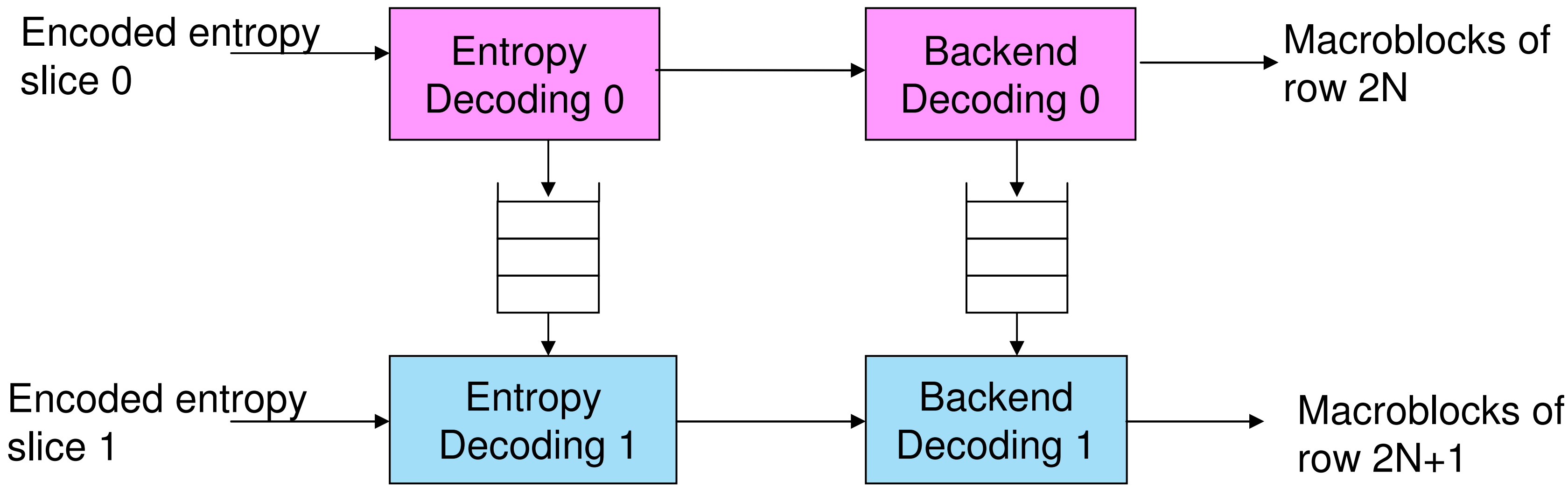




- Interleaved entropy slices satisfies backend dependencies (no buffering required)
- Processing done in raster scan order within slice
  - Favorable memory access, caching and low latency

IES Decoding Architecture

- Simple synchronization between engines with FIFOs
- Enable full CODEC parallelism



Massively Parallel CABAC (MP-CABAC) results

- Throughput increase for Class A, B and E video sequences in CfP submission
  - 2.78x to 4.49x for Constraint Set 1 (CS1)
  - 1.68x to 4.81x for Constraint Set 2 (CS2)
  - More parallelism possible by increasing number of IES

- Rate-distortion performance of individual tool compared to KTA2.6r1

	Proposal_HierB - KTA_HierB					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class A	-0.02	-0.01	-0.03	0.41	0.58	0.23
Class B	-0.01	-0.01	-0.02	0.60	0.91	0.39
Overall	-0.02	-0.01	-0.03	0.54	0.91	0.23

	Proposal_IPPP - KTA_IPPP					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	-0.02	-0.01	-0.03	0.63	0.97	0.33
Class E	-0.04	-0.03	-0.07	1.35	1.90	0.79
Overall	-0.03	-0.01	-0.07	0.90	1.90	0.33

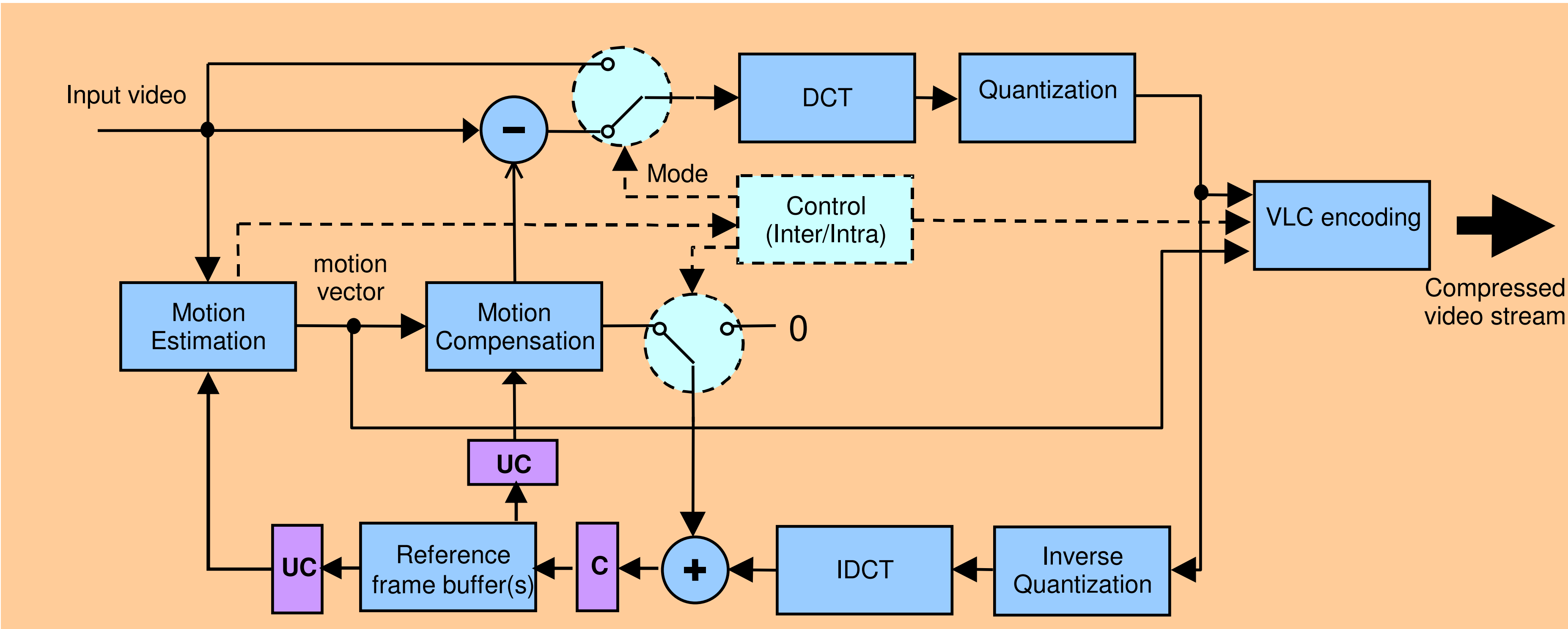
- *Throughput increase has been validated in hardware*

Compressed Reference Frame Buffer (CRFB)

- **Objective:** Reduce the ME/MC memory access bandwidth by half
  - Total system memory bandwidth limited because of cost and power.
  - Memory accesses are power hungry.
  - Limited memory bandwidth impacts video quality because of constraints on amount of data that can be loaded for ME.

- **Design Constraints:**
  - Guaranteed 2x compression ratio to enable random access to memory blocks (a row of 64x64 MBs).
  - Implemented in core video coding loop to avoid drift error caused by lossy compression.
  - Based on simple compression and decompression algorithms.
  - Enabled only on high-resolutions. (> 720p)

CRFB in Encoder Block Diagram



C → Compress reference frame buffer

UC → Decompress reference frame buffer

CRFB Algorithm - I

- Based on simple transform, quantization, DC prediction and entropy coding
- Transform:
  - 2D integer S-transform on 8x8 luma and 4x4 chroma blocks with 3 lifting steps.

$$\text{Forward}: \begin{cases} y[2n+1] = x[2n+1] - x[2n] \\ y[2n] = x[2n] + \left\lfloor \dfrac{y[2n+1]}{2} \right\rfloor \end{cases}$$

$$\text{Inverse}: \begin{cases} x[2n] = y[2n] - \left\lfloor \dfrac{y[2n+1]}{2} \right\rfloor \\ x[2n+1] = y[2n+1] + x[2n] \end{cases}$$

- Quantization:
  - Midtread quantizer operates on non-DC transform coefficients.
  - Quantization step sizes are powers of 2. (12 quantization indices)

CRFB Algorithm - II

- DC prediction:
  - DC coefficients are predicted using left, top, and top-left neighbor blocks.
  - Similar to MPEG-4.

- Entropy coding:
  - EG0 and EG3 codes used to encode quantized transform coefficients and DC prediction residual.
  - Quantization index is unary coded.

- Rate control:
  - Leaky-bucket like simple rate control to guarantee the fixed compression ratio for a memory access block.

CRFB Results

- Achieves 50% reduction in reference frame buffer memory access
- Reference frame buffer memory is half of that without compression
- Rate-distortion performance of individual tool compared to KTA2.6r1

	Proposal_HierB - KTA_HierB					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class A	0.00	0.00	0.00	0.00	0.00	-0.01
Class B	0.00	0.00	-0.01	0.21	1.24	0.00
Overall	0.00	0.00	-0.01	0.15	1.24	-0.01

	Proposal_IPPP - KTA_IPPP					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	0.00	0.00	-0.01	0.05	0.34	0.00
Class E	0.00	0.00	0.00	0.00	0.00	0.00
Overall	0.00	0.00	-0.01	0.03	0.34	0.00

Overall codec compression performance - I

- Comparison to AVC Alpha, Beta, and Gamma anchor

	Proposal - Alpha_anchor					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class A	1.16	1.33	1.02	-24.38	-18.27	-30.46
Class B	0.97	1.43	0.66	-30.33	-19.57	-40.28
Class C	1.32	1.56	1.13	-28.10	-24.97	-30.27
Class D	1.10	1.52	0.87	-23.07	-15.61	-35.95
Overall	1.12	1.56	0.66	-27.01	-15.61	-40.28

	Proposal - Beta_anchor					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	0.82	1.57	0.28	-23.48	-7.68	-34.63
Class C	0.62	1.07	0.01	-13.91	-0.19	-20.64
Class D	0.13	0.71	-0.47	-0.74	15.70	-14.85
Class E	1.14	1.52	0.82	-25.70	-20.36	-30.03
Overall	0.66	1.57	-0.47	-15.82	15.70	-34.63

	Proposal - Gamma_anchor					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	1.80	2.55	1.25	-43.95	-29.43	-55.44
Class C	1.63	2.22	1.20	-34.10	-26.69	-39.51
Class D	1.27	1.67	0.97	-27.59	-17.08	-44.01
Class E	2.32	2.71	1.97	-44.90	-40.22	-47.19
Overall	1.72	2.71	0.97	-37.58	-17.08	-55.44

Overall codec compression performance - II

- Comparison to KTA2.6r1 HierB and IPPP under VCEG-AJ10 common conditions

	Proposal_HierB - KTA_HierB					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class A	-0.02	-0.01	-0.02	0.50	0.67	0.31
Class B	-0.02	-0.02	-0.03	1.03	2.06	0.66
Class C	-0.01	0.00	-0.02	0.18	0.44	-0.04
Class D	0.00	0.00	-0.01	0.07	0.16	0.00
Overall	-0.01	0.00	-0.03	0.48	2.06	-0.04

	Proposal_IPPP - KTA_IPPP					
	BD-PSNR			BD-Rate		
	Avg	Max	Min	Avg	Max	Min
Class B	-0.02	-0.01	-0.03	0.79	1.21	0.36
Class C	0.00	0.01	-0.01	0.01	0.14	-0.19
Class D	0.00	0.01	-0.01	0.09	0.41	-0.11
Class E	-0.05	-0.04	-0.07	1.65	2.06	1.15
Overall	-0.02	0.01	-0.07	0.58	2.06	-0.19

Complexity Discussions - I

- OMDDT: Uses half the number of transform matrices compared to OMDDT
- MP-CABAC Syntax element partitioning
  - Entire CABAC does NOT have to be replicated
    - Finite state machine used for context selection, and the context memory are not replicated
  - Only the arithmetic coding engine is replicated
    - 10 to 15% of CABAC area
    - Only 0.5x increase in CABAC area for 5 partitions
- MP-CABAC Interleaved entropy slices
  - Hardware area of the entire CABAC (including the context memory) must be replicated for IES
  - Area increases linearly with amount of parallelism
  - Advantages over AVC slices
    - cross slice context selection
    - simple synchronization
    - reduction in memory bandwidth
    - low latency
    - improved workload balance

Complexity Discussions - II

- Compressed reference frame buffer (CRFB)

Table 43 – CRFB encoder complexity

Algorithm	**Number of Arithmetic Operations per Pixel**
Transform	- (1+1/4+1/16) *(2 additions and 1 shift) per Luma pixel - (1+1/4)*(2 additions and 1 shift) per Chroma pixel
Quantization	- 1 addition and 1 shift per pixel
Entropy coding	- 1 EG coding per pixel (additions and shifts depend on image statistics)
Rate Control	- 2 EG coding length computation to select Q_{idx}(12 candidates) and k=0,3 (2 candidates) per pixel. Worst case: log(12)*2. - 2 additions per pixel to accumulate the bit size of a CRFB block. Worst case: log(12)*2.

Table 44 – CRFB decoder complexity

Algorithm	**Number of Arithmetic Operations per Pixel**
Inverse Transform	- (1+1/4+1/16) *(2 additions and 1 shift) per Luma pixel - (1+1/4)*(2 additions and 1 shift) per Chroma pixel
Inverse Quantization	- 1 shift per pixel
Entropy decoding	- 1 EG decoding per pixel (additions and shifts depend on image statistics)

Conclusion

- Three video coding tools proposed:
  - Orthogonal Mode Dependent Directional Transform (OMDDT)
    (**Low Implementation Complexity:** Simplified spatial transform)
  - Massively Parallel CABAC (MP-CABAC)
    (**High Throughput Operation:** Entropy coding parallelization)
  - Compressed Reference Frame Buffer (CRFB)
    (**Low Power Operation:** Memory access bandwidth reduction)

- Recommend launching of core experiments for:
  - Intra transform design
    - Other related contributions to consider: Several
  - Parallelization of CABAC
    - Other related contributions to consider: [A105], [A116], …
  - Reference frame buffer compression
    - Other related contributions to consider: [A117]