SIPL SIGNAL & IMAGE PROCESSING LAB

Electronics
Computers
Communications

Technion

Content Insertion into Compressed Video in the Coding Domain

Nimrod Peleg

http://sipl.technion.ac.il

Tel-Hai Conference on Signal Processing and Multimedia Applications, June 2007

The Staff

- Prof. David Malah
- Yair Moshe
- Naama Hait
- Tamar Shoham – Thanks for the slides !

- 12 Undergraduate students

Algorithms, software implementations,

real-time implementations

Outline

- Content insertion: Motivation
- Classic video compression
- Video-in-video algorithms
- If time allows: H.264 compression novelties

Motivation

- There is a need for content insertion in the communication industry (logos, subtitles, advertisements, etc.)
- The insertion must support logos that change in time as well as in space
- The insertion should be fast and low complexity

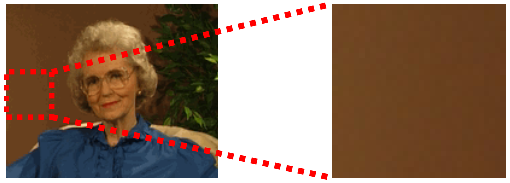
Broadcasting Vs. Personal video

Introduction to video compression

Redundancy ⇔ Irrelevancy in video clips

- Redundancy

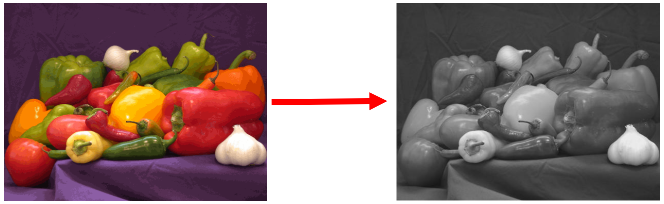
✓**Spatial**: pixel-to-pixel or spectral correlation

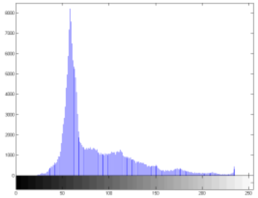


✓**Temporal**: frame-to-frame similarity

N N+1 N+2

✓**Statistical**: non-uniform distribution of data





Luminance Histogram

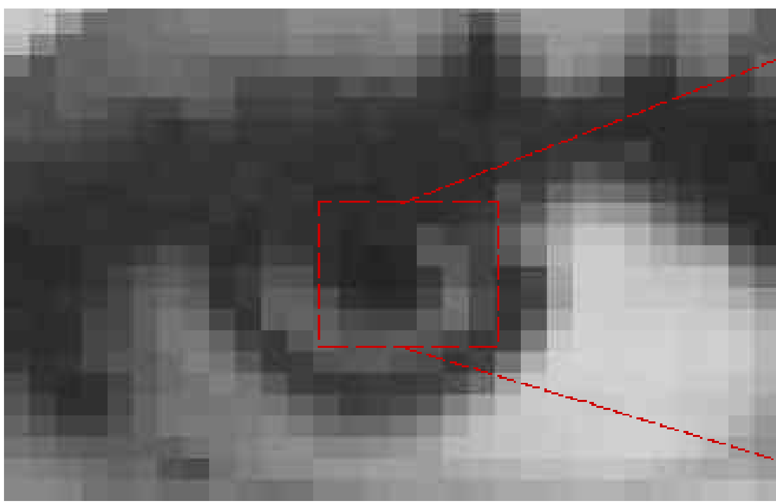
- Irrelevancy relates to an **observer viewing** an image

Redundancy + Irrelevancy ⇨ high compression ratio

⇧ loss

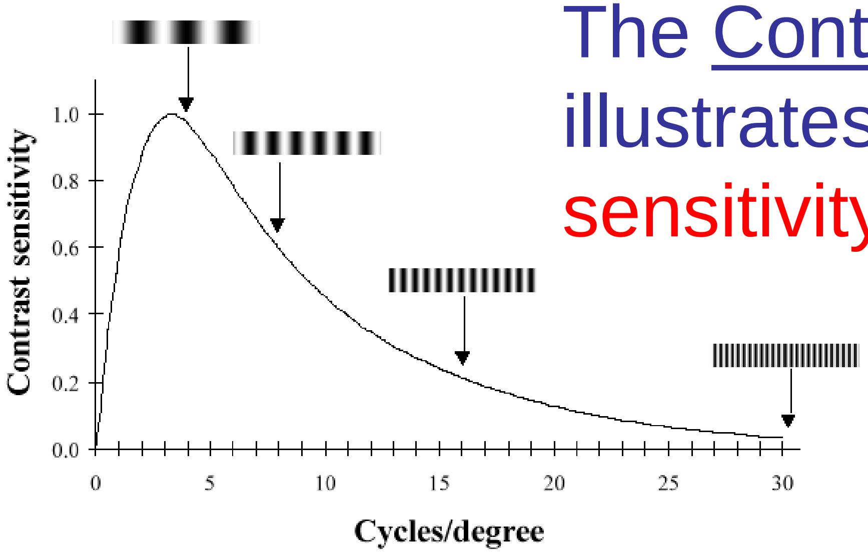
Spatial Redundancy & Irrelevancy

- What is the value of the missing pixel? It is 39.
- How critical is it to correctly reproduce it?

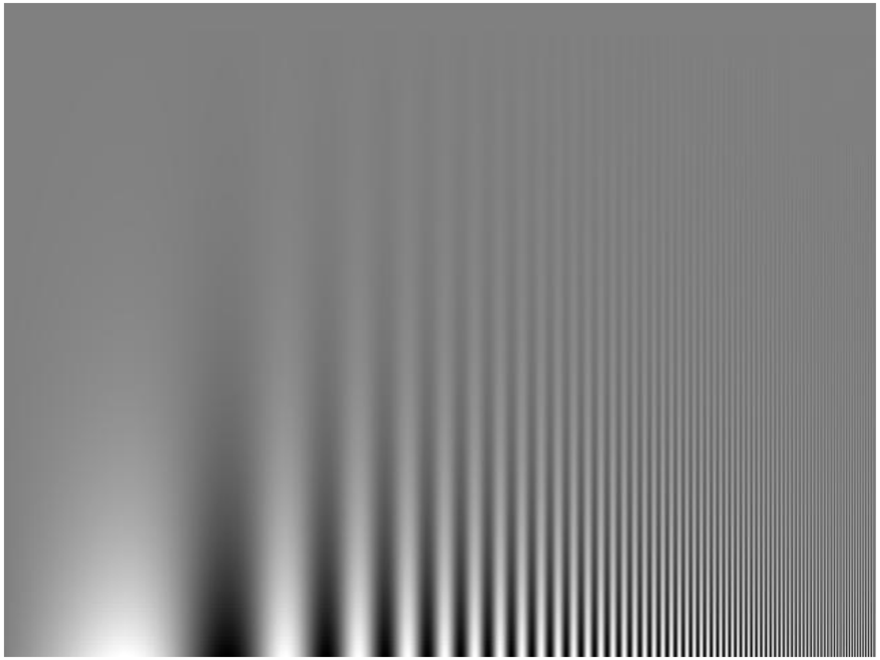


46	44	46	47	44	50	56
50	40	46	48	76	62	69
45	39	36	40	88	87	65
48	38	36	?	56	90	65
76	48	41	43	69	112	77
90	72	66	66	90	108	74
98	86	86	91	83	72	57

Irrelevancy: CSF



The Contrast Sensitivity Function illustrates the limited perceptual sensitivity to high spatial frequencies



Irrelevancy: Visual Masking

original

distortion in smooth area

distortion in busy area

Irrelevancy: Visual Masking

original

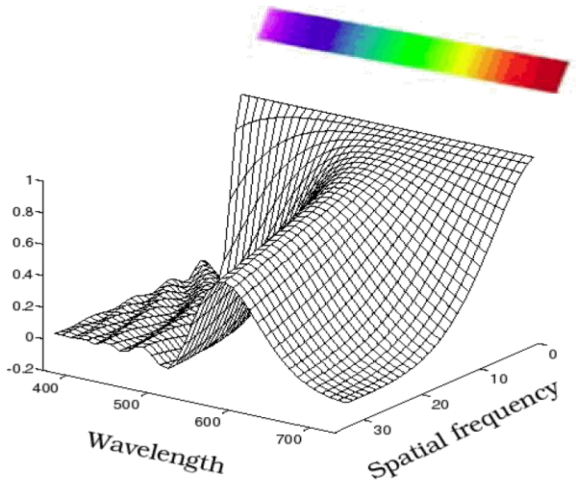
distortion in smooth area

distortion in busy area

Video Compression enablers

Video clips are:

- Spatially redundant
- Temporally redundant
- Statistically redundant

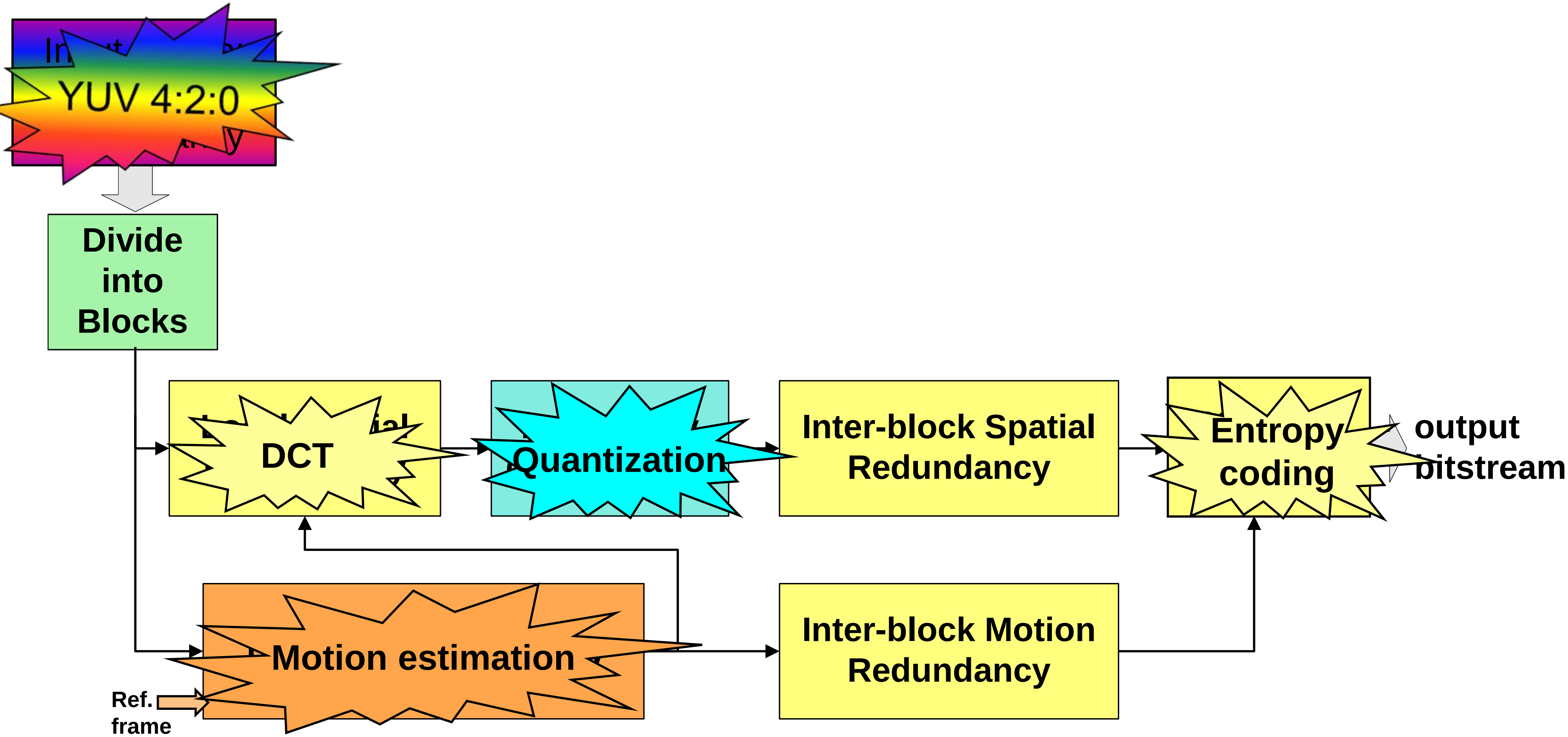


Chromatic Modulation Transfer Function

Human eyes are:

- Less sensitive to high spatial frequencies
- Less sensitive to chromatic resolution
- Less sensitive to distortions in “busy” areas

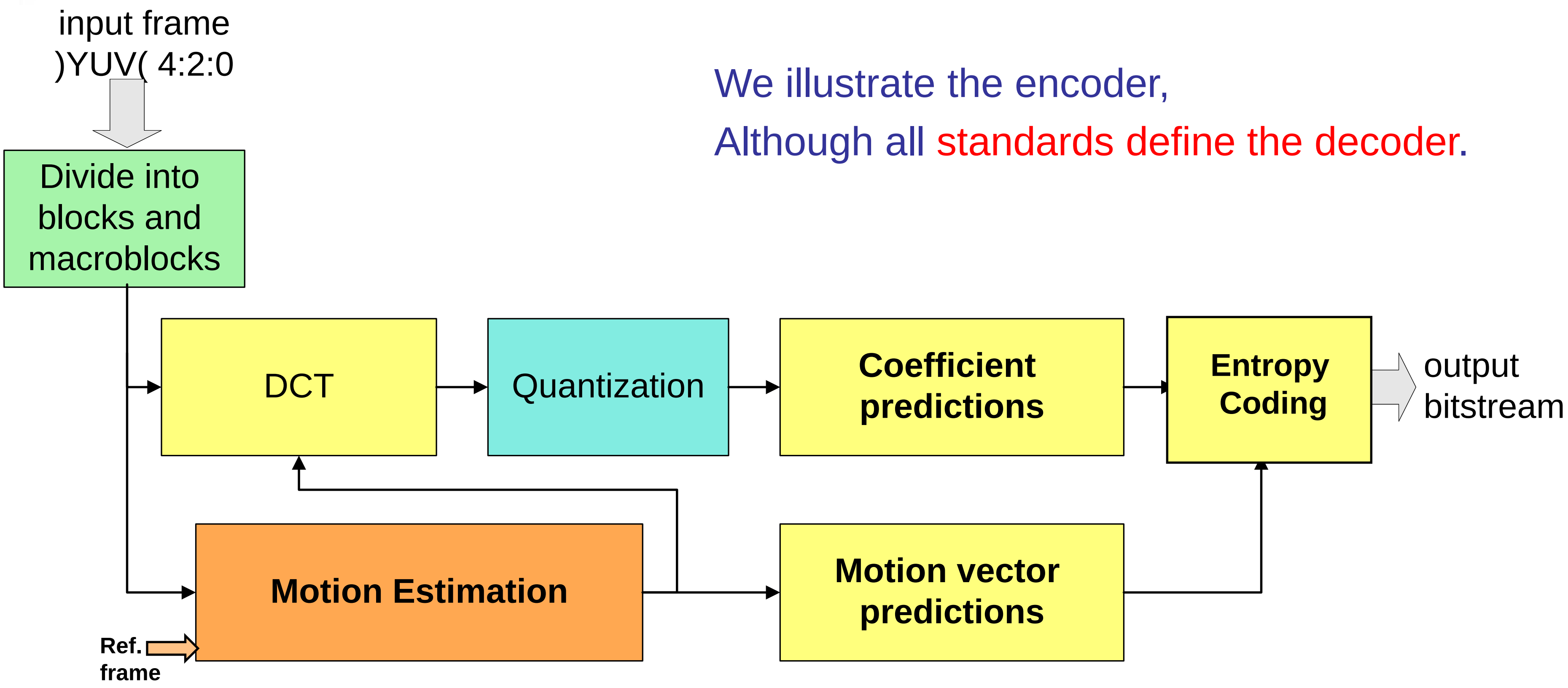
The video coding scheme



Each block removes some redundancy / irrelevancy element

The video coding scheme

We illustrate the encoder,
Although all standards define the decoder.

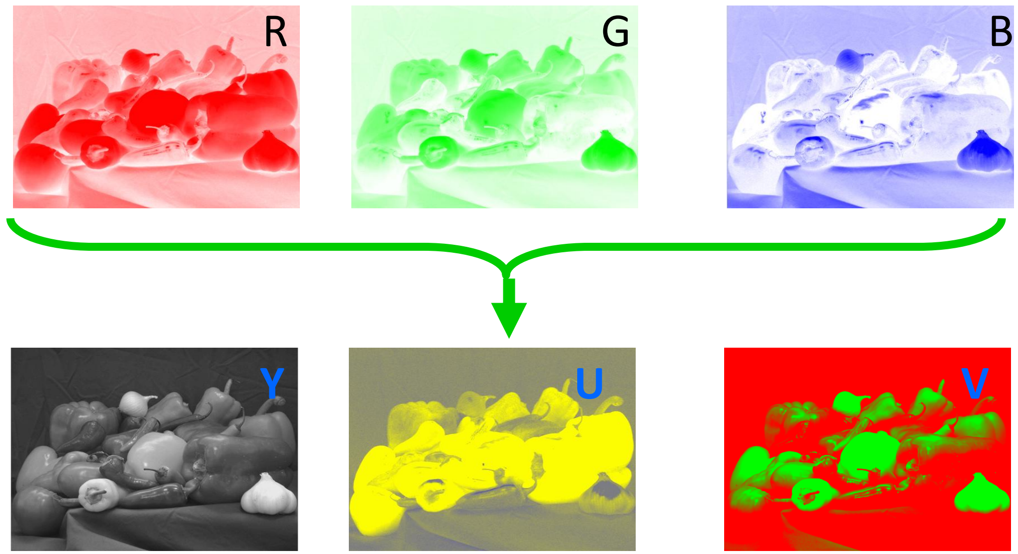


- In addition, various pre/post processing operations may be applied to the input/decoded frames.

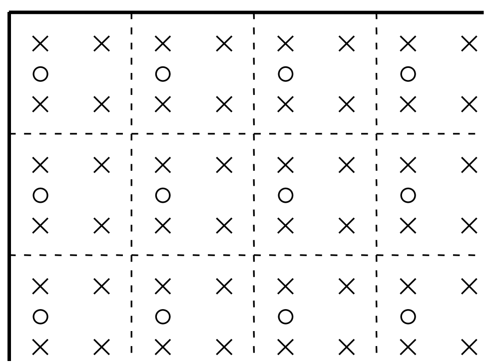
What is “YUV 4:2:0”?

YUV color representation

=

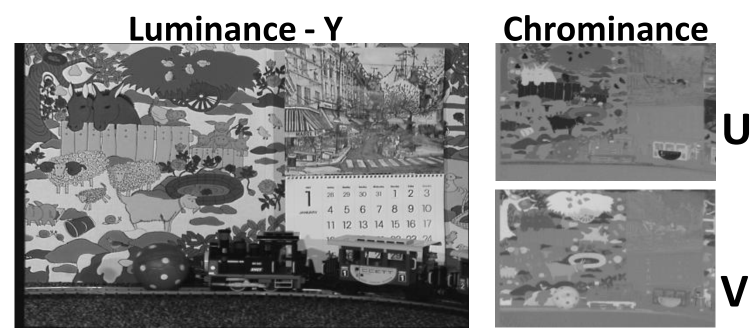


Chromatic down-sampling

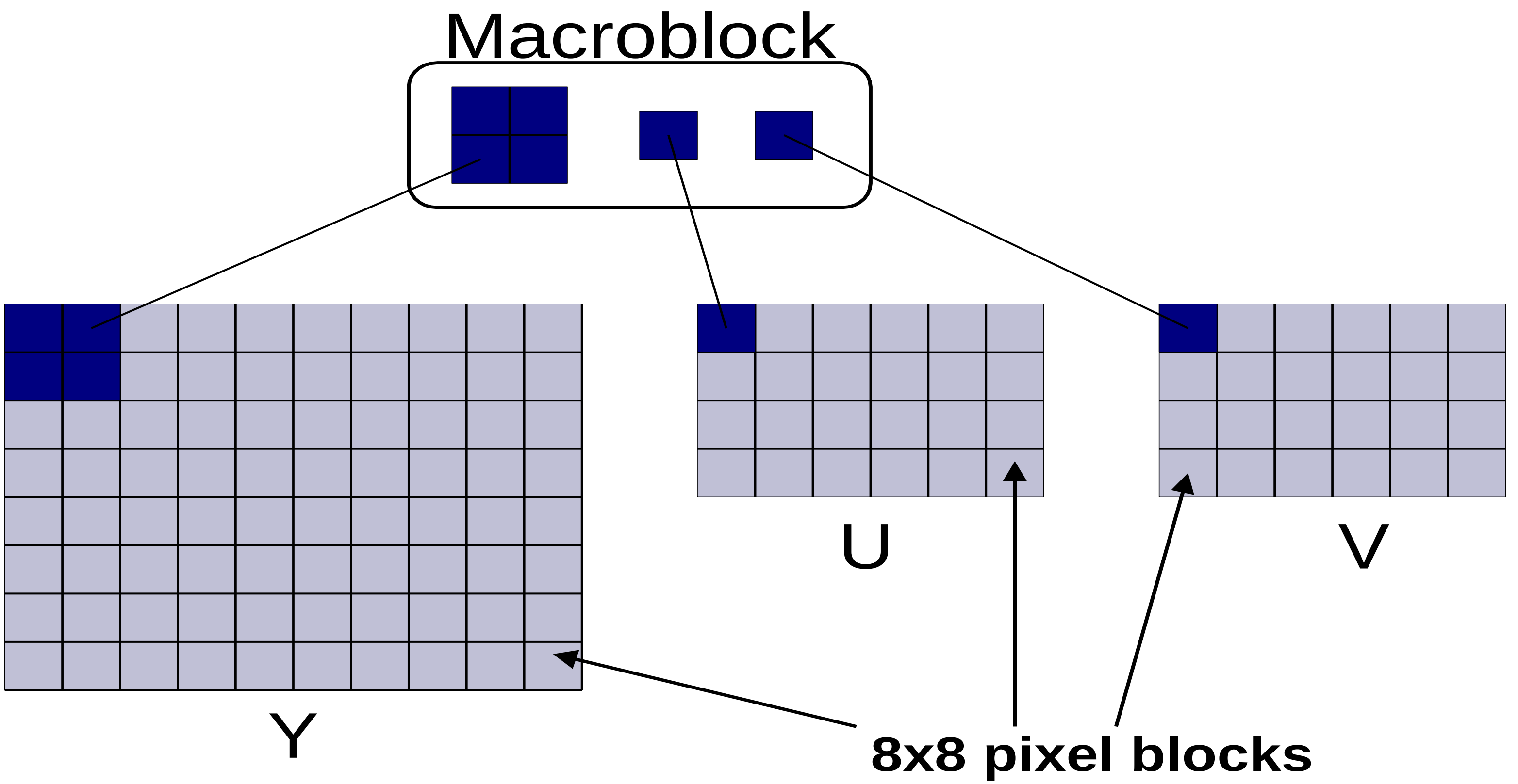


× Represent luminance samples

○ Represent chrominance samples



Blocks and macroblocks



DCT transform

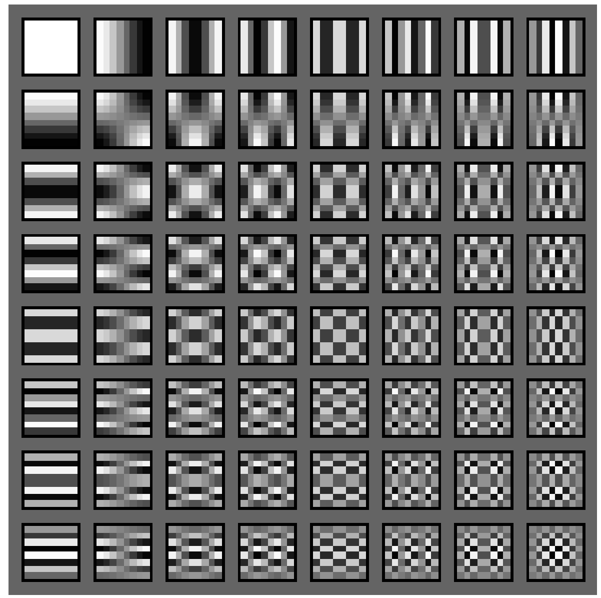
- The Discrete Cosine Transform is an energy-preserving, reversible transform.
- For natural images, DCT helps remove local spatial redundancy.

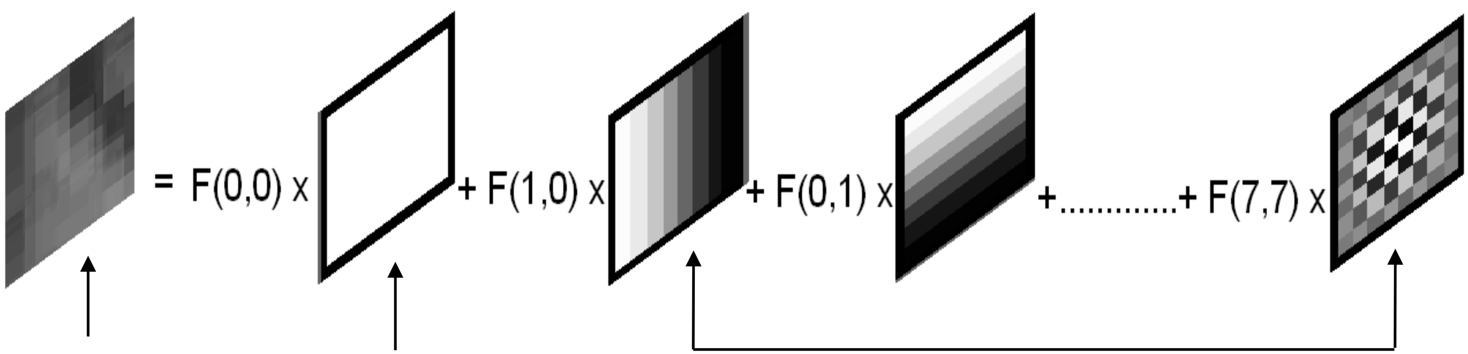
$$F(u,v)=\frac{2}{n}\cdot C(u)\cdot C(v)\cdot\sum_{k=0}^{n-1}\sum_{l=0}^{n-1}f(k,l)\cdot\cos\left[\frac{(2k+1)\cdot u\pi}{2n}\right]\cdot\cos\left[\frac{(2l+1)\cdot v\pi}{2n}\right]$$

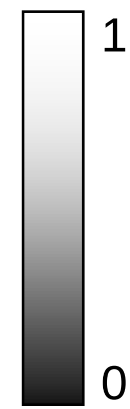
$$f(k,l)=\frac{2}{n}\cdot\sum_{u=0}^{n-1}\sum_{v=0}^{n-1}C(u)\cdot C(v)\cdot F(u,v)\cdot\cos\left[\frac{(2k+1)\cdot u\pi}{2n}\right]\cdot\cos\left[\frac{(2l+1)\cdot v\pi}{2n}\right]$$

where: $$C(w)=\begin{cases}\frac{1}{\sqrt{2}} & for\ \ w=0\\ 1 & otherwise\end{cases}$$

DCT basis functions







Original image block

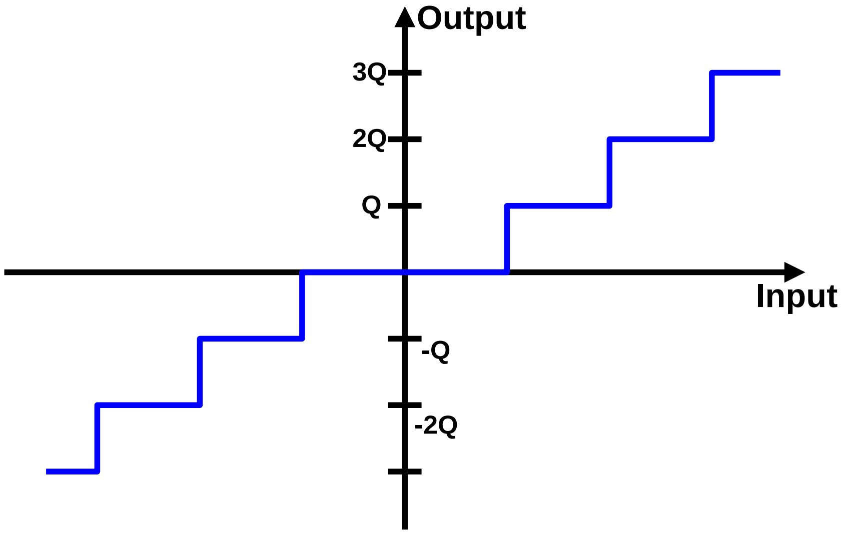
DC (flat) basis function

AC basis functions

Quantization

- A many-to-one mapping.
- Reduces the number of possible signal values at the cost of introducing errors.
- Quantizer step size selection controls the trade off between image quality and bit rate.

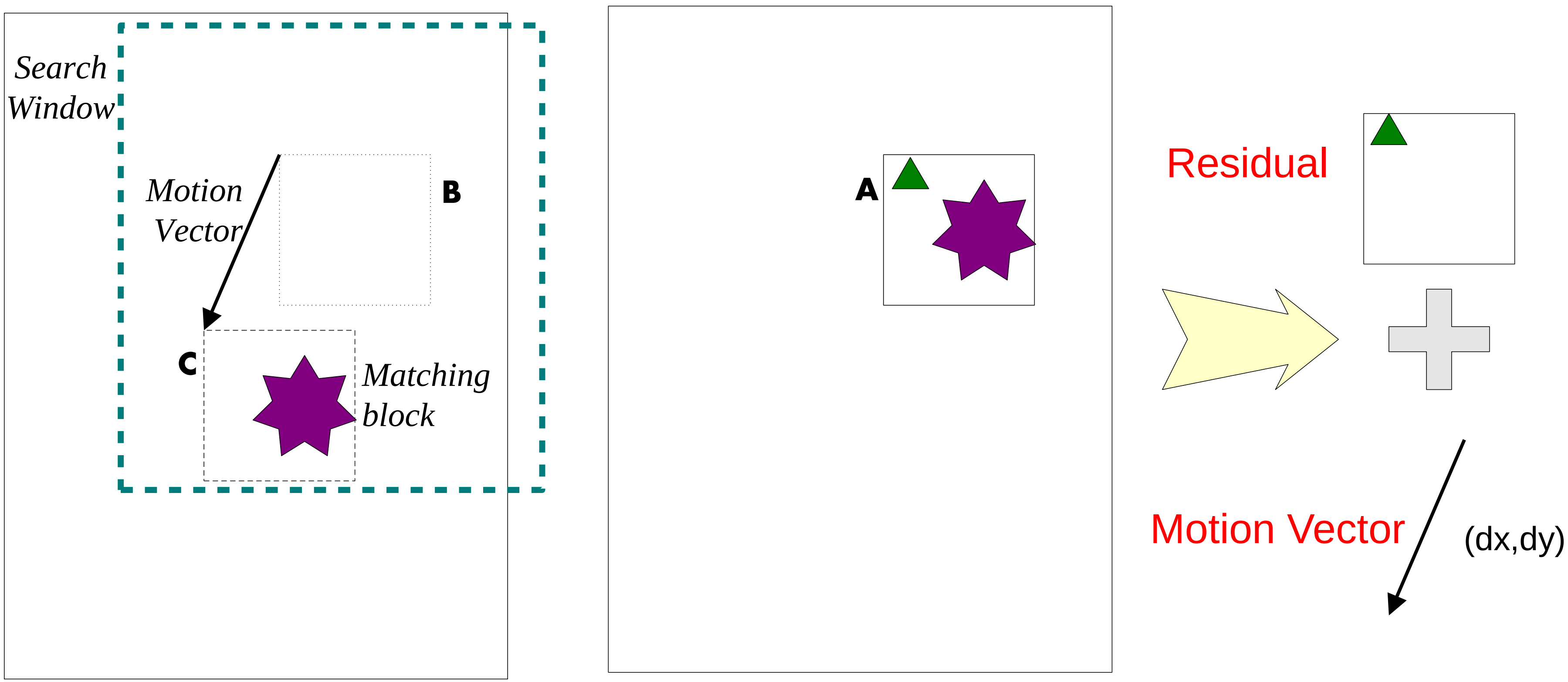
Uniform quantizer function:



Motion estimation

SAD example

- Estimates the displacement of image structures from one frame to another, not necessarily true motion

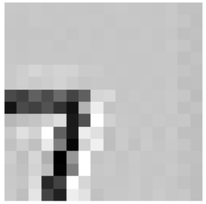


Reference frame Current Frame

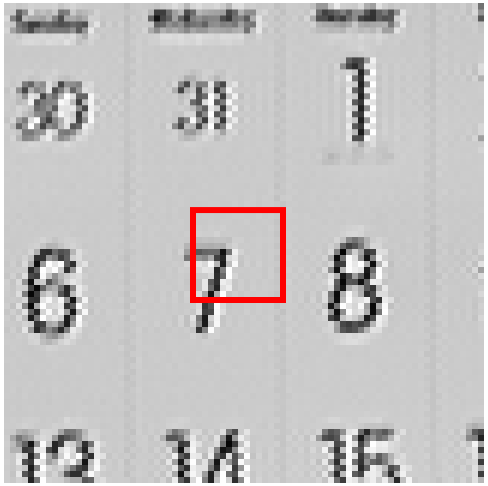
Motion Estimation - Example

Goal: search for minimum Sum of Absolute Differences

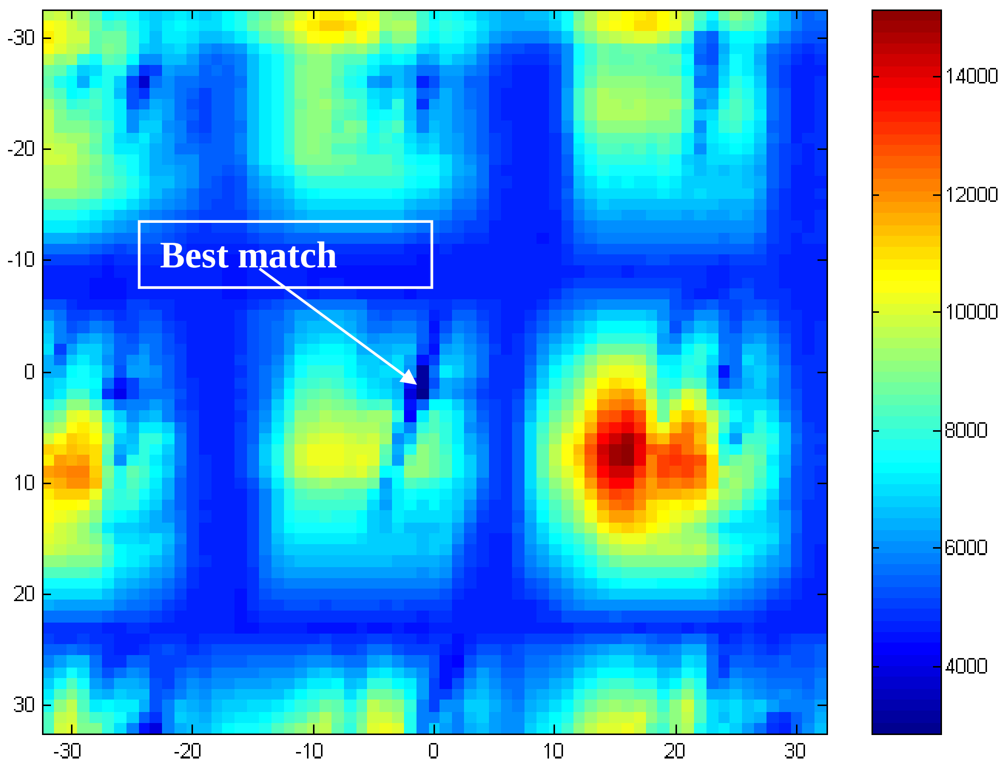
Current block



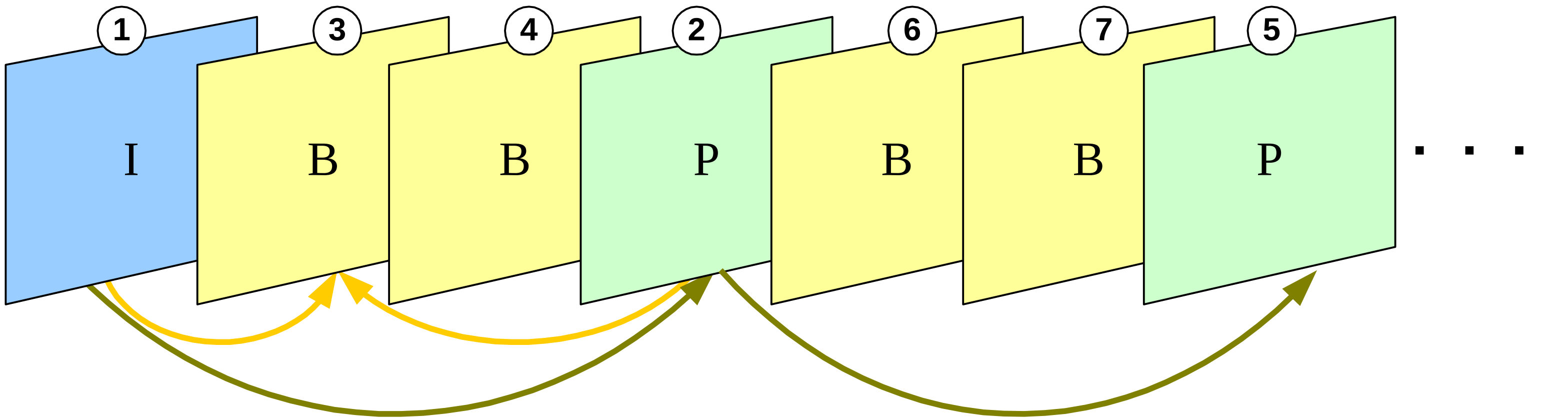
Reference region



SAD map



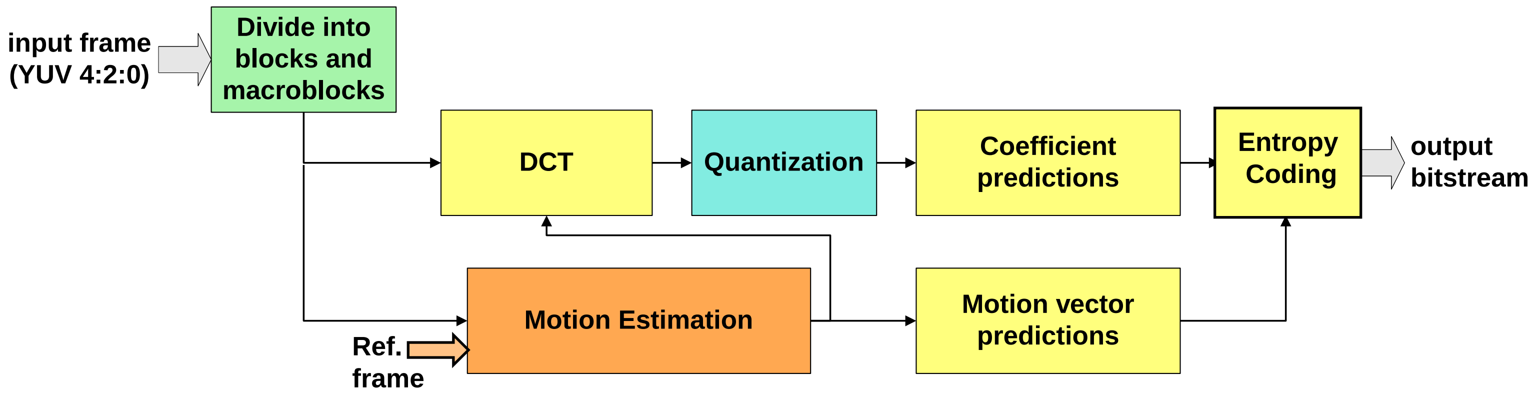
Frame types



- The selection of macroblock and frame types is performed by the encoder rate control module

	INTRA MB	INTER MB	SKIPPED MB
I frame	√	-	-
P/B frame	√	√	√

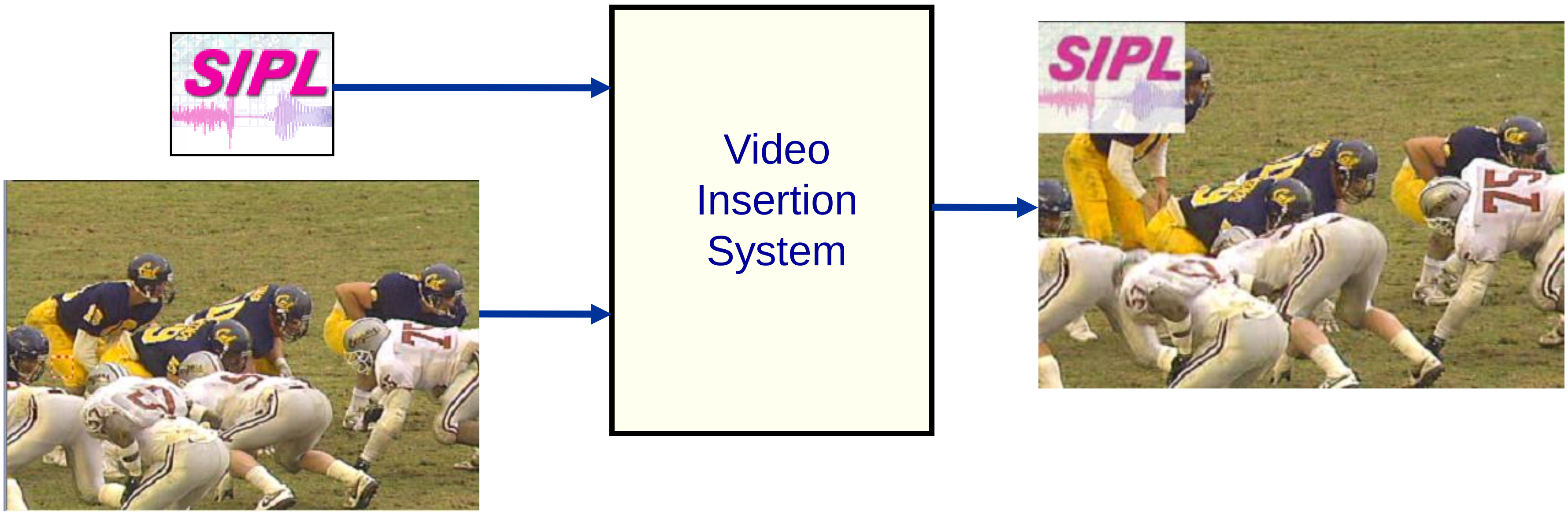
video coding - summary



- The loss is introduced by the quantizer.
- The top branch removes spatial & statistical redundancies and irrelevancies.
- The bottom branch removes temporal redundancy.
- The decoder performs the inverse operations to receive the reconstructed video sequence.

Content Insertion Algorithms

Content Insertion Algorithm



Content Insertion Challenges

- Segmentation: Distinguishing between affected ('variable') areas and unaffected ('constant') areas.

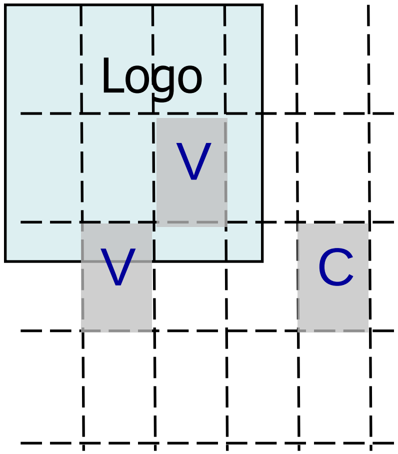
- Efficient handling of unaffected areas (non trivial due to predictions & entropy coding).

- Seamless content insertion in affected area.

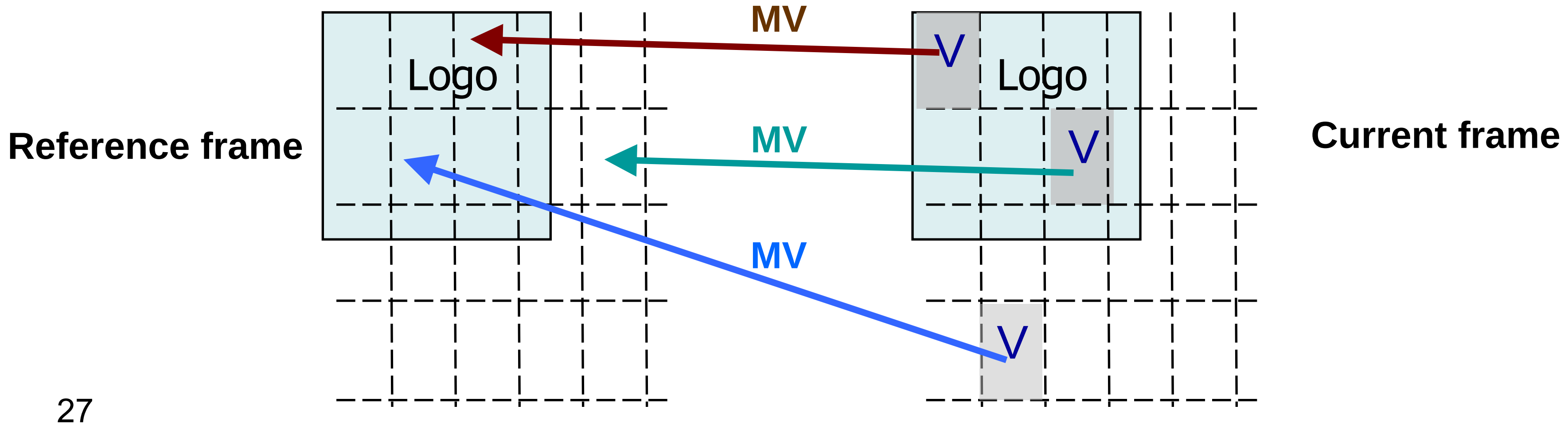
Important: The "logo" is NOT aligned to the MB grid !

'Constant' and 'Variable' Blocks

INTRA:



INTER: (additional Variable blocks due to motion)



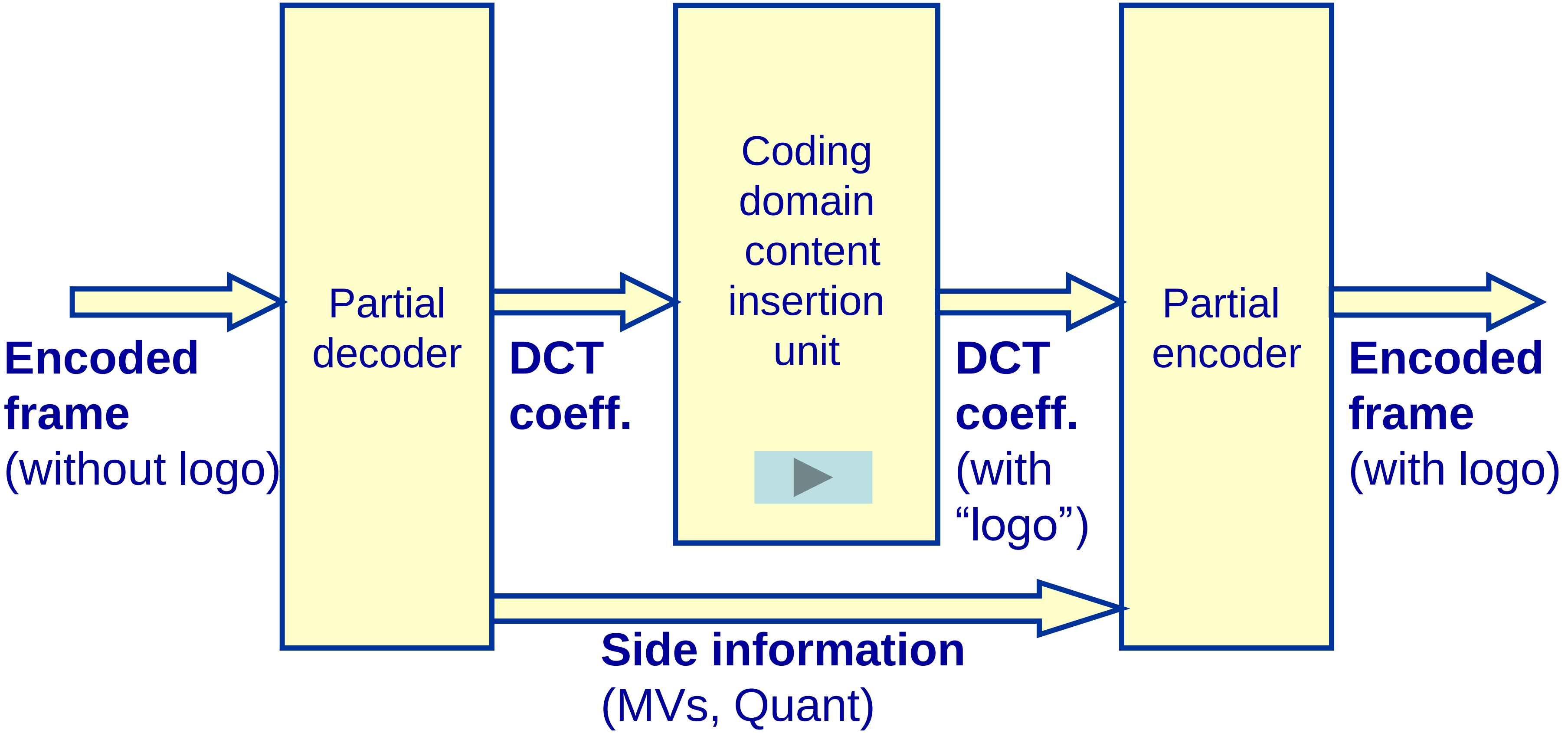
‘Constant’ and ‘Variable’ Blocks

Example: - 'constant’ and ‘variable’ blocks map

White – ‘variable’ blocks

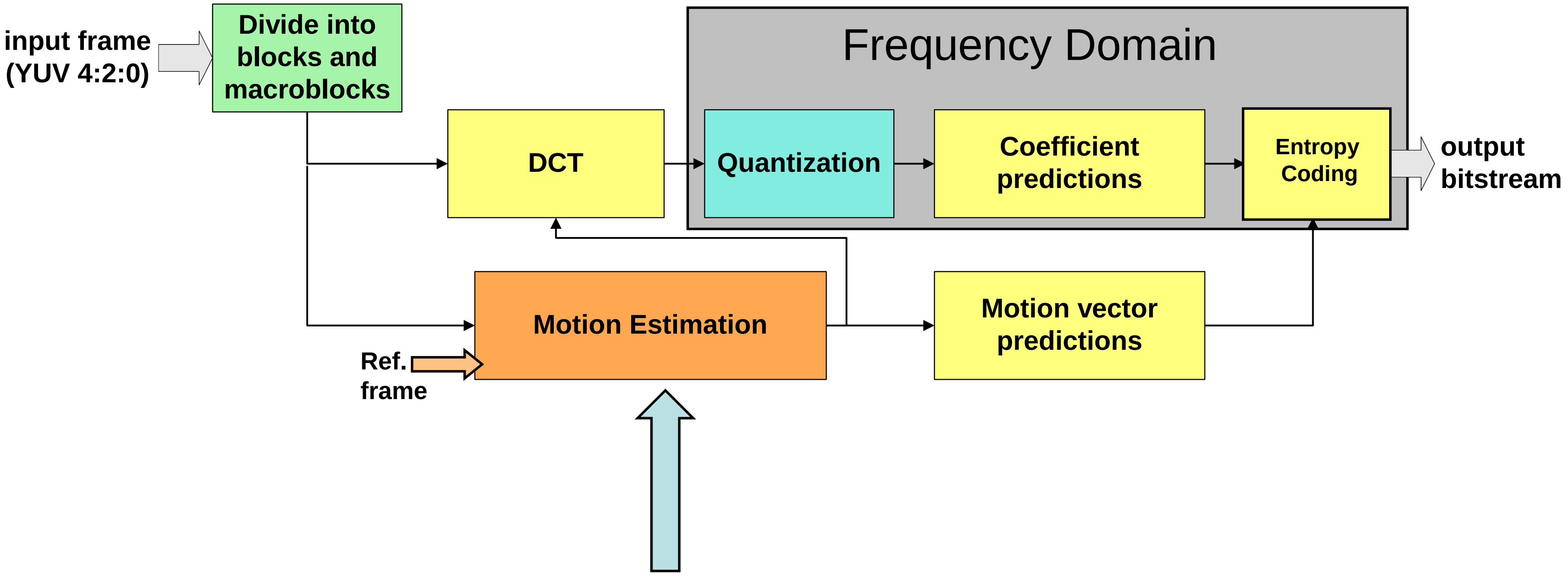
Black – ‘Constant’ blocks

Coding Domain Logo Insertion



MC-DCT Motivation

MC-DCT: Motion Compensation in the DCT domain



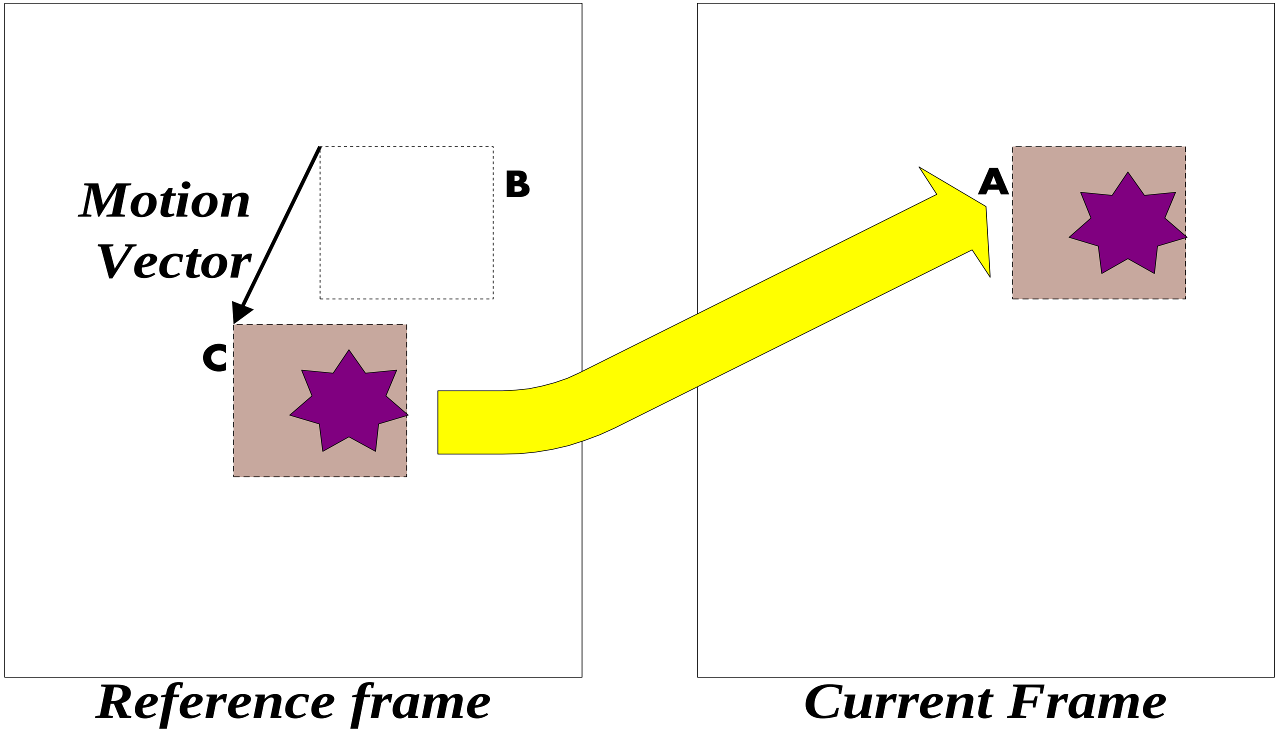
"Not natural" in the frequency domain

MC-DCT properties

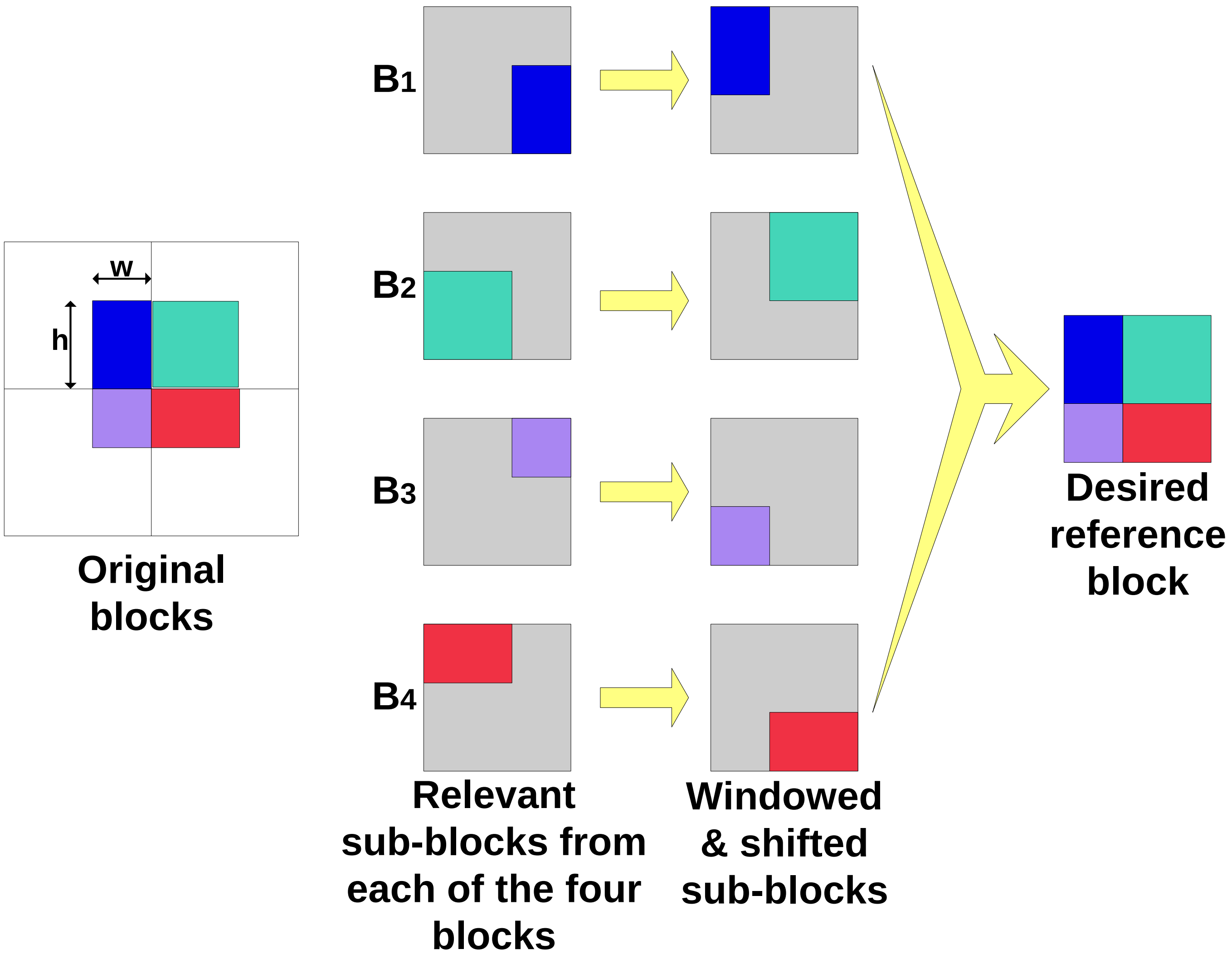
- Saves the IDCT-DCT operations. ☺
- Pixel and DCT motion compensation results are equivalent.
- MC-DCT complexity is higher but may only be required in affected areas.
- Enables other DCT domain operations such as DCT domain resize. ☺

Pixel motion compensation

Retrieve reference block, pointed to by motion vector, from reference frame:



Retrieval of Un-aligned block



Pixel MC: formalization

- A sub-block is obtained from an original block by pre & post multiplication with window/shift matrices, as given by: $S_i = H_i B_i V_i, \quad i = 1, \ldots, 4$
- The windowing/shifting matrices are defined as:

$$H_1 = \begin{bmatrix} 0 & I_h \\ 0 & 0 \end{bmatrix}, \quad V_1 = \begin{bmatrix} 0 & 0 \\ I_w & 0 \end{bmatrix} \qquad H_2 = \begin{bmatrix} 0 & I_h \\ 0 & 0 \end{bmatrix}, \quad V_2 = \begin{bmatrix} 0 & I_{8-w} \\ 0 & 0 \end{bmatrix}$$

$$H_3 = \begin{bmatrix} 0 & 0 \\ I_{8-h} & 0 \end{bmatrix}, \quad V_3 = \begin{bmatrix} 0 & 0 \\ I_w & 0 \end{bmatrix} \qquad H_4 = \begin{bmatrix} 0 & 0 \\ I_{8-h} & 0 \end{bmatrix}, \quad V_4 = \begin{bmatrix} 0 & I_{8-w} \\ 0 & 0 \end{bmatrix}$$

- The required block is: $\hat{B} = \sum_{i=1}^{4} S_i$

Pixel MC - example

example of extracting the bottom right 3x2 area from a 4x4 matrix:

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 11 & 12 & 13 & 14 \\ 21 & 22 & 23 & 24 \\ 31 & 32 & 33 & 34 \\ 41 & 42 & 43 & 44 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} =$$

$$\begin{bmatrix} 21 & 22 & 23 & 24 \\ 31 & 32 & 33 & 34 \\ 41 & 42 & 43 & 44 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 23 & 24 & 0 & 0 \\ 33 & 34 & 0 & 0 \\ 43 & 44 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

DCT domain MC

DCT , an orthogonal transform, is distributive to matrix multiplications : $DCT(AB) = DCT(A)DCT(B)$

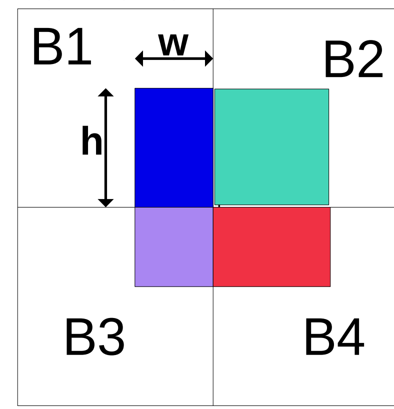
Therefore:

$$DCT(\hat{B}) = \sum_{i=1}^{4} DCT(H_i B_i V_i) = \sum_{i=1}^{4} DCT(H_i) DCT(B_i) DCT(V_i)$$

- DCT of the manipulation matrices is performed off-line, and stored for each possible offset.

Performing DCT domain MC

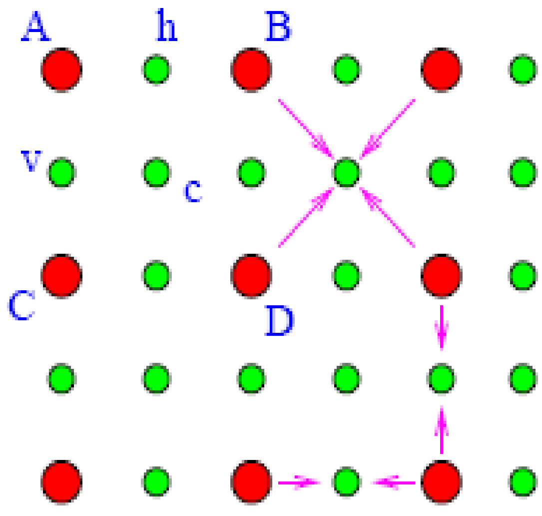
1. Calculate w and h from the motion vector.
2. If the block is aligned (w=h=0), the desired DCT coefficients are immediately available.
3. Otherwise, for:
   - ($w = 0$) & ($h \neq 0$) get S1 and S3,
   - ($w \neq 0$) & ($h = 0$) get S1 and S2,
   - ($w \neq 0$) & ($h \neq 0$) get S1, S2, S3 and S4.



4. Perform matrix multiplications & summations.

Half-Pixel Motion Compensation

Half pixel resolution motion vectors require retrieval of an interpolated block from the reference frame.



Horizontal Interpolated pixel

$h = (A+B)/2$

Vertical interpolated pixel

$v = (A+C)/2$

Central interpolated pixel

$c = (A+B+C+D)/4$

Half pel resolution MC-DCT

The windowing/shifting/interpolation matrices for retrieving a half-pel resolution reference block are of the form:

$$Hhp_1 = \frac{1}{2}\left\{\begin{bmatrix} 0 & I_h \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & I_{h-1} \\ 0 & 0 \end{bmatrix}\right\}, \quad Vhp_1 = \frac{1}{2}\left\{\begin{bmatrix} 0 & 0 \\ I_w & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ I_{w-1} & 0 \end{bmatrix}\right\}$$

Where, w & h are ceil() of the half pixel motion vectors, (8 possible values).

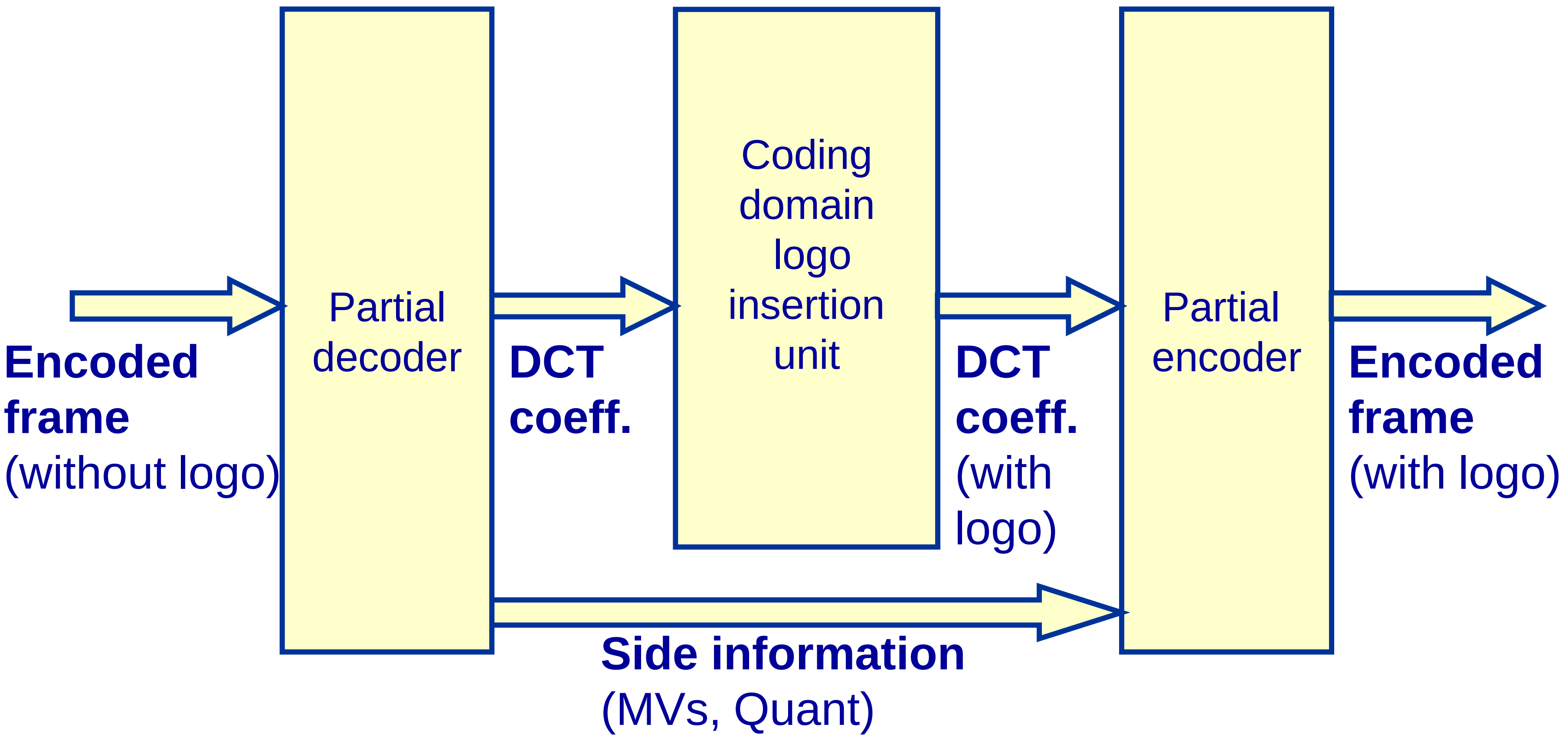
Half pel resolution MC-DCT

- The required interpolated motion compensated block in the DCT domain is given by:

$$DCT\left(\hat{B}_{hp}\right) = \sum_{i=1}^{4} DCT\left(Hhp_i\right) DCT\left(B_i\right) DCT\left(Vhp_i\right)$$

- The calculation steps are identical to those of the full-pel case.

Compressed Domain Logo Insertion



THANK YOU

H.264 compression novelties

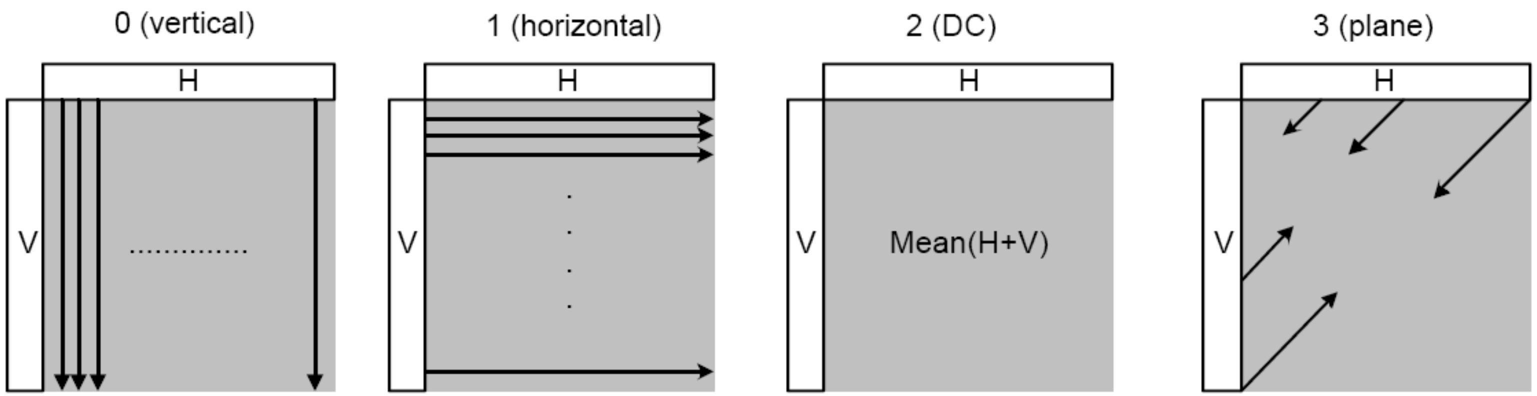
H.264 main innovations - 1

- 4x4 basic block size, with integer transforms (ICT).
- Tools for improving temporal prediction:
  - Multiple reference frame mechanism.
  - Large variety of block size and shapes for motion compensation.
  - 1/4 pixel resolution motion vectors combined with high quality interpolation filters.
  - Advanced prediction of MVs between adjacent blocks.

H.264 main innovations – 2

Spatial prediction

- INTRA block content is predicted from neighbors
- Removes redundancies between adjacent blocks
- Performed in the pixel domain
- MANY different prediction modes supported



4x4 luma prediction modes

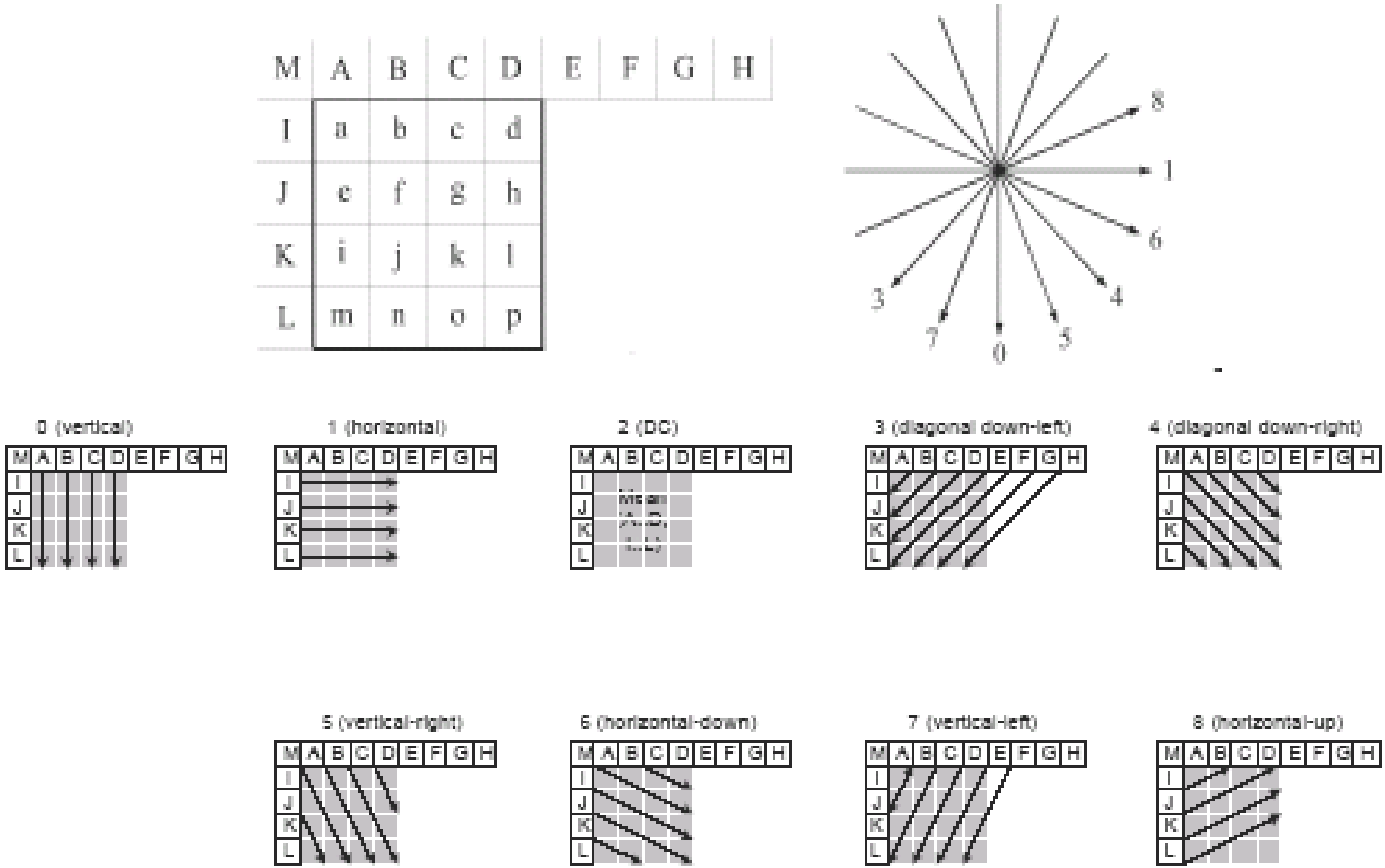


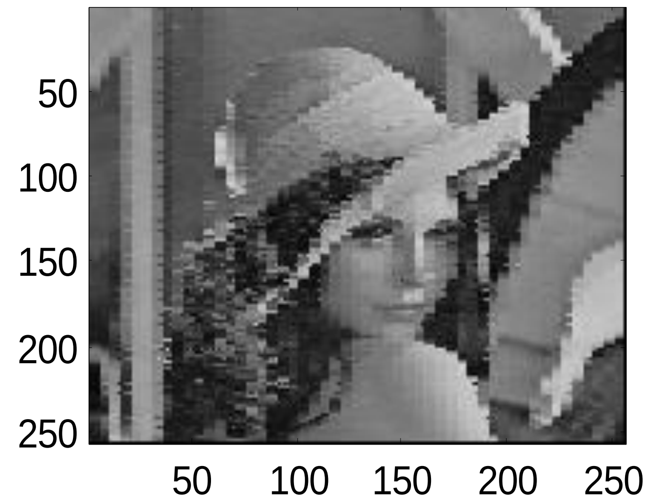
Figure 3 4x4 luma prediction modes

INTRA prediction example

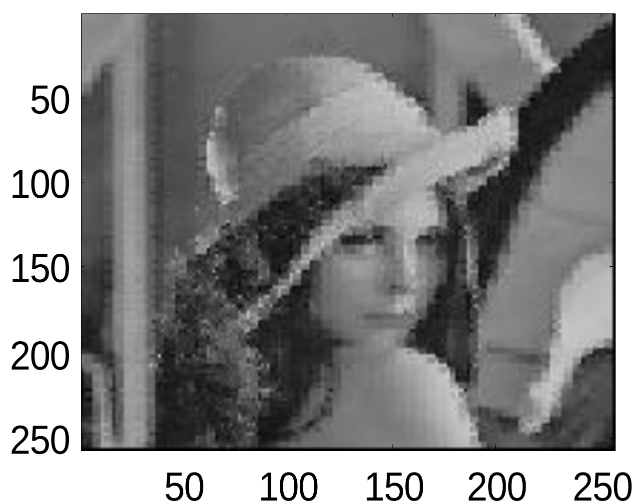
Lena

Vertical Prediction

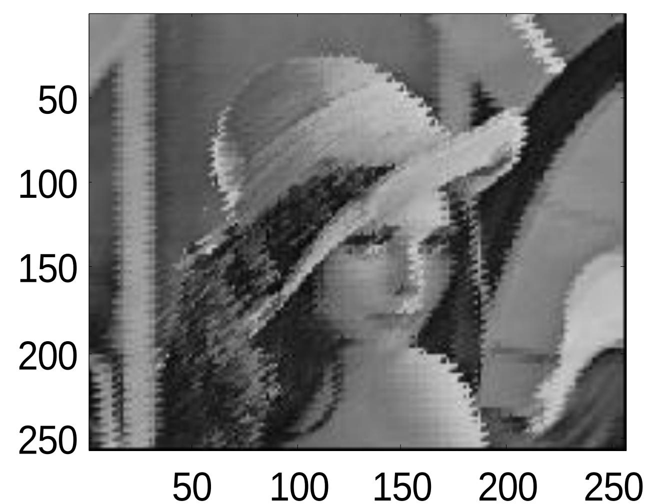
Horizontal Prediction



DC (mean) Prediction



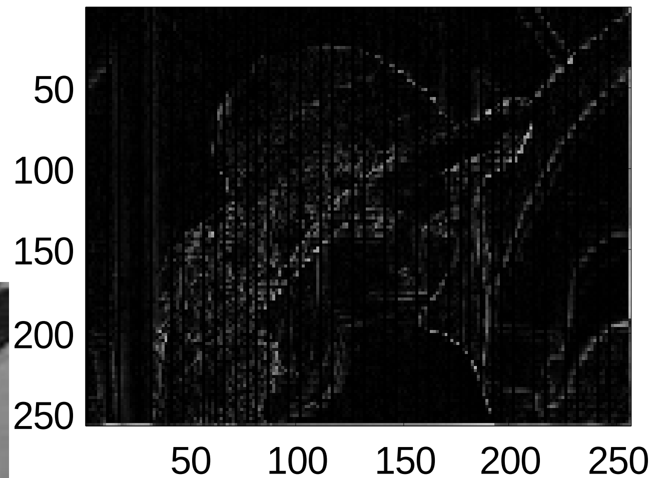
Diagonal Down Left Prediction



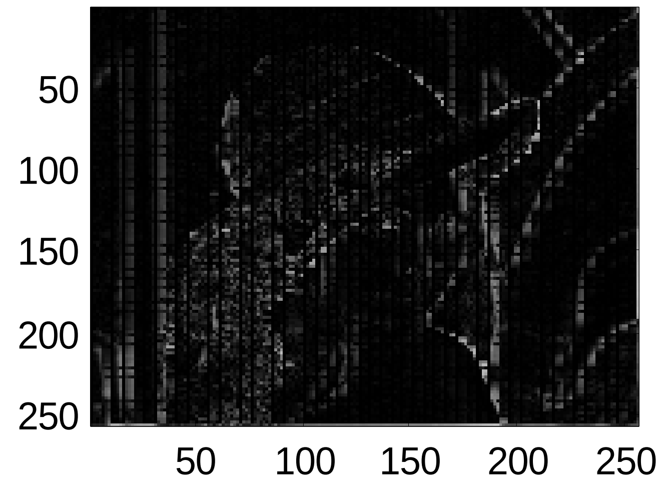
INTRA prediction example – cont.

Lena

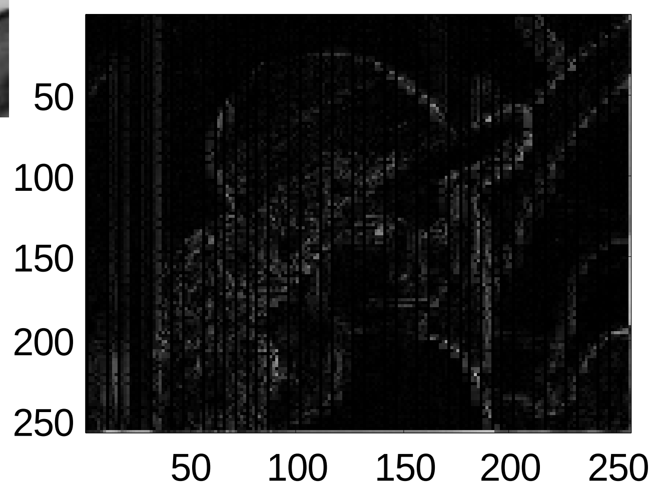
Vertical Prediction Error



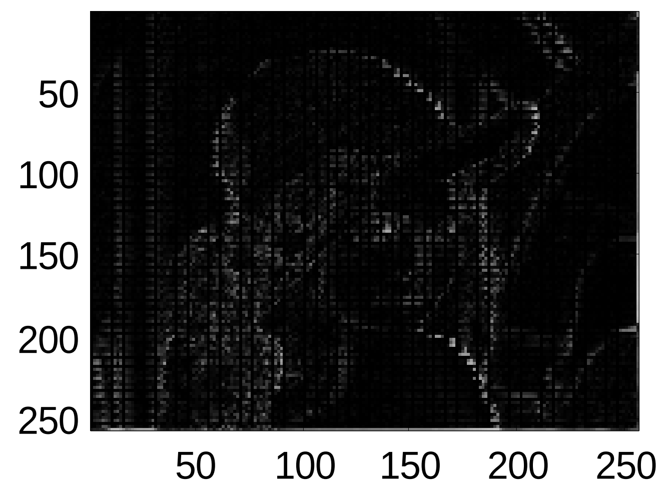
Horizontal Prediction Error



DC (mean) Prediction Error



Diagonal Down Left Prediction Error



H.264 main innovations – 3

- Efficient context-adaptive entropy coding.
- In loop deblocking filter

Figure 4
Performance of the deblocking filter for highly compressed pictures.
Left: without the deblocking filter. *Right:* with the deblocking filter.

Additional Video in video Challenges in H.264

Challenges - 1

- Advanced spatial predictions complicate affected/unaffected segmentation.
  - INTRA frames: Logo area prorogates via intra prediction to the entire frame.
  - INTER frames: MVs are spatially predicted, therefore local MV changes propagate through the frame.

Challenges - 2

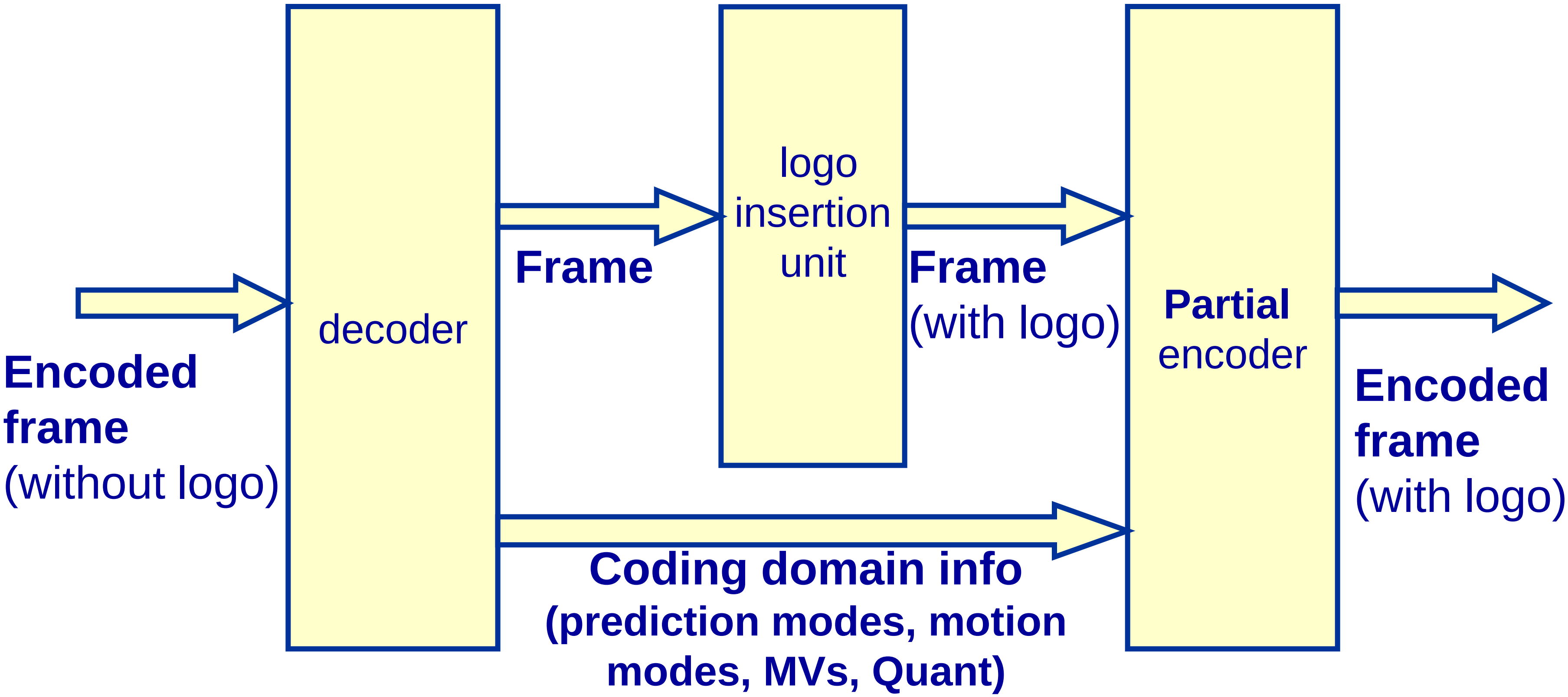
- MC-ICT, the integer transform equivalent to MC-DCT must be developed and evaluated.
- The many motion modes and ¼ pel resolution complicate the MC-ICT and require re-evaluation of its profitability.
- Effect of in-loop deblocking filter must be evaluated.
- Sophisticated context-adaptive entropy coding causes any change to affect the coding of the entire frame.

Video in Video – H.264

- Motion estimation remains the most “expensive” part of encoding.
- Macroblock mode selection requires evaluation of many different modes and consumes a significant part of encoding resources.
- Transform complexity is no longer a video-in-video bottleneck.

⇨ *H.264 Partial encoder should operate in the* ***coding domain***

Coding Domain Logo Insertion



Compressed domain flow

Progress Report

Progress Report

- MPEG-2 static logo insertion – completed.
  (~70-80% reduction in run-time compared to naive solution)

- Static logo insertion into H.264 INTRA Baseline frames – completed.
  (~50% gain in initial tests on Nokia Baseline software)

- Static logo insertion into H.264 INTER Baseline frames – in progress.

- DSP implementation – in progress.