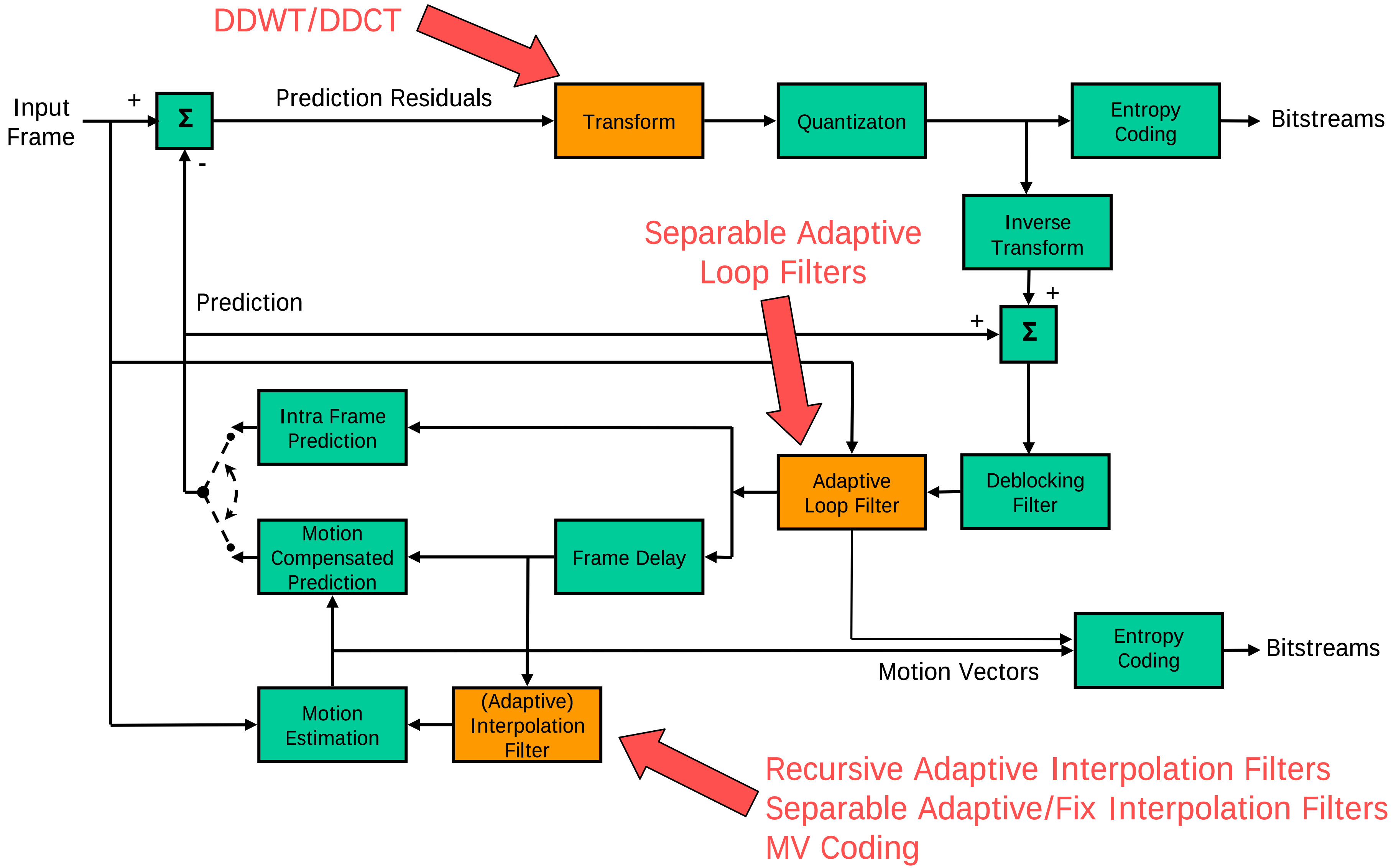
Video Coding Technology Proposed by Sony

Summary of Proposal

- 6 new tools to enhance coding efficiency of AVC
  - SFIF (Separable Fixed Interpolation Filter)
    - High precision separable filter
  - SAIF (Separable Adaptive Interpolation Filter)
    - AIF filter sets for single pred and bi-pred
    - Reduction of AIF filter overhead to improve B-Pic.
  - RAIF (Recursive Adaptive Interpolation Filter)
    - Time and spatial adaptation of filters
    - Single pass encoding
    - No need to transmit filter coefficients
  - MV coding
    - Adaptive MV predictor selecting
  - Directional Transform (DDCT/DDWT)
    - Directional transform to preserve edge
    - Simple integer transform without training data
    - Number of transforms are reduced using symmetry
  - SALF (Separable Adaptive Loop Filter)
    - Simple separable Loop Filter to improve prediction image

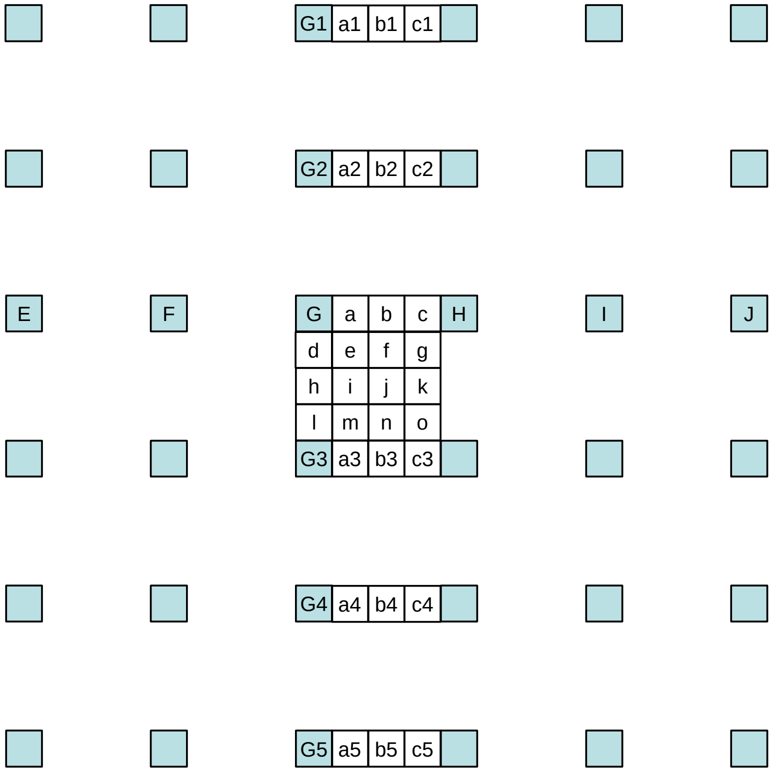
Codec Architecture



MC interpolation filter

Separable Fixed Interpolation Filter (SFIF)

To improve computational accuracy, both quarter pel and half pel value are derived directly by separable interpolation filter with higher precision



Step 1:
Horizontal interpolation is applied to derive pixels a, b and c

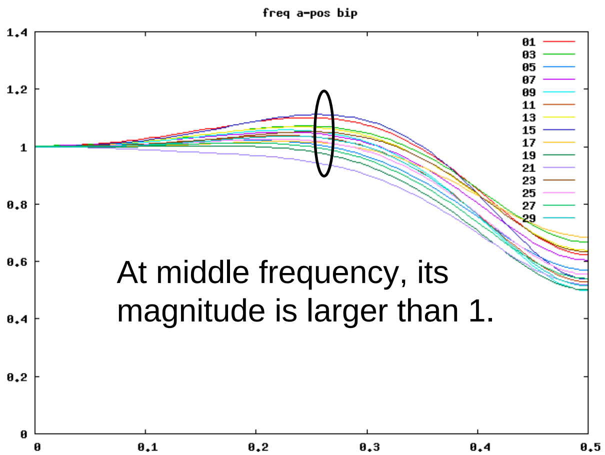
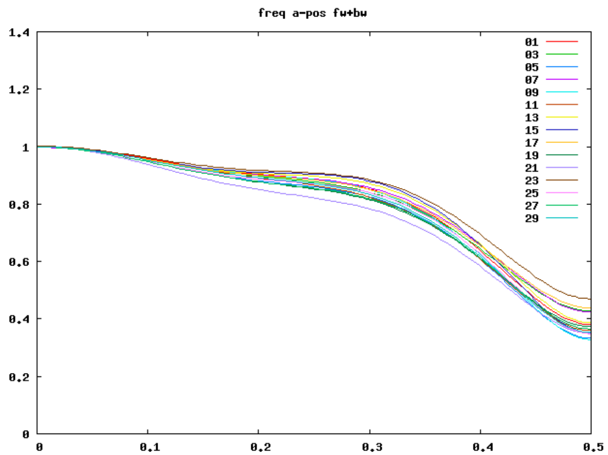
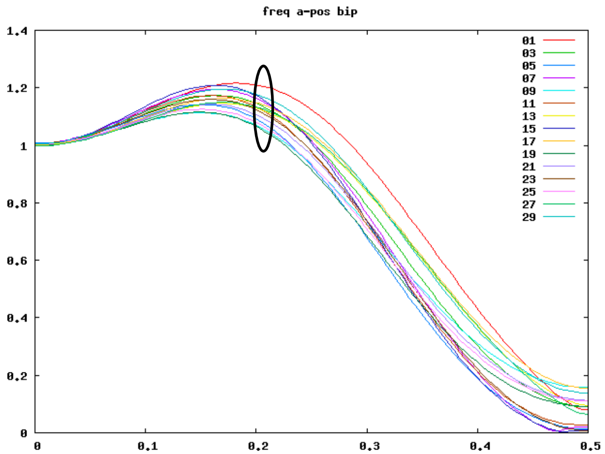
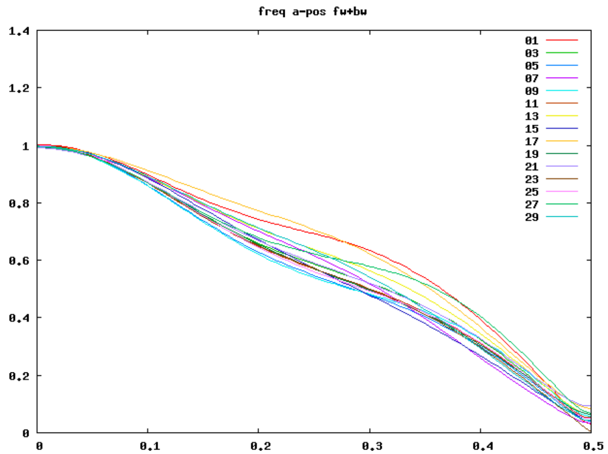
Step 2:
Vertical interpolation is applied to derive pixels d-o

Filter coefficients:
{3, -15, 111, 37, -10, 2, 3, -17, 78}

Separate Filters for Bi vs. Single

- Motivation
  - Compensation for loss of high frequency in bi-prediction due to averaging of L0 and L1 references
- Proposal
  - Switching filters based on bi-prediction vs. single prediction mode
    - In AIF scheme, it estimates 2 sets of filters for bi-pred and single pred
    - In FIF scheme, 2 sets of filters are pre-designed

Frequency response at a-position

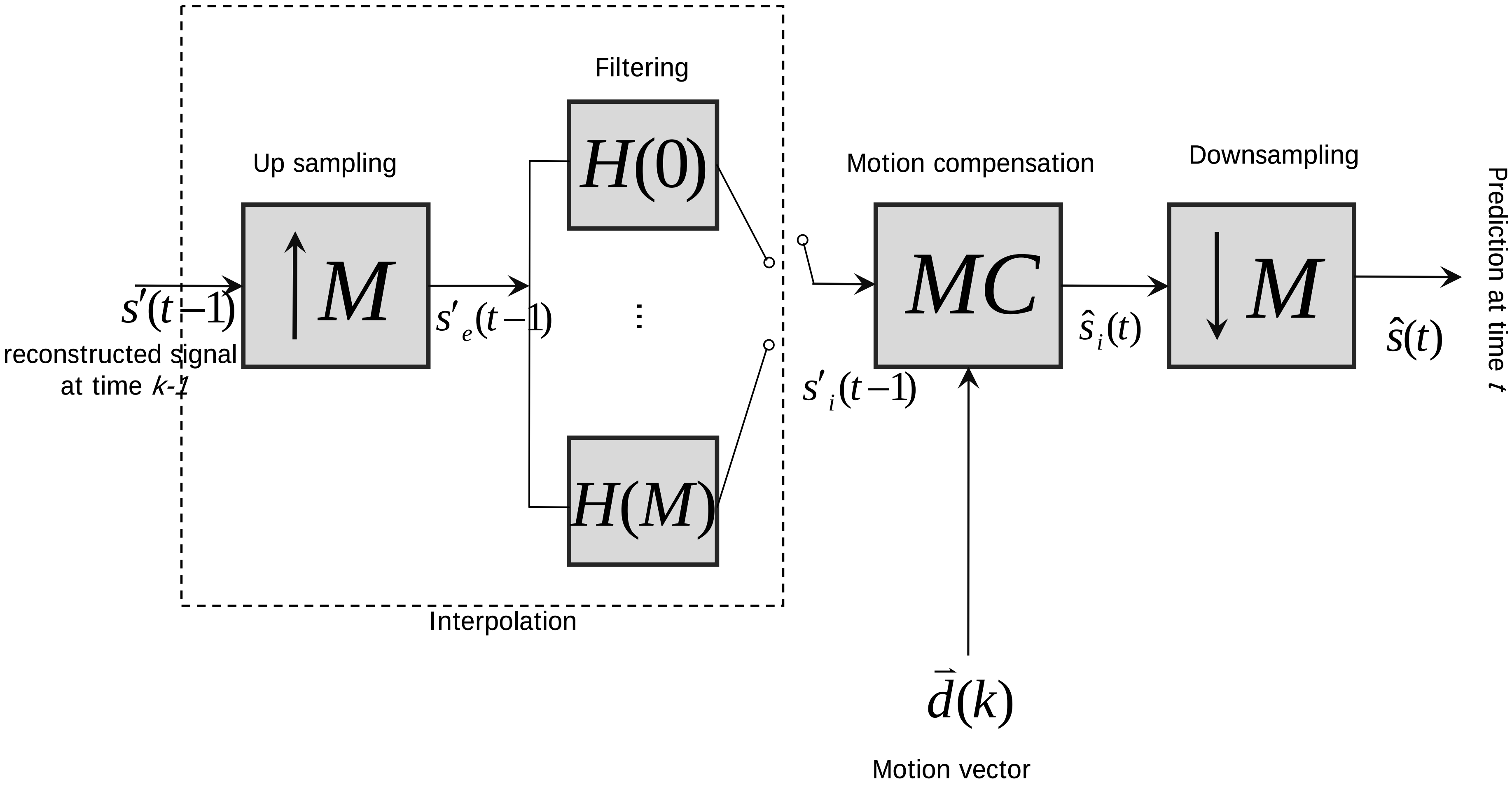
	Bi-prediction	Single prediction
QP22		
QP37		

Reduction of Filter Coefficients

Sub pel posision	P slice	B slice
a	6	6
b	3	3
c	6	same as reflection of a
d	6	6
e	6	same as reflection of d
f	6	same as reflection of d
g	6	same as reflection of d
h	3	3
i	3	same as h
j	3	same as h
k	3	same as h
l	same as reflection of d	same as reflection of d
m	same as reflection of e	same as reflection of d
n	same as reflection of f	same as reflection of d
o	same as reflection of g	same as reflection of d
Total of number of coefficients	51	18 (36 if two sets of filters are used)

Recursive Adaptive Interpolation Filter (RAIF)

Motion Compensated Prediction



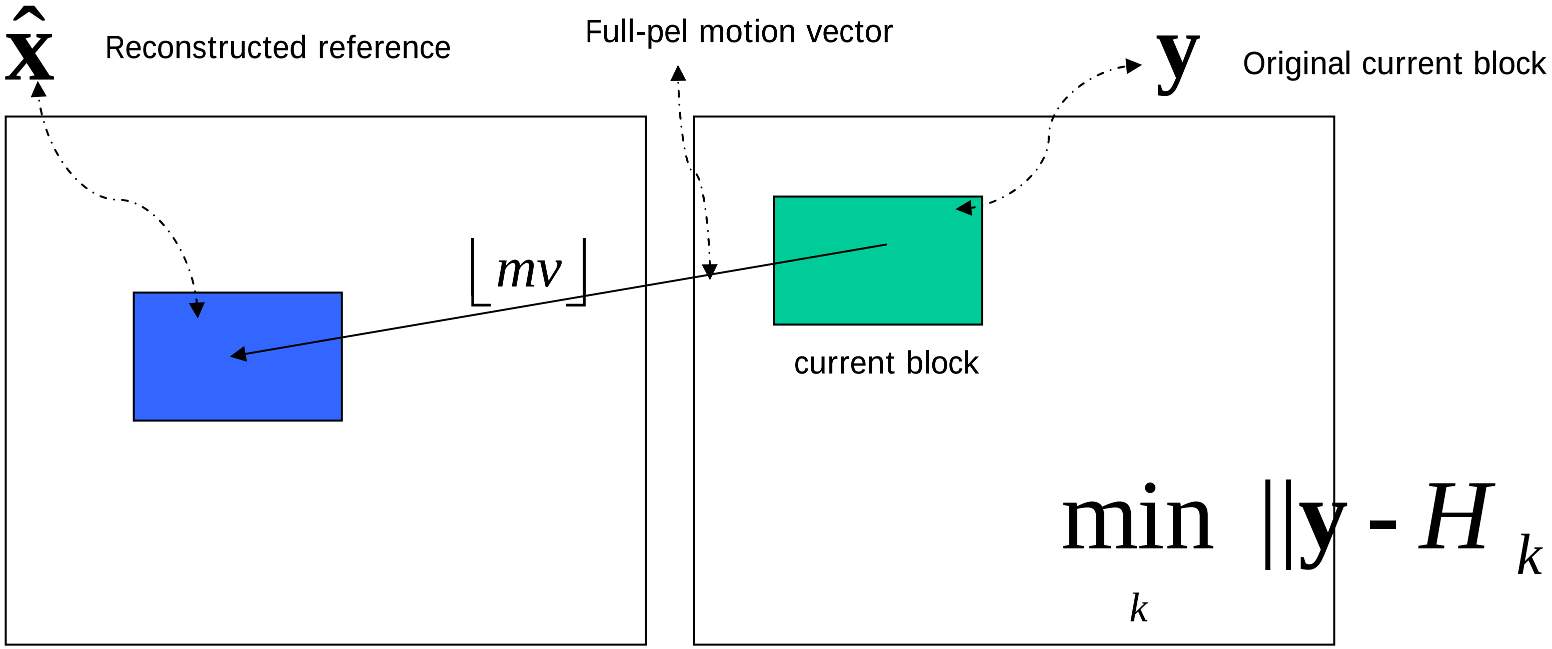
RAIF Features

- Motivation:
  - Time and spatial adaptation as opposed to AVC's fixed interpolation filters

$$H = \{1, -5, 20, 20, -5, 1\}$$

- Proposed technique:
  - Time and spatial adaptivity of the filters
    - Filters are updated within a picture
  - Filter computation is recursive based on the reconstructed sample values
    - No need to signal filter coefficients to the decoder since it can independently compute the filter coefficients from the previously decoded signal

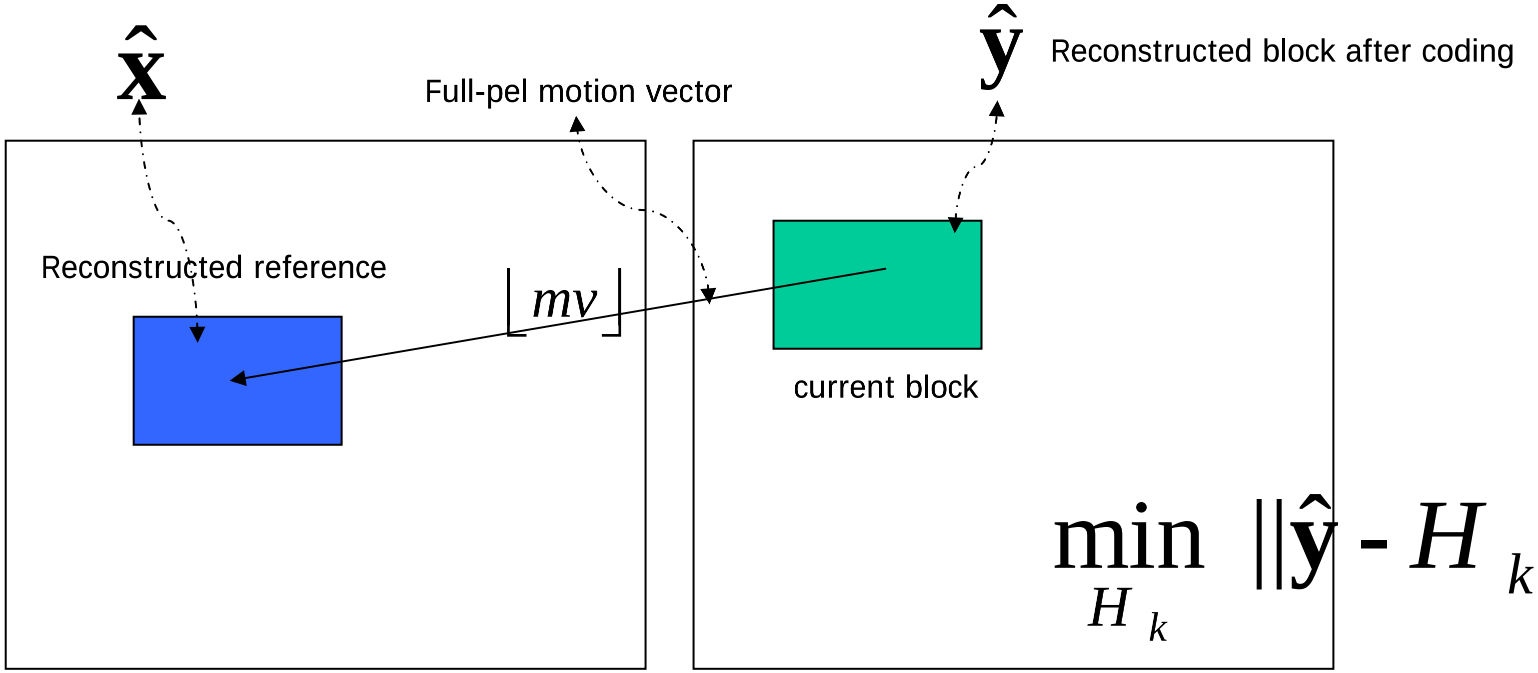
RAIF Computation-First Stage



$$\min_k \| \mathbf{y} - H_k \hat{\mathbf{x}} \|_1$$

- Motion estimation
  - Obtain a full-pel motion vector
  - Find the best filter index *k*
    - *This is equivalent to subpel motion estimation in AVC*

RAIF Computation-Second Stage



$$\min_{H_k} \|\hat{\mathbf{y}} - H_k \hat{\mathbf{x}}\|_2$$

- Filter update
  - Perform residual coding and obtain $\hat{\mathbf{y}}$
  - Update filter index k based on the newly coded block in L_2 norm

Filter Update

- The best filter for each block is obtained

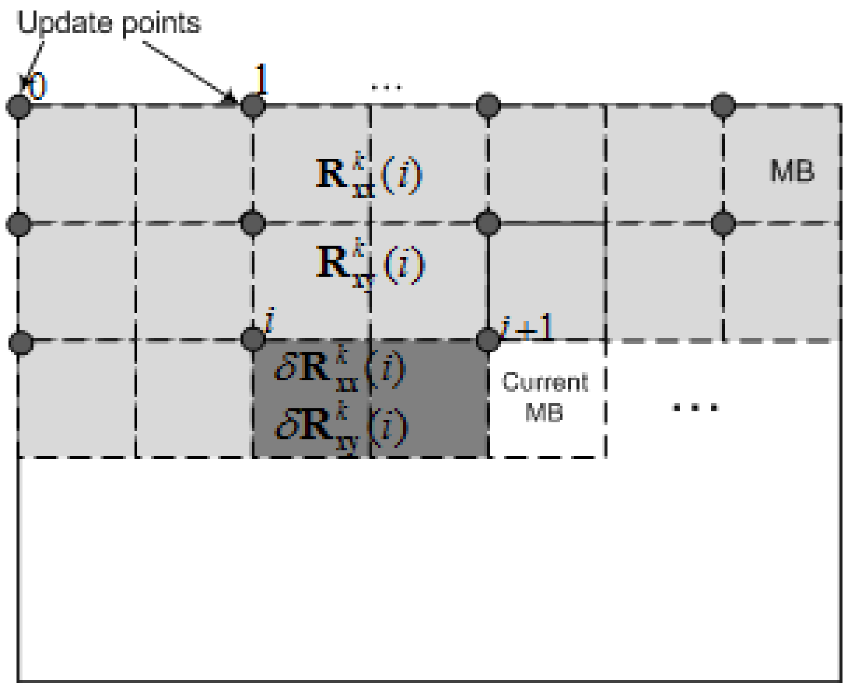
$$\min_{H_k} \|\hat{\mathbf{y}} - H_k \hat{\mathbf{x}}\|_2 \quad \Longleftrightarrow \quad H_k = (\mathbf{R}^k_{\mathbf{xx}})^{-1}\mathbf{R}^k_{\mathbf{x}\hat{\mathbf{y}}}$$

- However, a slow adaptation of the filters to the local statistics provides the best performance
  - Old statistics should be also considered in updating the filters

Filter Update

- Initially (at IDR pictures) set

$$H_k(0) = \left(\mathbf{R}_{\mathbf{xx}}^k(0)\right)^{-1} \mathbf{R}_{\mathbf{xy}}^k(0)$$



- Before reaching update point, compute the update auto- and cross-correlation matrices from reconstructed signal

$$\delta\mathbf{R}_{\mathbf{xx}}^k(i) \qquad \delta\mathbf{R}_{\mathbf{xy}}^k(i)$$

- Update auto- and cross-correlation matrices and thus the corresponding filters

$$\mathbf{R}_{\mathbf{xx}}^k(i+1) = (1-\alpha)\,\mathbf{R}_{\mathbf{xx}}^k(i) + \alpha\;\delta\mathbf{R}_{\mathbf{xx}}^k(i)$$

$$\mathbf{R}_{\mathbf{xy}}^k(i+1) = (1-\alpha)\,\mathbf{R}_{\mathbf{xy}}^k(\mathrm{i}) + \alpha\;\delta\mathbf{R}_{\mathbf{xy}}^k(i)$$

$$H_k = (\mathbf{R}_{\mathbf{xx}}^k)^{-1}\mathbf{R}_{\mathbf{x}\hat{\mathbf{y}}}^k$$

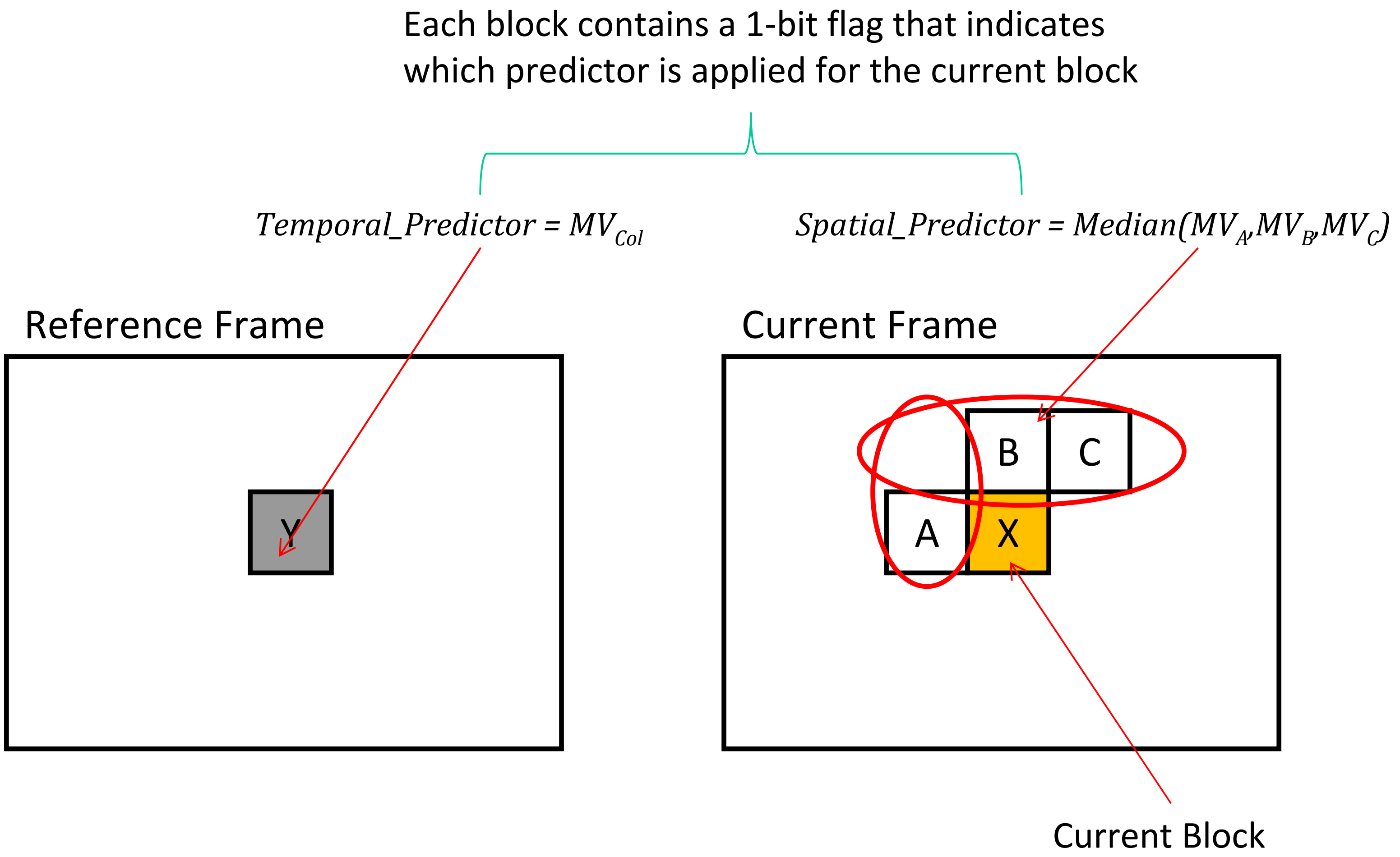
RAIF vs. AIF

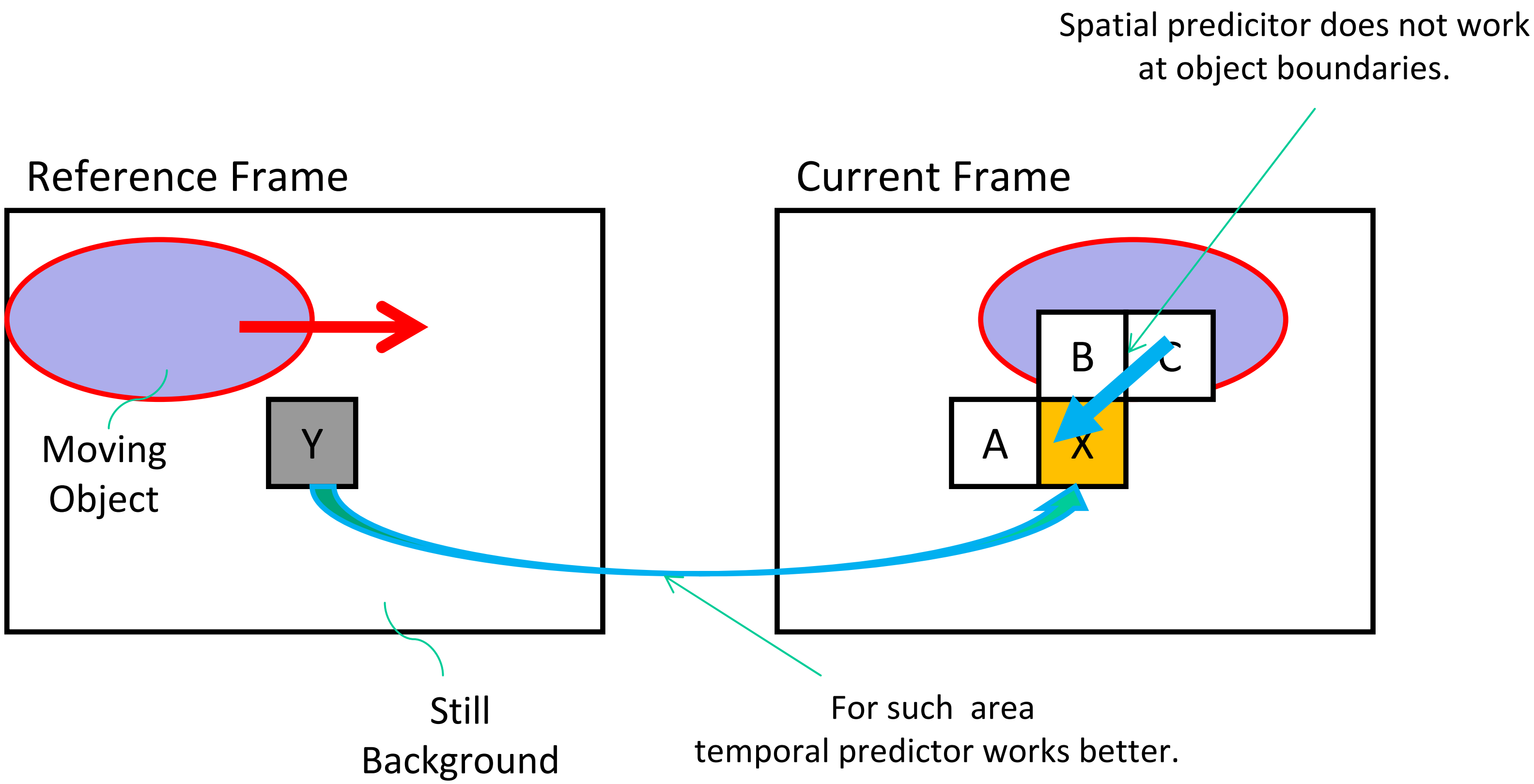
- Spatial adaptation to local frame statistics as well as time adaptation
- No need to transmit filter coefficients as the decoder computes them from reconstructed signal
- Lower encoder complexity (single pass encoding outperforms two pass AIF) at the expense of higher decoder complexity

Motion Vector Coding

Motion Vector Coding [1/]

- MV predictor of AVC
  - Spatial predictor w median of 3 neighboring MVs (other than temporal direct mode)
- Proposal
  - In addition to this spatial predictor, temporal predictor can be used
- Each MB contain a 1-bit flag that indicates which predictor is applied for the current block.


Each block contains a 1-bit flag that indicates
which predictor is applied for the current block
Temporal_Predictor = MV_Col
Spatial_Predictor = Median(MV_A,MV_B,MV_C)
Reference Frame
Current Frame
Y
B
C
A
X
Current Block



If mv_prediction_definition_flag=0,
severe degradation is observed
b/w boundaries of the moving men and the floor.

Setting as mv_prediction_definition_flag=1
provides better subjective quality
as well as better coding efficiency.

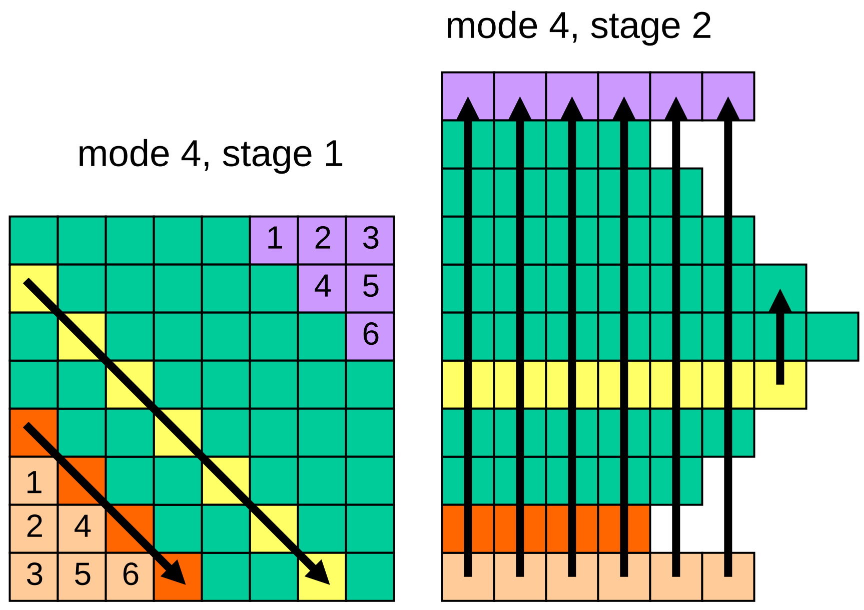
Transform

The DDCT Transforms

- For each intra prediction mode, DDCT provides
  - A fixed transform
  - A fixed scanning pattern
- Two stage transforms:
  - Stage 1: transform along the prediction direction
  - Stage 2: transform perpendicular to the prediction direction
- Fixed scanning pattern: depends on
  - Intra prediction mode
  - QP

Example (DDCT 8x8, Mode 4)

- Corner treatment: order the pixels as numbered and feed into a 6-point DCT



Properties of the DDCT

- Adaptivity:
  - Each intra prediction mode will be associated with a different transform and fixed scanning pattern
- Directionality:
  - The transform are conducted first along the prediction direction
  - Hence, potentially help to preserve the visual quality
- Symmetry
  - Mode 0, 1, 2 are the same; mode 3 and 4 are similar; mode 5-8 can be derived from a “core” transform
  - Hence, reduce the hardware complexity
- Complexity:
  - Similar to a separable transform

Quantization

- Quantization

$$Y_Q = [Y \cdot Q(QP\%6) + f_{enc}] >> (22 + QP/6)$$

$$Q = [410 \quad 364 \quad 328 \quad 287 \quad 260 \quad 231]$$

$$f_{enc} = 2^{22+QP/6}/3$$

- Dequantization

$$Y = Y_Q \cdot R(QP\%6) << QP/6$$

$$R = [40 \quad 45 \quad 50 \quad 57 \quad 63 \quad 71]$$

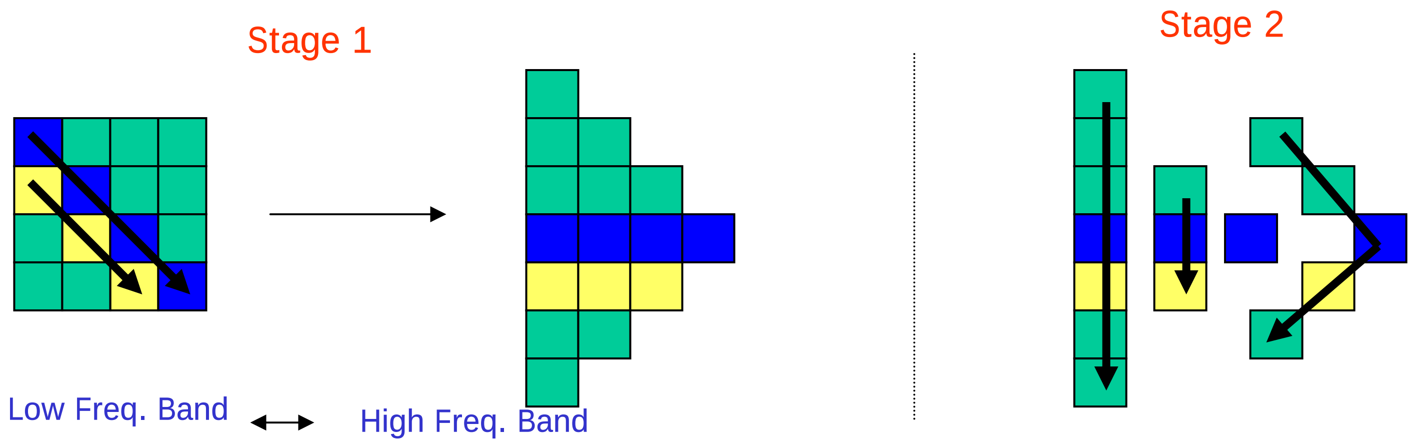
- Reconstruction

$$X_S = (X + f_{dec}) >> 20$$

$$f_{dec} = 2^{19}$$

DDWT

- Motivation of DDWT:
  - Design intra transforms for better visual quality:
    - Transform along the prediction direction for better compactness
    - Short-tap filters to reduce ringing artifacts
- Properties:
  - Fixed transforms with integer arithmetic
  - Symmetry to reduce the number of transforms
    - Similar transforms between modes 0-1, between modes 3-4, and between modes 5-8
- Transforms:
  - Stage 1: along the intra prediction direction
  - Stage 2: grouping based on frequency and spatial localization
- Example of 4x4, mode 4:



DDWT

- Some other implementation issues
  - Non-directional modes: integer DCT
  - Fixed scanning pattern based on
    - The intra prediction mode and
    - QP
- Intermediate transforms:
  - Derived from the Haar wavelet transform
    - With multiple decomposition levels
    - Might have different sizes, odd sizes
  - Special treatment of the “DC” coefficients
    - Problem: different scaling because of different transform sizes
    - Solution: modifying Haar to ensure no DC leakage into AC coefficients
- Mixed DDCT-DDWT Transforms
  - DDCT in stage 1 and DDWT in stage 2
  - Advantages from both techniques:
    - DDCT: energy compactness in smooth lines (along the direction)
    - DDWT: preserve edges thanks to short-tap filters (across the edges)

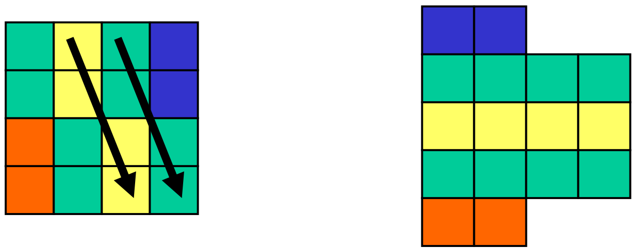
Example of Transforms

$$W_3 = \begin{bmatrix} 74 & 74 & 74 \\ -104 & 52 & 52 \\ 0 & -91 & 91 \end{bmatrix}$$

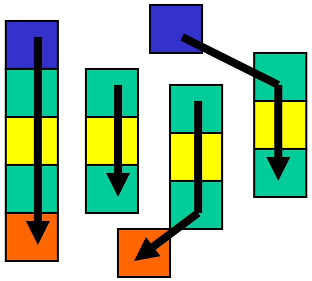
$$W_4 = \begin{bmatrix} 64 & 64 & 64 & 64 \\ 64 & 64 & -64 & -64 \\ 91 & -91 & 0 & 0 \\ 0 & 0 & 91 & -91 \end{bmatrix}$$

Example (DDWT 4x4 Mode 5)

mode 5, stage 1



mode 5, stage 2



Low Freq. Band ←→ High Freq. Band

Properties of the DDWT

- Localization:
  - Both in time and frequency,
  - Inherited from wavelet
- Adaptivity:
  - Each intra prediction mode will be associated with a different transform and fixed scanning pattern
- Directionality:
  - The transform are conducted first along the prediction direction
  - Hence, potentially help to preserve the visual quality
- Symmetry
  - Mode 0, 1, 2 are the same; mode 3 and 4 are similar; mode 5-8 can be derived from a “core” transform
  - Hence, reduce the hardware complexity
- Complexity:
  - Similar to a separable transform

Quantization

- Quantization

$$Y_Q = [Y \cdot Q(QP\%6) + f_{enc}] >> (22 + QP/6)$$

$$Q = [410 \quad 364 \quad 328 \quad 287 \quad 260 \quad 231]$$

$$f_{enc} = 2^{22+QP/6}/3$$

- Dequantization

$$Y = Y_Q \cdot R(QP\%6) << QP/6$$

$$R = [40 \quad 45 \quad 50 \quad 57 \quad 63 \quad 71]$$

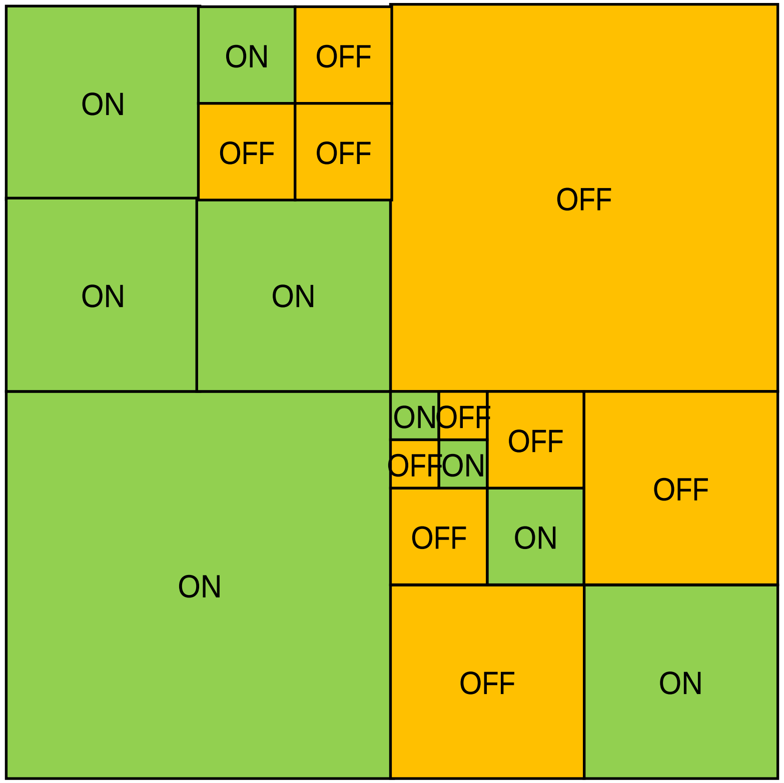
- Reconstruction

$$X_S = (X + f_{dec}) >> 20$$

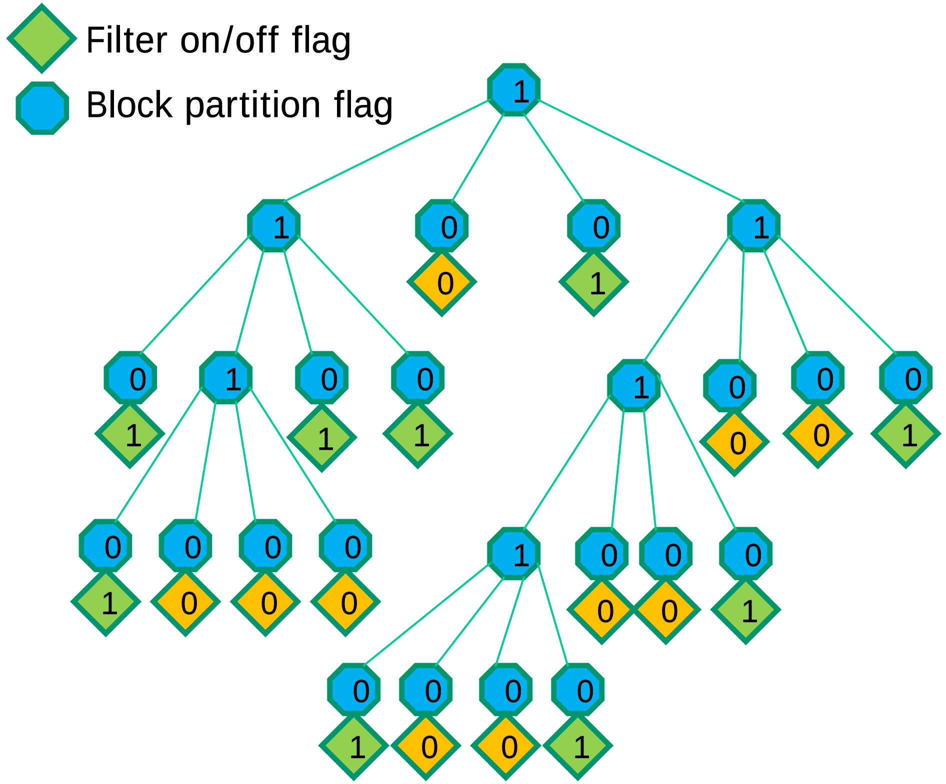
$$f_{dec} = 2^{19}$$

Adaptive Loop Filter

Quadtree Partition and Signaling



Example quadtree partition



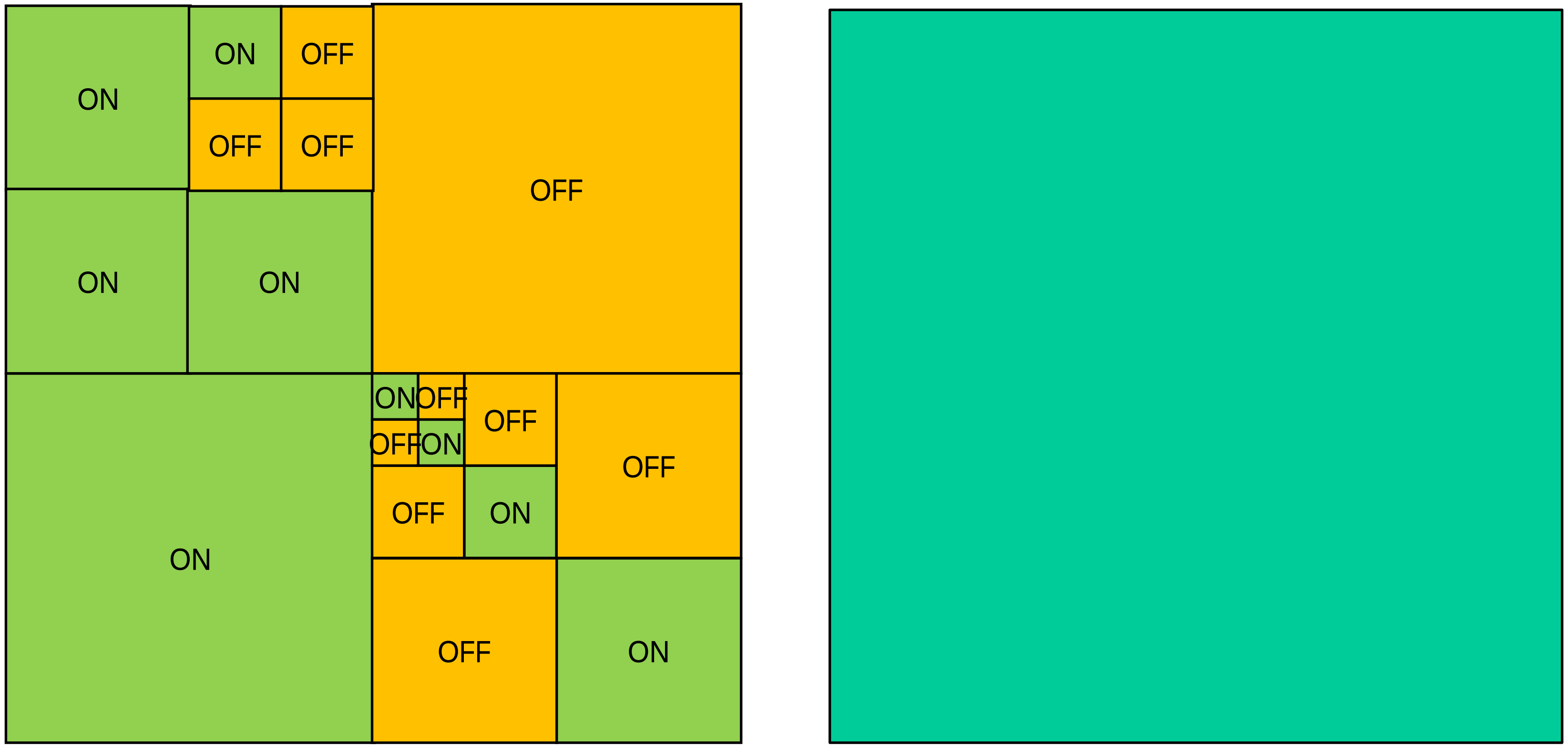
Example quadtree data structure

- The quadtree partition is determined by multi-passes rate-distortion optimization

Non-Separable Filter for I

Picture X after Deblocking

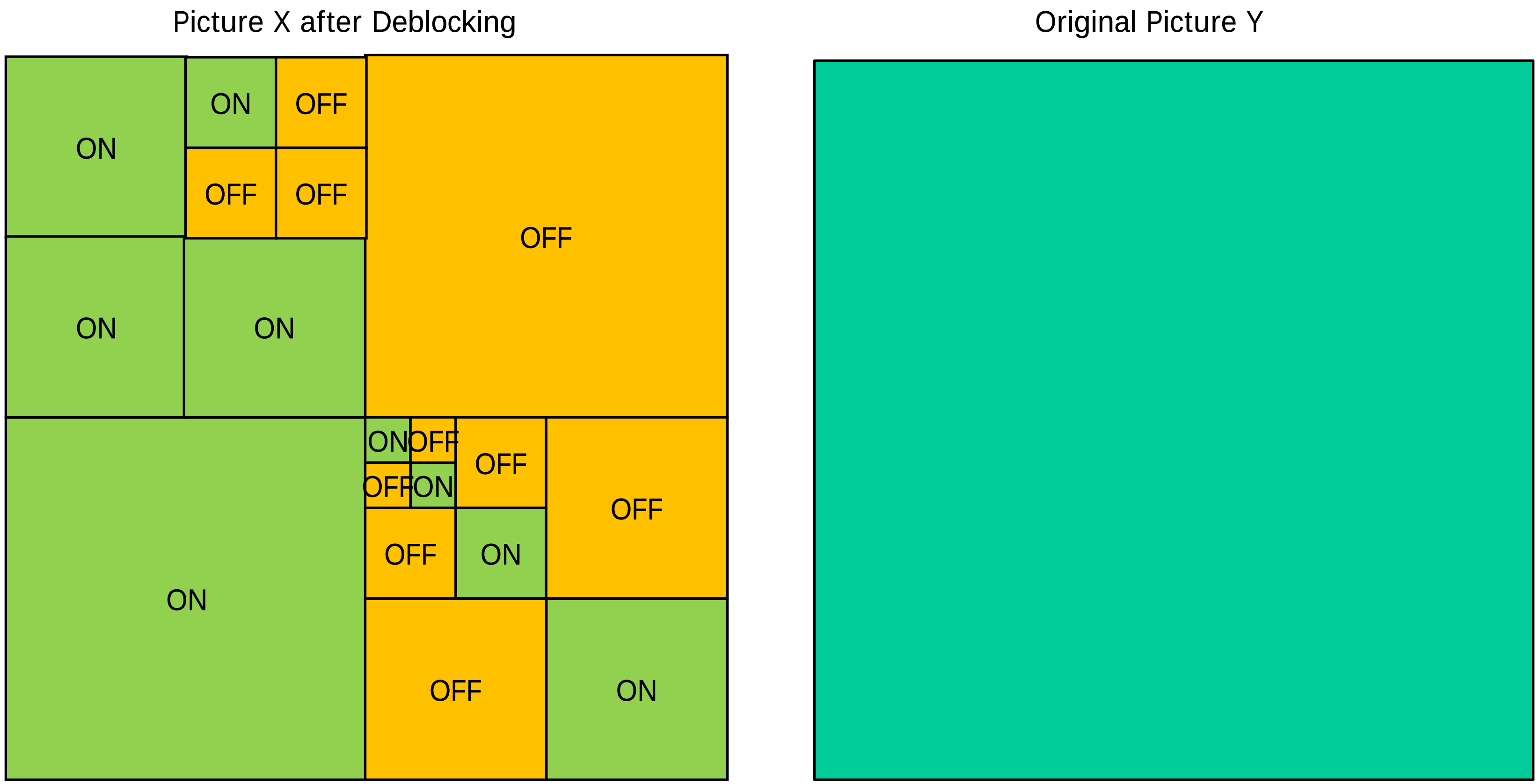
Original Picture Y



To maximize spatial quality, find zero-phase non-separable filter f_0 and DC bias f_l which minimizes

$$\sum_{m \text{ in "ON" region}} \left(y[m] - \sum f_o[k]\, x[m-k] - f_l \right)^2$$

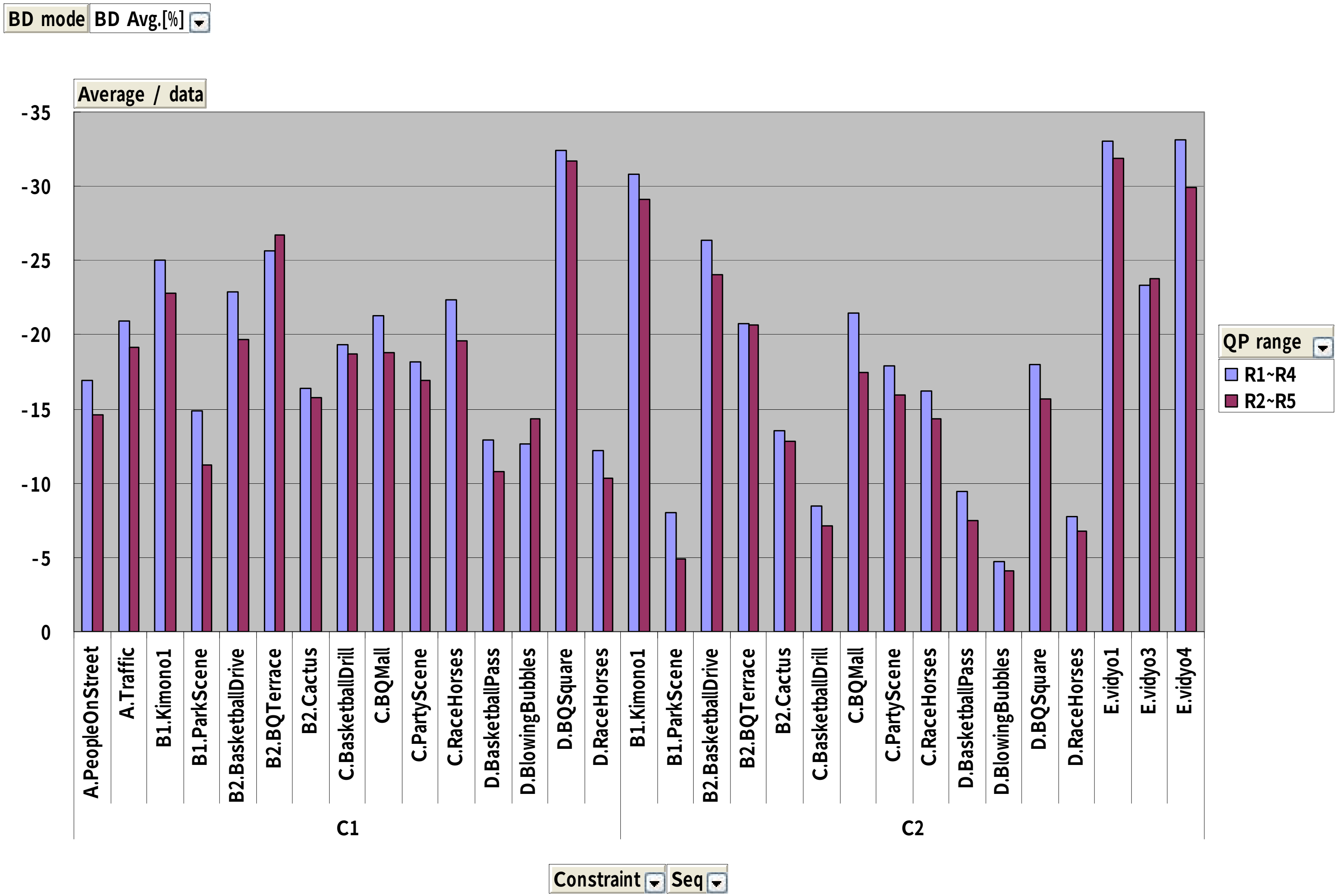
Separable Filter for P and B

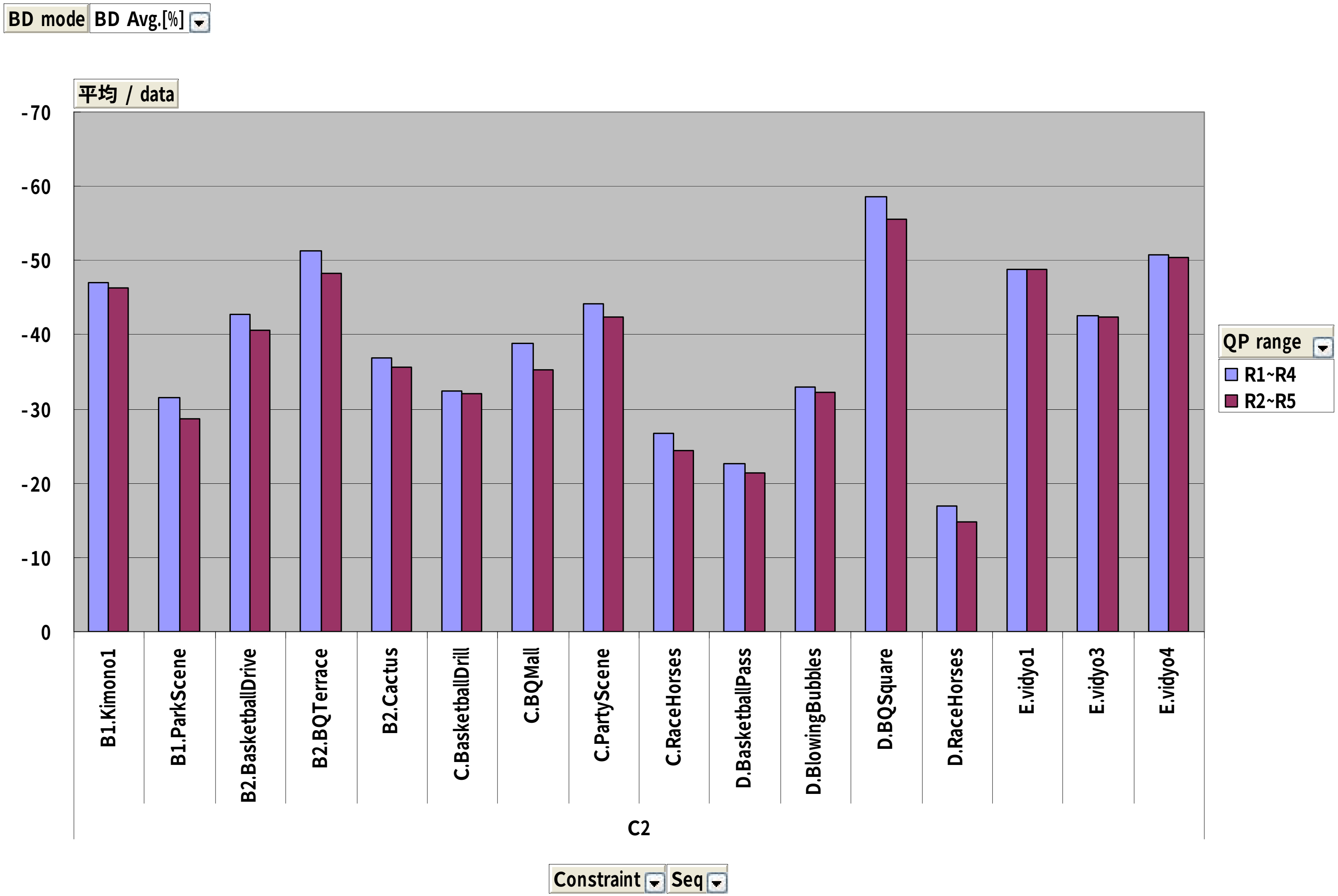


For better temporal stability, find separable filter f_0 and DC bias f_l which minimizes

$$\sum_{m \text{ in "ON" region}} \left(y[m] - \sum_{k} f_o[k]\, x[m-k] - f_l \right)^2 + \sigma \left(\sum_{k} \left(f_o[k] \right)^2 + f_l^2 \right)$$

Performance Comparison


BD mode
BD Avg.[%]
Average / data
-35
-30
-25
-20
-15
-10
-5
0
QP range
R1~R4
R2~R5
A.PeopleOnStreet
A.Traffic
B1.Kimono1
B1.ParkScene
B2.BasketballDrive
B2.BQTerrace
B2.Cactus
C.BasketballDrill
C.BQMall
C.PartyScene
C.RaceHorses
D.BasketballPass
D.BlowingBubbles
D.BQSquare
D.RaceHorses
B1.Kimono1
B1.ParkScene
B2.BasketballDrive
B2.BQTerrace
B2.Cactus
C.BasketballDrill
C.BQMall
C.PartyScene
C.RaceHorses
D.BasketballPass
D.BlowingBubbles
D.BQSquare
D.RaceHorses
E.vidyo1
E.vidyo3
E.vidyo4
C1
C2
Constraint
Seq



BD mode
BD Avg.[%]
平均 / data
-70
-60
-50
-40
-30
-20
-10
0
B1.Kimono1
B1.ParkScene
B2.BasketballDrive
B2.BQTerrace
B2.Cactus
C.BasketballDrill
C.BQMall
C.PartyScene
C.RaceHorses
D.BasketballPass
D.BlowingBubbles
D.BQSquare
D.RaceHorses
E.vidyo1
E.vidyo3
E.vidyo4
C2
QP range
R1~R4
R2~R5
Constraint
Seq

Summary of Proposal

- 6 new tools to enhance coding efficiency of AVC
  - RAIF (Recursive Adaptive Inerpolation Filter)
  - SFIF (Separable Fixed Interpolation Filter)
  - SAIF (Separable Adaptive Interpolation Filter) with improvement for B-pic
  - MV coding
  - Directional Transform (DDCT/DDWT)
  - SALF (Separable Adaptive Loop Filter)