Coding Sensitive based Approximation Algorithm for Power Efficient VBS-DCT VLSI Design in HEVC Hardwired Intra Encoder

Liangliang Chang, Zhenyu Liu, Xiangyang Ji,Dongsheng Wang
Tsinghua University, Beijing

Introduction

In this paper, according to the coding quality sensitive analysis, we approximate the DCT by decomposing the $\mathbf{R}$-matrix into several sparse butterfly-structure multiplications in series as In Ref. [1] , and further eliminate 25% computation in the row- and column-wise 1D transforms. The proposed algorithms outperform the counterpart In Ref. [2] in terms of coding quality, hardware-cost and power-cost saving.

Simplified RDO in HEVC Intra Coding

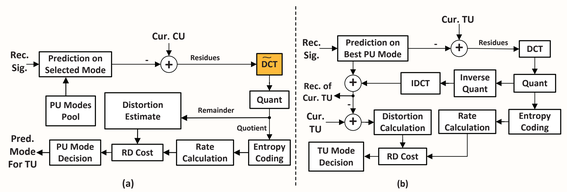

Figure 1: Simplified RDO Procedure for Optimal PU and TU Mode Decision ((a) PU mode decision procedure with low-cost DCT (b) TU mode decision procedure)

We proposed a low complex RDO for PU mode decision according to Ref. [2], where the pseudo DCT ($\widetilde{\mathrm{DCT}}$) substitutes the original DCT as shown in Fig. 1(a). During TU mode RDO, the original DCT must be adopted to avoid drifting problem, as shown by Fig. 1(b).

Coding Sensitive based Approximated DCT

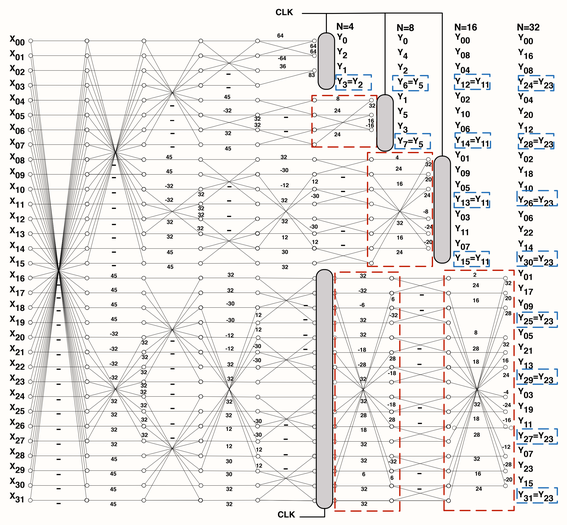

Figure 2: The whole structure of our proposed VLSI architecture for variable block size approximate DCT

In order to simplify the arithmetics in DCT, which consequently saves the hardware cost and power consumption, we propose a approximating algorithm to decompose the DCT matrix and devise the corresponding hardware architecture in this section.

DCT Matrix Decomposition

The original DCT matrix is decomposed as the multiplication of a butterfly structure and a block diagonal matrix in HEVC. In specific, the N × N DCT matrix is formulated as

$$\mathbf{A}_N = \mathbf{P}_N \begin{bmatrix} \mathbf{A}_{\frac{N}{2}} & \mathbf{O}_{\frac{N}{2}} \\ \mathbf{O}_{\frac{N}{2}} & \mathbf{R}_{\frac{N}{2}} \end{bmatrix} \mathbf{B}_N, \tag{1}$$

where $\mathbf{P}_N$ is a permutation matrix that changes the output data from its natural order to bit-reversed order. $\mathbf{B}_N$ represents the butterfly structure. $\mathbf{R}_{\frac{N}{2}}$ constitutes the information of even rows in $\mathbf{A}_N$.
We approximate DCT in HEVC PU modes decision stage. That is the approximated $\mathbf{R}_{N/2}$, i.e., $\tilde{\mathbf{R}}_{N/2}$ has the form of

$$\tilde{\mathbf{R}}_{\frac{N}{2}} = [45\mathrm{M}1] \cdot \mathrm{M}2 \cdot [32\mathrm{M}3] \cdot \mathrm{M}4 \cdot \ldots \cdot [32\mathrm{M}(2\log_2 \mathrm{N} - 3)], \tag{2}$$

where the operation $[\cdot]$ is rounding. We further simplified the multipliers in the last few stages, which are indicated by the red dash line blocks in Fig.2. The simplified counterparts are illustrated in Tab.1. This method is only used in the last stages to minimize errors like the principle of our scaling schemes. To keep the uniform amplitude after the pseudo DCT, additional right shift operations are required. Specifically, the outputs of $\tilde{\mathbf{R}}_{N/2}$ will be signed right shifted by $s_{N/2}$ bits. The values of $s_{N/2}$ are provided in Tab.2.

Table 1: Coefficients of original and simplified $[32\mathrm{M}5]$ in $\mathbf{R}_8$

Original	28	9	31	15	25	3
Simplified	24	8	32	16	24	2

Table 2: Definition of $s_{N/2}$

$\tilde{\mathbf{R}}_{N/2}$	$\mathbf{R}_4$	$\mathbf{R}_8$	$\mathbf{R}_{16}$
$s_{N/2}$	4	9	14

We employ the 2-stage pipeline structure to improve the maximum clock speed of our pseudo DCT hardwired engine.

High Frequency Coefficient Approximation

We divided the N × N row-wise transform coefficient matrix into 4 regions according to the column-major order. Namely, columns $[0, N/4-1]$ are in the region LL; columns $[N/4, N/2-1]$ construct the region LH; similarly, $[N/2, 3N/4-1]$ and $[N/2, 3N/4-1]$ are denoted as HL and HH, respectively.

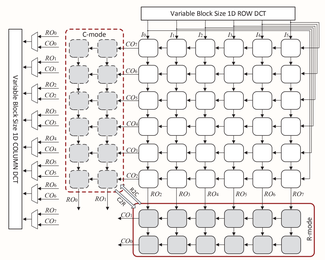

Figure 3: Hardware reusing transpose register file architecture (Each line ($\mathrm{I_i}$, $\mathrm{RO_i}$ and $\mathrm{CO_i}$) represents 4 pixels.)

Table 3: Coding quality comparisons after dropping 1/4 coefficients belonging to different frequency domains for 8 × 8, 16 × 16 and 32 × 32 DCT (BDBR: unit %)

Seq.	DCT 8×8				DCT 16×16				DCT 32×32			
	LL	LH	HL	HH	LL	LH	HL	HH	LL	LH	HL	HH
A	3.39	2.05	1.51	1.17	2.84	1.49	1.20	1.10	1.39	1.14	1.10	1.10
B	2.56	1.72	1.41	1.23	2.71	1.48	1.22	1.16	2.03	1.30	1.16	1.14
C	2.56	1.59	1.16	0.95	1.75	1.04	0.87	0.84	0.99	0.86	0.83	0.82
D	2.52	1.59	1.26	1.06	1.76	1.11	0.98	0.95	1.08	0.94	0.93	0.92
E	3.44	1.83	1.34	1.11	3.57	1.45	1.16	1.06	2.26	1.17	1.07	1.05
F	1.12	0.28	0.20	0.16	0.60	0.14	0.10	0.10	0.26	0.11	0.09	0.09
Ave.	2.60	1.51	1.14	0.95	2.20	1.12	0.92	0.87	1.33	0.92	0.86	0.85

The coding performance sensitivity to the coefficients in the above four regions is different, which is exhibited by the experiments shown in Tab.3. In our design, the coefficients of HH are approximated by the last column in region HL. This approximation scheme is explained in Fig.2. For example, the coefficients in 32 × 32 region HH, i.e., $\{\mathrm{Y_i} | \mathrm{i} \in [24, 32]\}$, are approximated by Y_{23}. As compared with the direct dropping method, our approximation can decrease BDBR by 0.30% in average.

Hardware Reusing Transpose Register File

Fig.3 shows our TRF hardware structure, and each square represents a basic unit that deals with 4 × 4 transposition. The gray squares are the reusable units. The overall capacity of TRF is 32 × 24 pixels. The registered data in TRF are configured to shift in horizontal or vertical, and these two configurations are interchanged with each other. The inputs of TRF come from row-wise DCT. When data are fed in from the top, in the first 24 cycles, the reusable units are configured in the C-mode and the shift direction is downward. In this period, $\mathrm{RO_i}$, where $\mathrm{i} \in [0:7]$, are dispatched to the column DCT unit. In the next 8 cycles, the reusable units are converted to R-mode that buffers the pixels from $\mathrm{RO}_2 \sim \mathrm{RO}_7$. In the following 32 cycles, $\mathrm{I_i}$ are dispatched from right that leads to the leftward shift. In the first 24 cycles of this period, the reusable units are in R-mode, and the column DCT uses $\mathrm{CO_i}$ as inputs; In the last 8 cycle of this period, the reusable units return to C-mode, and its function is to buffer the pixels from $\mathrm{CO_i}$.

Experiment and Conclusions

26 typical video sequences were tested with intra_main coding. Our algorithms obtained 15.9% time saving and outperformed the counterpart in coding quality. In specific, our approximated DCT algorithms only cause an averaging 1.03% BDBR increase as compared to the averaged 1.61% BDBR increase from Hadamard base algorithms. We further devised the VLSI implementation of the proposed approximate DCT algorithms.

Table 4: Hardware Implementation Performance Comparisons

Design	MaxSpeed [MHz]	HardwarCost [k gates] 1D-DCT	TRF	Total	Power [mW]	Cycle
Proposed	311	34.9	38.3	73.2	12.69	N+3/4N
[?]'s	418	15.9	59.6	75.5	14.21	2N
Primitive	311	199.7	60.4	260.1	42.64	2N

Our optimizations, including the matrix decomposition and the high frequency prediction, contribute to 82.5% hardware saving in 1D-DCT engines as compared to the primitive design while make the same maximum speed as the original DCT engine.In summary, our DCT engine could save 71.9% hardware cost and 70.2% power cost than the primitive design, and achieved the slightly better performance than the Hadamard counterpart.

Table 5: Coding Quality and Time Saving of Proposed RDO-based Intra Prediction Modes Decision Algorithms(Sequence A-C)

Class	Sequence	Hadamard[2] BP [dB]	BR [%]	Δ [%]	Proposed BP [dB]	BR [%]	Δ [%]
A	PeopleOnStreet	-0.113	2.24	17.1	-0.060	1.17	16.2
	Traffic	-0.102	2.10	15.0	-0.053	1.09	14.2
B	BasketballDrive	-0.056	2.21	17.1	-0.040	1.56	14.9
	BQTerrace	-0.087	1.67	17.4	-0.063	1.23	14.7
	Cactus	-0.066	1.94	17.2	-0.043	1.25	16.5
	Kimono	-0.069	2.23	17.2	-0.033	1.07	15.6
	ParkScene	-0.095	2.33	16.3	-0.057	1.39	14.8
	Tennis	-0.062	2.22	15.3	-0.036	1.28	17.9
C	BasketballDrill	-0.053	1.15	14.5	-0.038	0.81	16.5
	BasketballDrillText	-0.057	1.10	14.4	-0.040	0.78	16.4
	BQMall	-0.078	1.44	14.7	-0.054	0.99	14.6
	PartyScene	-0.089	1.27	14.5	-0.065	0.92	14.3
	RaceHorsesC	-0.098	1.63	14.8	-0.067	1.11	14.4

Table 6: Coding Quality and Time Saving of Proposed RDO-based Intra Prediction Modes Decision Algorithms(Sequence D-F)

Class	Sequence	Hadamard[2] BP [dB]	BR [%]	Δ [%]	Proposed BP [dB]	BR [%]	Δ [%]
D	BasketballPass	-0.087	1.56	15.2	-0.055	0.99	16.5
	BlowingBubbles	-0.090	1.61	16.3	-0.063	1.12	17.4
	BQSquare	-0.069	0.90	14.8	-0.060	0.78	15.8
	RaceHorses	-0.101	1.53	14.4	-0.065	1.06	14.9
	Keiba	-0.111	1.85	14.1	-0.070	1.16	14.2
E	Vidyo1	-0.094	2.11	14.9	-0.049	1.10	15.4
	Vidyo3	-0.071	1.45	18.2	-0.053	1.08	19.2
	Vidyo4	-0.080	1.94	16.4	-0.050	1.22	16.9
	Johnny	-0.075	1.96	15.4	-0.047	1.22	15.6
	KristenAndSara	-0.080	1.71	18.9	-0.052	1.10	19.2
F	SlideEditing	-0.080	0.59	16.5	-0.072	0.53	17.2
	ChinaSpeed	0.001	-0.03	15.7	0.020	-0.25	16.1
	SlideShow	-0.094	1.12	15.9	-0.093	1.11	16.2
Average		-0.079	1.61	15.9	-0.052	1.04	15.9

References

[1] W.H. Chen, CH Smith, and S. Fralick, A Fast Computational Algorithm For the Discrete Cosine Transform, *Communications, IEEE Transactions on*

[2] J. Zhu, Z. Liu, and D. Wang, Fast Prediction Mode Decision With Hadamard Transform Based Rate-Distortion Cost Estimation For HEVC Intra Coding, *Image Processing (ICIP), 2013 IEEE International Conference on*