

HDTV1080p HEVC Intra Encoder with Source Texture based CU/PU Mode Pre-decision

Jia Zhu, Zhenyu Liu, Dongsheng Wang

Speaker: Jia Zhu
Graduate Student of
Tsinghua Univ., Beijing, China

Jan-22, 2014

Outlines

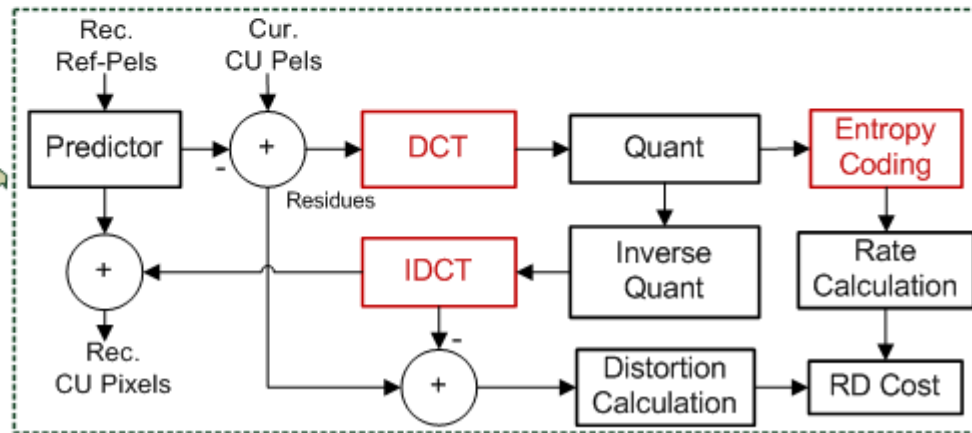
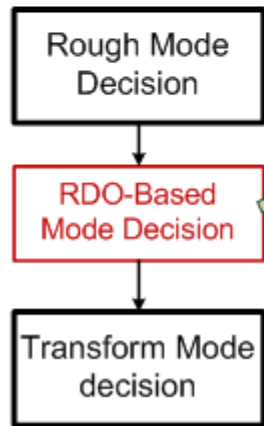
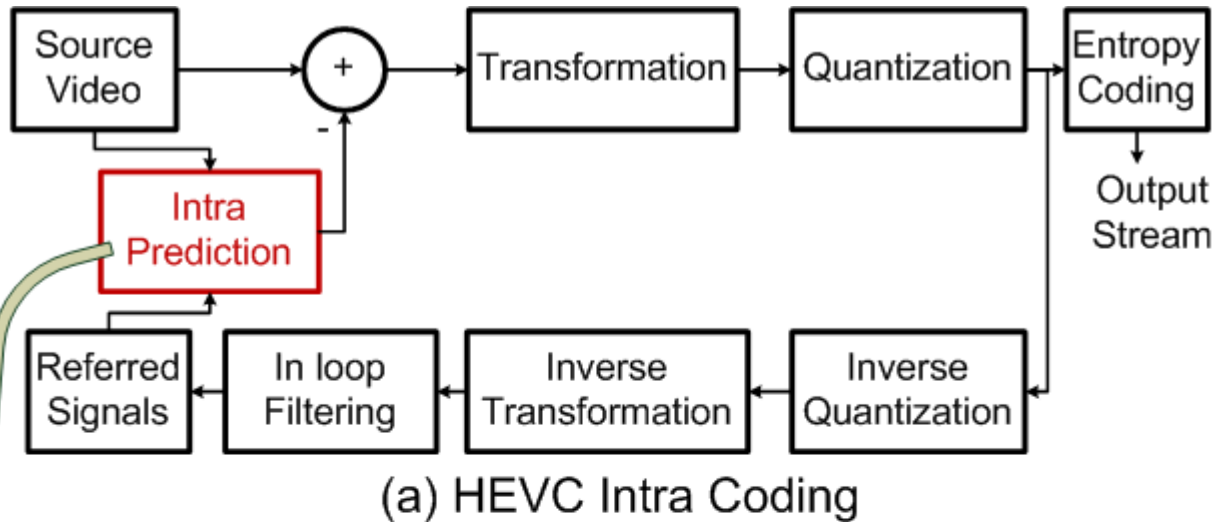
- Background Introduction
- Proposed Method
 - Predict Error Model
 - Pre CU Filtering
 - Intra Coding Process
- Hardware Design
- Experiment Result Soft. & Hard.
- Conclusion

HEVC Overview

- Emerging Video Coding Standard
 - developed by JCT-VC, finished in Jan-2013
 - aims to fulfill growing quality and resolution requirements
 - save 35-40% bit-rate cost compared to H.264/AVC
- Main Profile
 - quad-tree structure, CTU, luma and chroma
 - raise candidate parameter vector \vec{p} , including CU/PU/TU
 - RDO, $\vec{p}_0 = \arg \min_{\vec{p}} \{D(\vec{p}) + \lambda \cdot R(\vec{p})\}$
- Difficulties in Real-time Intra Coding
 - adopt large scale CTU, 64x64 against 16x16
 - employ 35 PU modes for more accuracy prediction
 - more \vec{p} , and explosive RD calculation.

HEVC Intra Coding

Time and hardware consuming, works could be done to move out redundancy.



(b) Intra Mode Decision

(c) Full RD-Cost Calculation

Existent Work in Fast Intra Coding

- Primary categories:
 - Low-complexity RD-cost algorithms
 - Filtering out most impossible modes
 - Early termination based on pre-defined thresholds
- Theory basis:
 - Simplify estimation in non-critical cases
 - Explore spatial/temporal correlation
 - Reduce candidate number via image textures
- Existed Problems:
 - early termination method is sensitive to worst cases
 - image textures only used in PU search, rarely in CU
 - serial process is not change, parallel design is hard

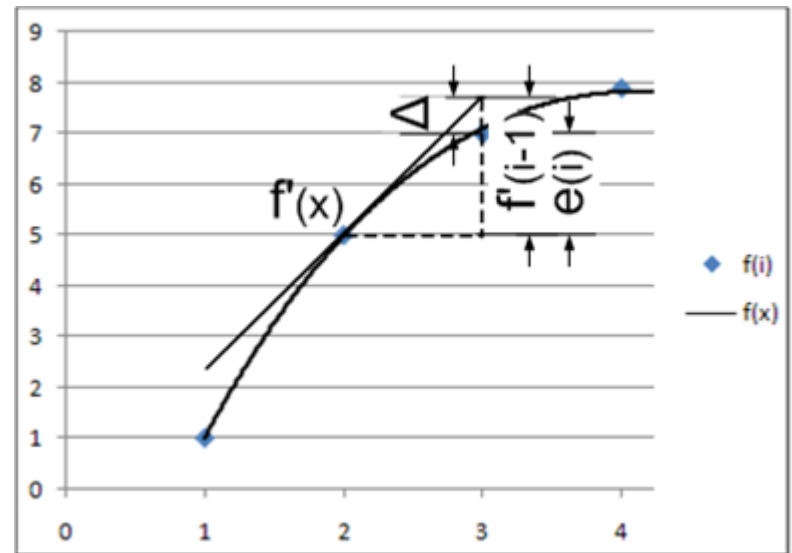
Original Idea

- Target
 - real-time, stable high throughput
 - low hardware cost
 - low coding effect loss
- Method
 - design parallel CU search engines
 - propose pre-CU mode filtering, 2 CU mode
 - base on source image texture, quantified by edge strength, introduce none extra delay in pipeline
 - embed other fast algorithms in each CU search engine

Texture based Predict Error Estimation

- Image Texture
 - obtained directly from source image
 - quantified as Edge Strength(ES)

- Estimate Predict Error
 - 1D example, $x, f(x), i, f(i)$
 - $f'(x)$ is derivation, $f'(i-1)$
 - use $f(i-1)$ to predict $f(i)$
 - $e(i)$ is the real PE value
 - $f'(i-1)$ is an estimation
 - then,
$$e(i) = f'(i-1) - \Delta$$
$$\approx f'(i-1)$$



Linear Predict Error Models in a CB

- Original Definition: $PE_k = (P_k - C_k)^2$
- Model Suppose: $\widetilde{PE}_k = \underset{\blacktriangle}{a} \cdot Qs^2 + \underset{\blacktriangle}{b} \cdot ES_k$
- Involved Items:
 - Qs is quantization step, ES_k is Edge Strength of k_{th} pixel
- Parameter Study:
 - Weighted least squares estimation:

$$\arg \min_{\{a, b_k\}} \left\{ \sum_{\tau=0}^{M-1} \left[\sum_{k=1}^{N^2-1} \omega_k \cdot \underline{(PE_k(\tau) - \widetilde{PE}_k(\tau))^2} + \omega_{N^2} \cdot \underline{\left(\sum_{k=1}^{N^2-1} PE_k(\tau) - \sum_{k=1}^{N^2-1} \widetilde{PE}_k(\tau) \right)^2} \right] \right\}$$

each pixel weight $\omega_k = 1$, sum. of all pixels weight $\omega_{N^2} = \frac{1}{N^2}$

PE Model Classification

- Histogram of ES
 - axis x, 33 angular mode
 - axis y, projected ES

- Prominent Direction

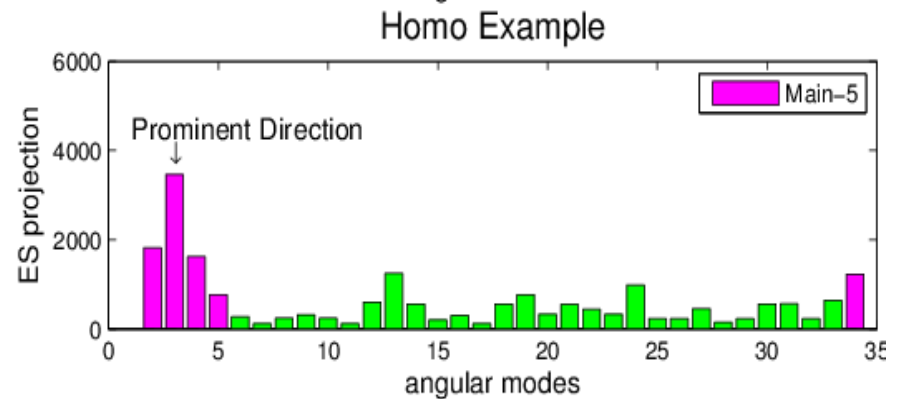
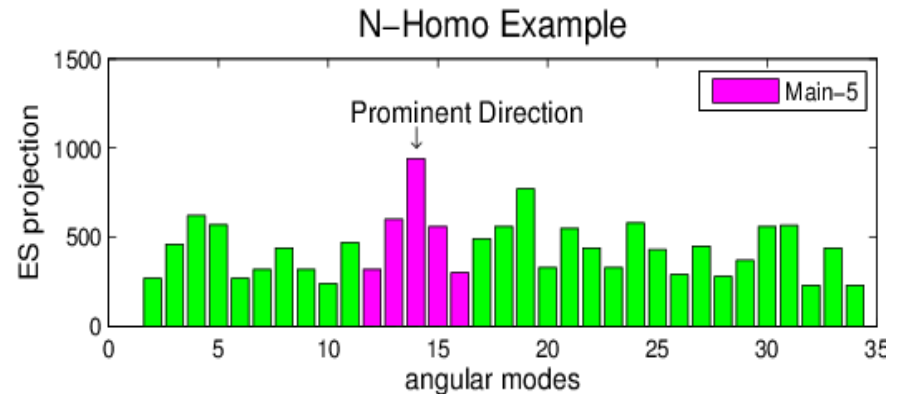
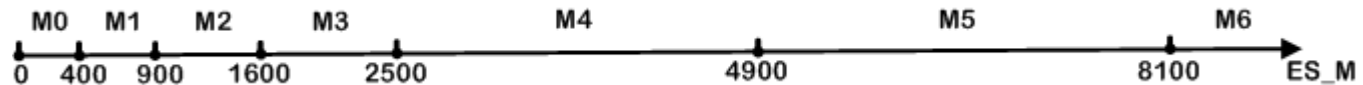
- 4 classes

- Directional Homogeneity

- main-neighbor-5 vs. all
 - 2 classes

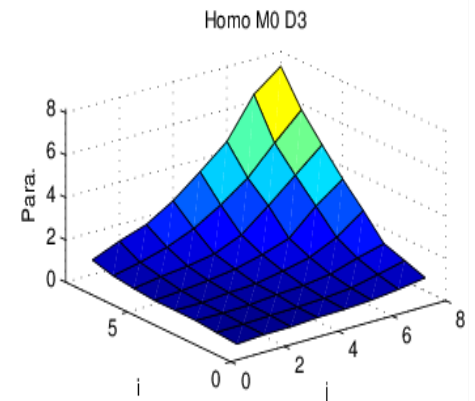
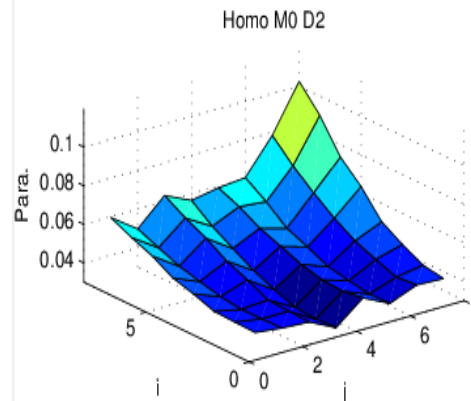
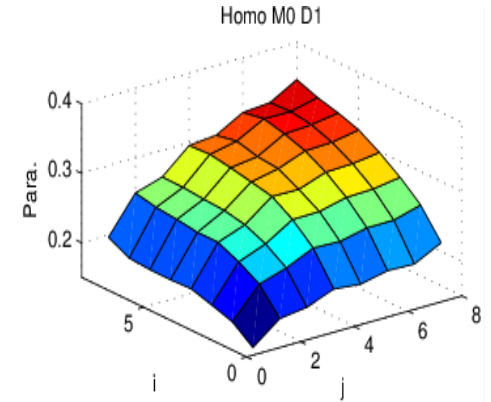
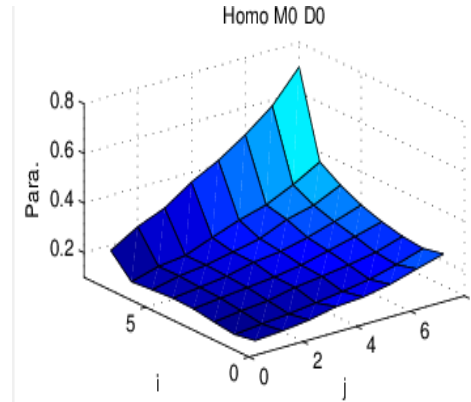
- ES Amplitude

- 7 classes



Model Para. Examples

- Prominent Direction
 - D0, 07-13, horizontal
 - D1, 14-22, -45 degree
 - D2, 23-29, vertical
 - D3, others, 45 degree
- Homogeneity
 - homo
- ES Amplitude
 - M0: $ES < 400$



PE based RD Cost Estimation

■ Pixel Rate&Distortion Estimation

$$\tilde{R}_k = \alpha \cdot \omega_r \cdot \widetilde{PE}_k \quad , \quad \tilde{D}_k = \omega_d \cdot \widetilde{PE}_k$$

- $\alpha = 7 / 64$, rate conversation factor
- ω_r, ω_d , weighting factors, theory and experience based

■ RD Cost of NxN Blocks

- N=4,8,16,32

$$RD_N = \sum_{k=0}^{N^2-1} (\tilde{R}_k + \tilde{D}_k)$$

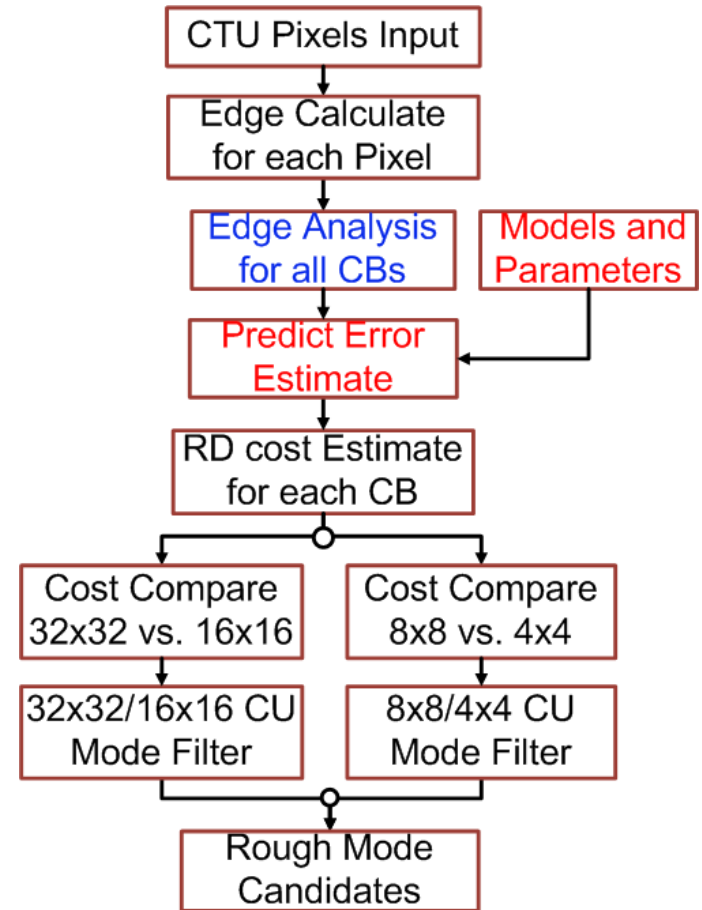
■ RD Cost of partitioned Block

$$RD_{\oplus N} = \sum_{n=0}^3 RD_{\frac{N}{2}}(n) + 3 \cdot \alpha \cdot (\gamma_{mode} + \gamma_{cbf})$$

- 4 sub-blocks' RD cost and merge cost
- $\gamma_{mode} = 4$, $\gamma_{cbf} = 1$, prediction mode bits and code-block-flag bit

Pre Modes Filtering

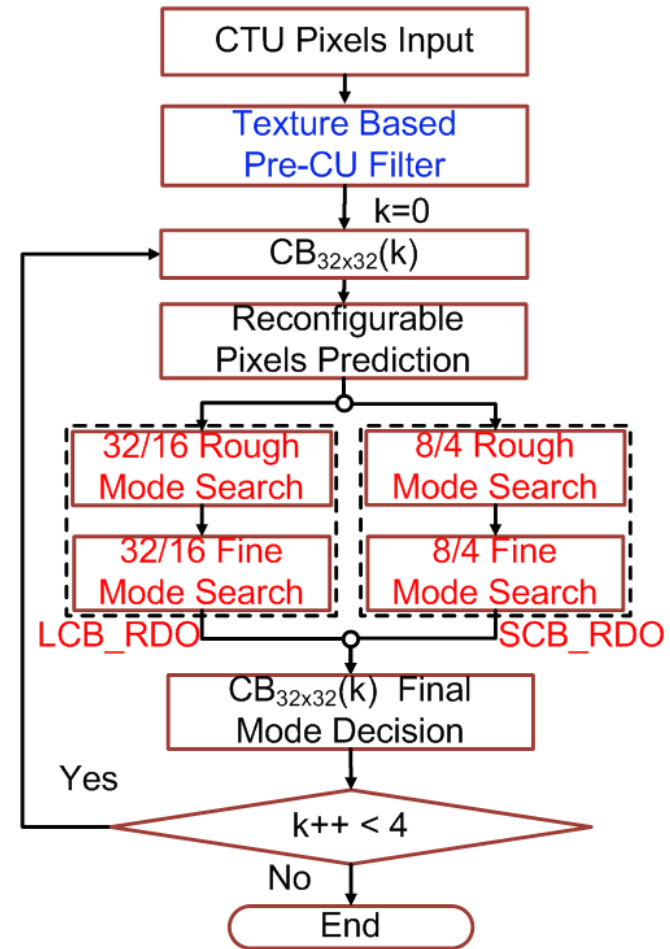
- Edge Calculation
 - 1 pixel cal. once, map to all CBs
- ES based Models Classify:
 - each CB is an unit, through EA
- Predict Error Estimate
 - corres. model and para.
- RD estimation
- Pre-CU Mode Filtering
 - LCB: 32x32 vs. 4 16x16
 - SCB: 8 x 8 vs. 4 4 x 4



Pre CU filtering flows

Panoramic View of Proposed Method

- CTU luma partition
 - 64x64 → 4 32x32 CBs
 - 64x64 is abandoned
- Pre-process:
 - CU mode filtering model
- Parallel PU search:
 - fast search applied
 - **LCB_RDO**: 32 / 16
 - **SCB_RDO**: 8 / 4
- Final Mode Decision.



Proposed RDO process

Pre-CU Filtering Structure

■ ES Analysis

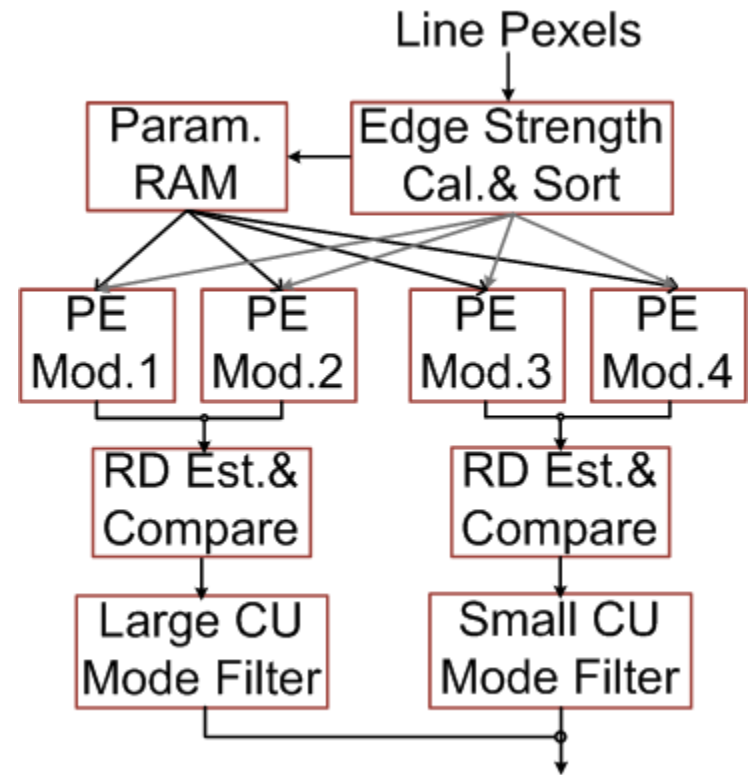
- pixels input row by row
- block finished->class info.
- send class info, save ES

■ PE Models & RD Estimation

- Mod.1,2,3,4: size 32,16,8,4
- row by row cal. & acc.

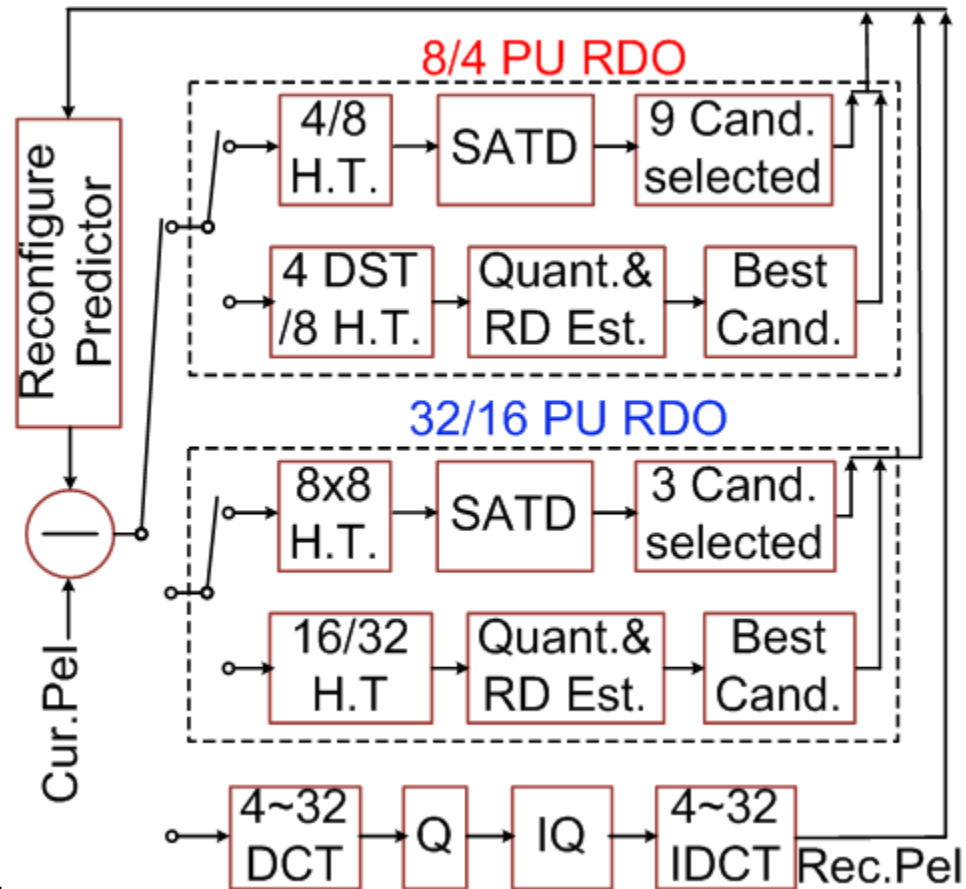
■ Mode Filter

- $RD_N, RD_{\oplus N}, N \in \{8, 32\}$
- mode reserved cases: $\{32, 8\}, \{16, 8\}, \{32, 4\}, \{16, 4\}$

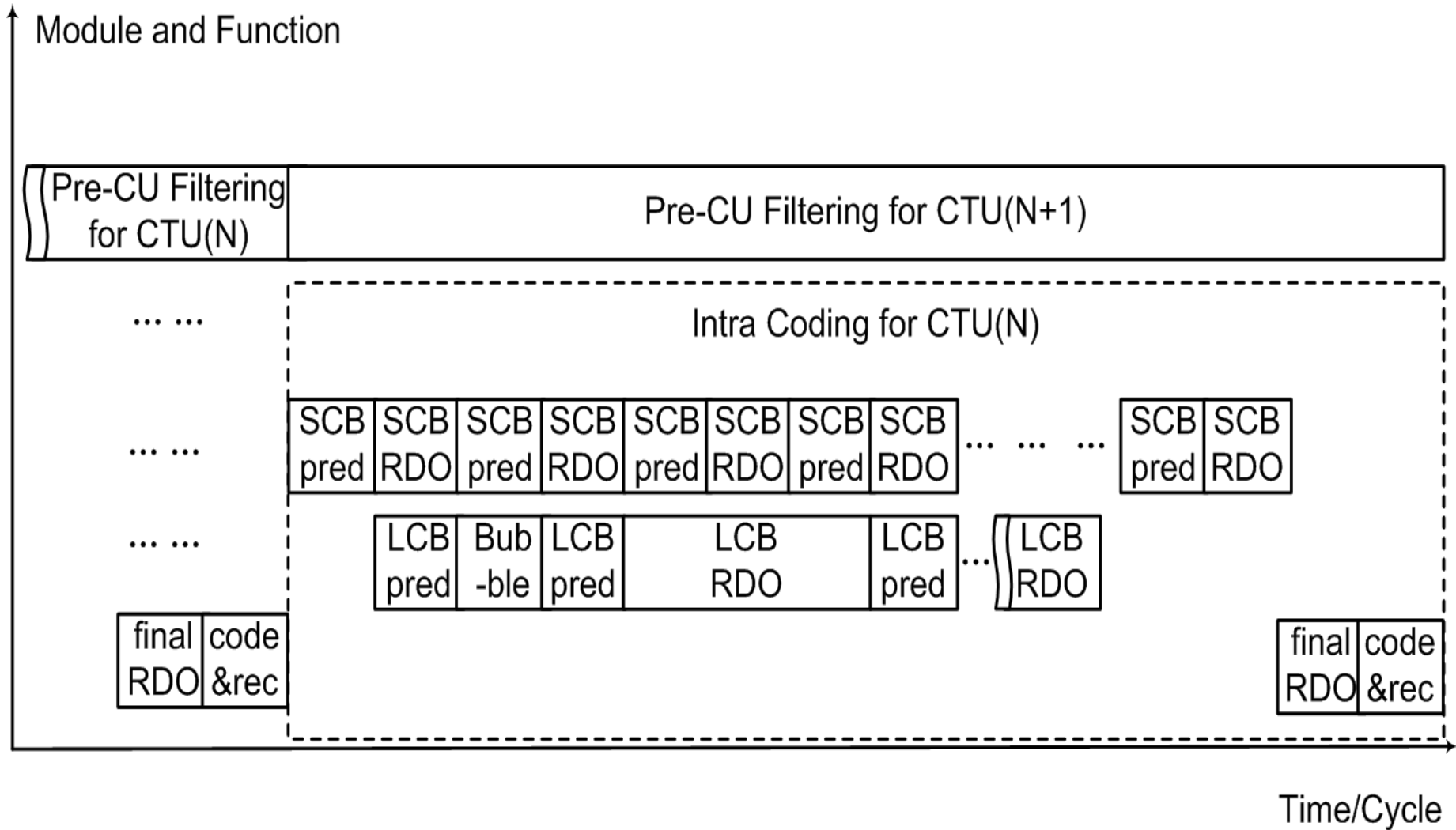


PU/TU Mode Search Structure

- Predictor
 - 128 pixel*mode/cycle
 - timing conflict: interrupt
 - critical path: small one
- Two Search Engines
 - SCB PU RDO
 - LCB PU RDO
 - fast search used
- Trans. Mode Decision
 - compare the best 2 cand.
 - search the best TU mode



Coding Timing Analysis



Coding Performance

- Environment
 - reference:HM-10
 - sequence: typical 22
 - QP={22,27,32,37}
- Result

Class	Sequence	BD-PSNR [dB]	BD-Rate [%]	Time Saved [%]
A	PeopleOnStreet	-0.21	4.61	61.4
	Traffic	-0.21	4.34	61.9
B	BasketballDrive	-0.17	6.73	61.7
	BQTerrace	-0.19	4.32	64.9
	Cactus	-0.14	4.28	72.6
	Kimono	-0.12	4.39	68.1
	ParkScene	-0.11	3.39	58.3
	Tennis	-0.18	5.92	62.0
C	BasketballDrill	-0.21	4.63	60.0
	BasketballDrillText	-0.21	4.75	60.6
	BQMall	-0.20	4.15	58.4
	RaceHorses	-0.19	3.38	58.6
D	BassketballPass	-0.24	4.80	58.8
	BlowingBubbles	-0.19	3.44	54.2
	BQSquare	-0.15	1.97	55.1
	Keiba	-0.21	4.02	60.3
E	SlideEditing	-0.41	2.94	61.1
	Vidyo1	-0.25	6.04	64.8
	Vidyo3	-0.23	5.35	63.4
	Vidyo4	-0.21	5.19	63.6
	Johnny	-0.21	5.15	63.5
	KristenAndSara	-0.24	5.86	63.6
Average		-0.20	4.53	61.7

	BD- PSNR [dB]	BD- Rate [%]	Time Saved [%]
Max	-0.41	6.73	72.6
Min	-0.11	1.97	54.2
Average	-0.20	4.53	61.7

Hardware Consumption

- Environment
 - described with Verilog HDL
 - synthesized with DC, TSMC90nm 1P9M technology
- Results
 - maximum speed, 357 MHz
 - fulfill HD1080p@44fps real-time intra coding

Module	Pre-Mode Filter	Rcnf. Predictor	32/16 CU RDO	8/4 PU RDO	Rcns. Datapath	Total
Gates(K)	214.1	817.3	781.3	450.6	507.2	2269.0
Pwr(mW)	26.2	101.4	25.2	32.9	32.2	217.9

Conclusion

- Fast HECV Intra Encoder
 - EdgeStrength based PredictError models
 - pre-CU/PU mode filtering
 - parallel fast search engines
- Results and Contributions
 - averagely 61.7% time save while 0.20dB BD-PSNR loss
 - stable and robust acceleration
 - 57% hardware saved totally in mode searching
 - max speed: 357MHz with TSMC90
 - support 4:2:0 HD1080p@44fps HEVC real-time encoding

THANK YOU!