Video Coding...

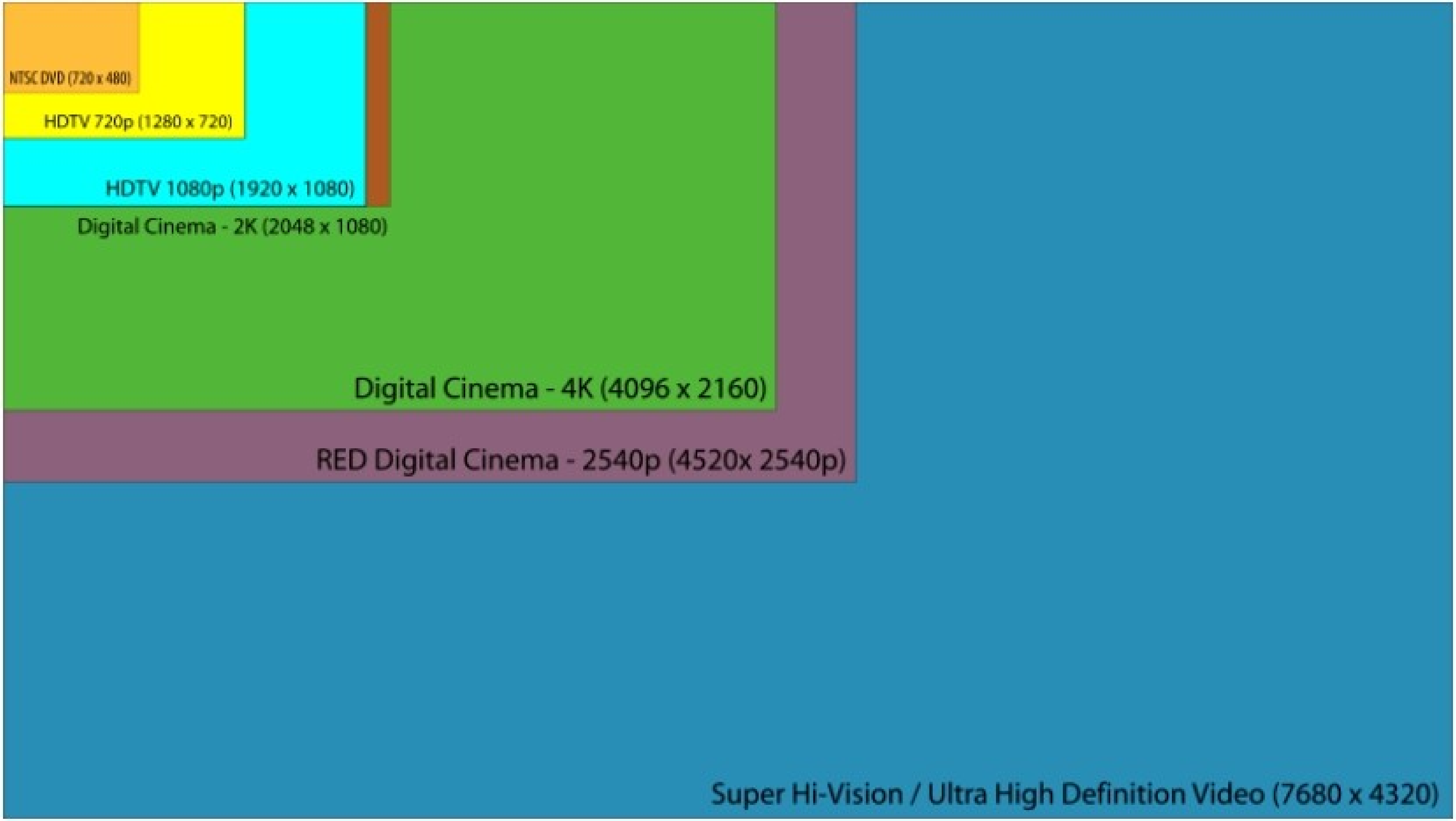
Adi Kouadio - Ing. Dipl. EPFL

Agenda

- Media Landscape…
- Codec Overview
- Emphasis on H.264/AVC
- Patents battles and VP8
- GPU acceleration

- http://www.youtube.com/watch?v=ExCqfVwlnxU

Increasing (TV) resolution...



Different viewing means

VOD

3D

UHDTV

Streaming

HDTV

Video Conferencing

User Generated Content (UGC)

Contribution

Acquisition

Video surveillance

Editing

Mobile (HD)TV

Medical Imagery

data

compression systems

SVC MVC DNxHD(vc2)

MPEG-4

(M)JPEG2000

HEVC

H.264/AVC

MJPEG

VP8

WMV(vc1)

Dirac (vc3)

Theora

MPEG-2

Clarification...

- **MPEG-2** coding (nothing to do with the Transport Stream MPEG2-TS !)
  - Traditional broadcast Broadcast, progressively replaced by H.264/AVC
- **MPEG-4** **Part 2** : (Visual) object coding
- **MPEG-4** **Part 10** / **H.264/AVC** : Advanced video coding.
  - 50% bit-rate gain over MPEG-2.
- **SVC** Scalable Video coding
  - Based on H.264/AVC (core coding) with inter-Layer (spatial, temporal, snr) coding. mainly used in videoconferencing (see www.vidyo.com)
- **MVC** Multiview video coding
  - based on H.264/AVC with Inter-view prediction.

- M

Codec Overview

Video

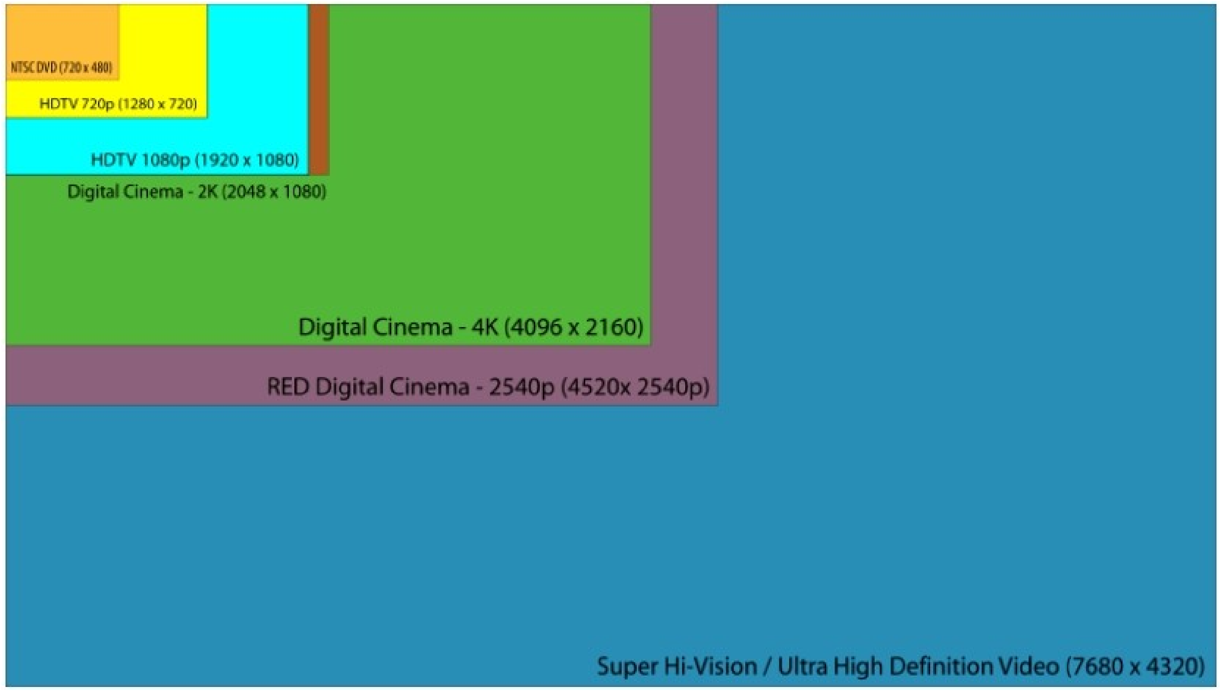
- Succession of images with same resolution taken/shot at regular intervals.
- Characteristics :
  - **Resolution...**
  - **Color Space...**
  - **Bit Depth...**
  - **Frame Rate...**
  - **Scanning mode...**
    - Interlaced (entrelacée)
    - Progressive

Scanning Mode

- **Scanning mode** : describes how pixels were captured
- Interlaced (noted **i** in the format syntax e.g 1080i25)
    - A frame is composed of 2 sub-fields shot at **different** time instant.
        - **odd lines** field
        - **even lines** field
    - clever trick to reduce bandwith and increase frame rate.
- Progressive (noted **p** in the format syntax)
    - All lines in the picture are taken at the same time instant.

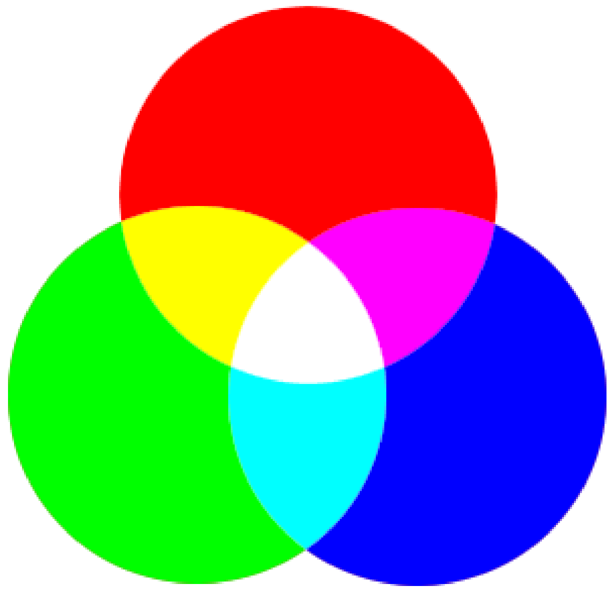
Resolution / bit depth

- **Resolution** = Number of lines vs column in the image = Number of pixels.
  - In HDTV pixels are scared :
    - 1920x1080 (1080i or 1080p)
    - 1440x1080 (usually1080i)
    - 1280x720 (720p...)
  - In SDTV pixels could be rectangular!
- Not to mix with aspect ratio (4:3 or 16:9)
- **Bit-depth** = number of representation bits for each pixel.
  - Responsible (partly) for color tone reproductions and contouring...



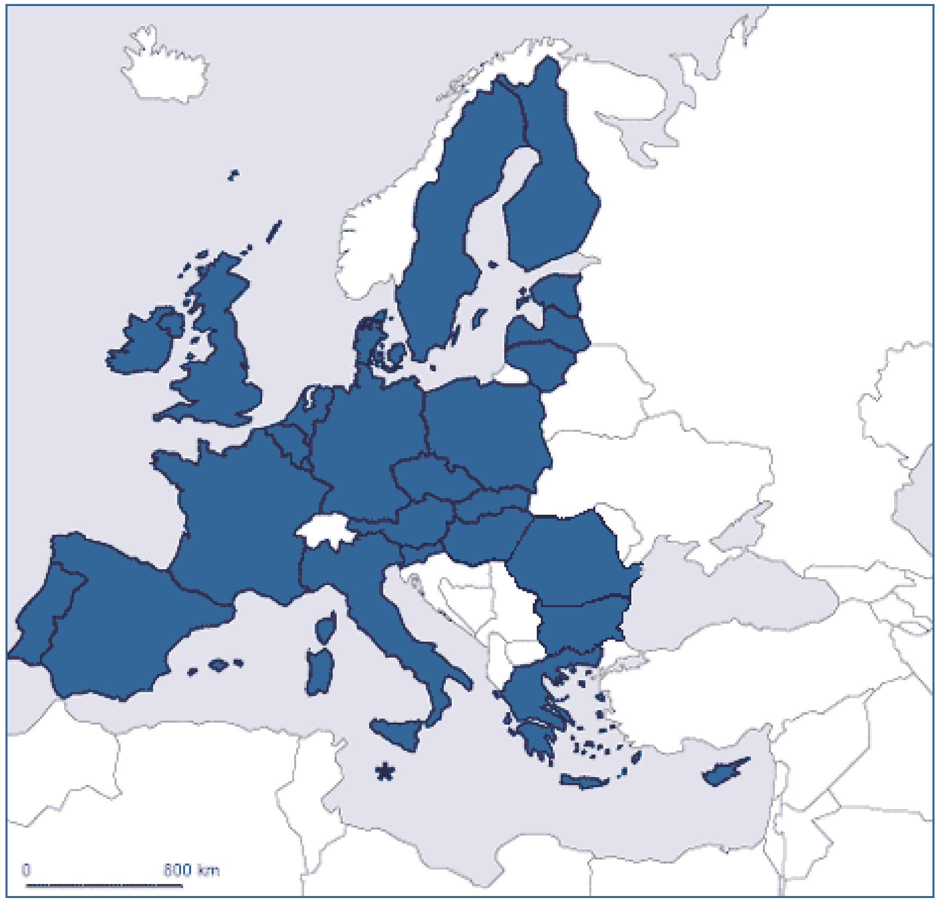
Colour space

- An image is an array of 3 or more layers/channel with equal bit depth.
    - **1920x1080x3 = #samples in image**
- In general, the coloured images are in **RGB** color space.
    - ITU Rec 601 specify color primaries values for SDTV
    - ITU Rec 709 specify color primarie and functions for HDTV
- **but for coding efficiency...**
    - colour **differences** are used to reduce correlation between channels
        - YCbCr = luminance, Red color diff and Blue color diff.
    - color channels are down-sampled (chroma sampling) less human sensitivity.
        - 4:2:2 ==> 2 Y samples for 1 Cr and 1 Cb => Y-Cr-Y-Cb (Professional)
        - 4:2:0 ==> 4 Y samples for 1 Cr and 1 Cb (consumer)



Frame Rate

- Number of **frames (not field)** per second.
- Higher frame rate provides better motion portrayal (higher ***temporal resolution***)
    - 25 / 50Hz content in EU
    - 30 / 60Hz content in US / JP

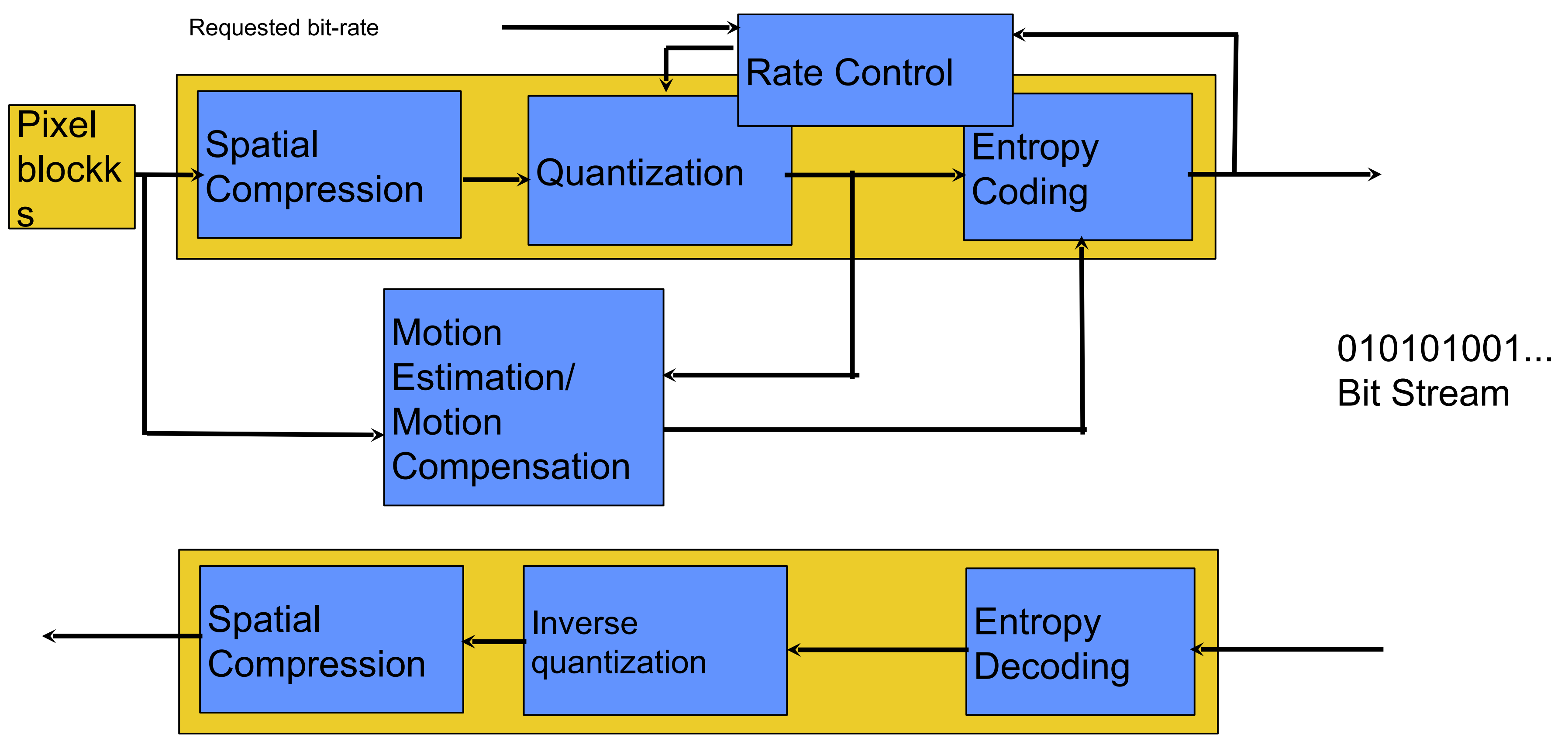


Constraints..

- Depends on the application
  - **Complexity...**
    - **Asymmetric codecs**
      - high compexity in encoder, lower/light in decoder (broadcast, mobile TV, web etc). e.g. MPEG Family.
    - **Symmetric codecs**
      - Applies in only very specific cases (JPEG2000 in contribution networks)
  - **Coding efficiency...**
    - Provide the best quality at the lowest bit-rate possible using of course lightweight tools and low memory.

Compression…

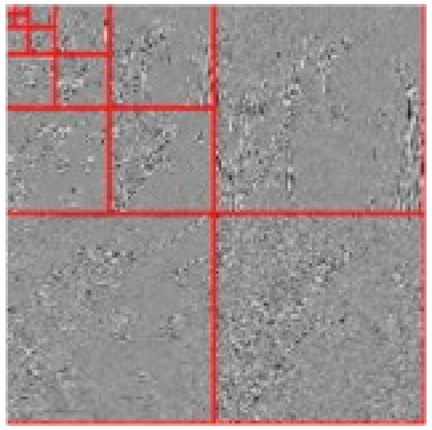
A typical codec...



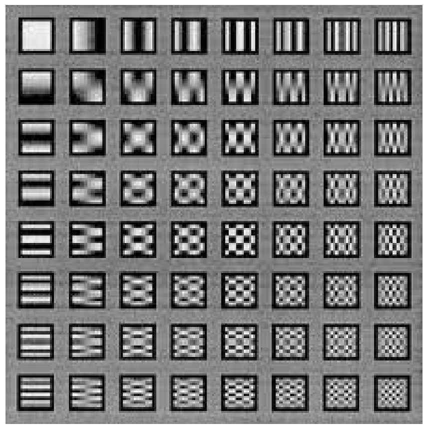
Process…

- User defines output bit-rate, encoding structure etc.
  - helps **rate control** and transmission **buffer size**.
- select frame, **divide into macroblocks.**
- locate similar macroblocks and reduce redundancy by using **predictions.**
- separate chroma and luma of each block.
- apply a **transformation** to each macroblock.
- **quantize** resulting coefficients.
- reduce statistical redudancy by applying **entropy coding**.
- store coded bitstream into an **elementary stream (ES).**

...Spatial



- Removes the spatial redundancy available in spatially adjacent pixels
- Use a Spatial Frequency Transform
  - Discrete Cosine Transform - DCT (MPEG , JPEG)
    - ➢ operates on blocks of pixels (Macroblocks)
    - ➢ decomposes the spatial frequencies into the sum of
  - Discrete Wavelet Transform - DWT (JPEG 2000)
    - ➢ Series of low and high pass filters
    - ➢ operates on the whole picture.



- Transform results are set of coefficients to be quantized

...Quantization

Quantization parameter (QP)

« The place where we loose information »

- Reduces the representation precision of data samples
  - **Scalar quantization**
  - **Variable quantization**
  - **Weighted quantization**
    - according to human perception models
    - The smaller the quantization step the better the fidelity...

Rate control

« controls that the number of bits used match the required/requested bit-rate»

- defines bit budget for a GOP based on target bit-rate and buffer occupancy
- Not standardised in MPEG
- Only non-normative guidance available
  - §examples available in Test model softwares.

...Predictions

Two Types :

- Intra (Spatial) Prediction (The frame is self-compressed).
- Inter (temporal) Prediction (use information from preceding and future frames)

Intra prediction use :

- Previously neighbouring coded macroblocks as potential estimates.

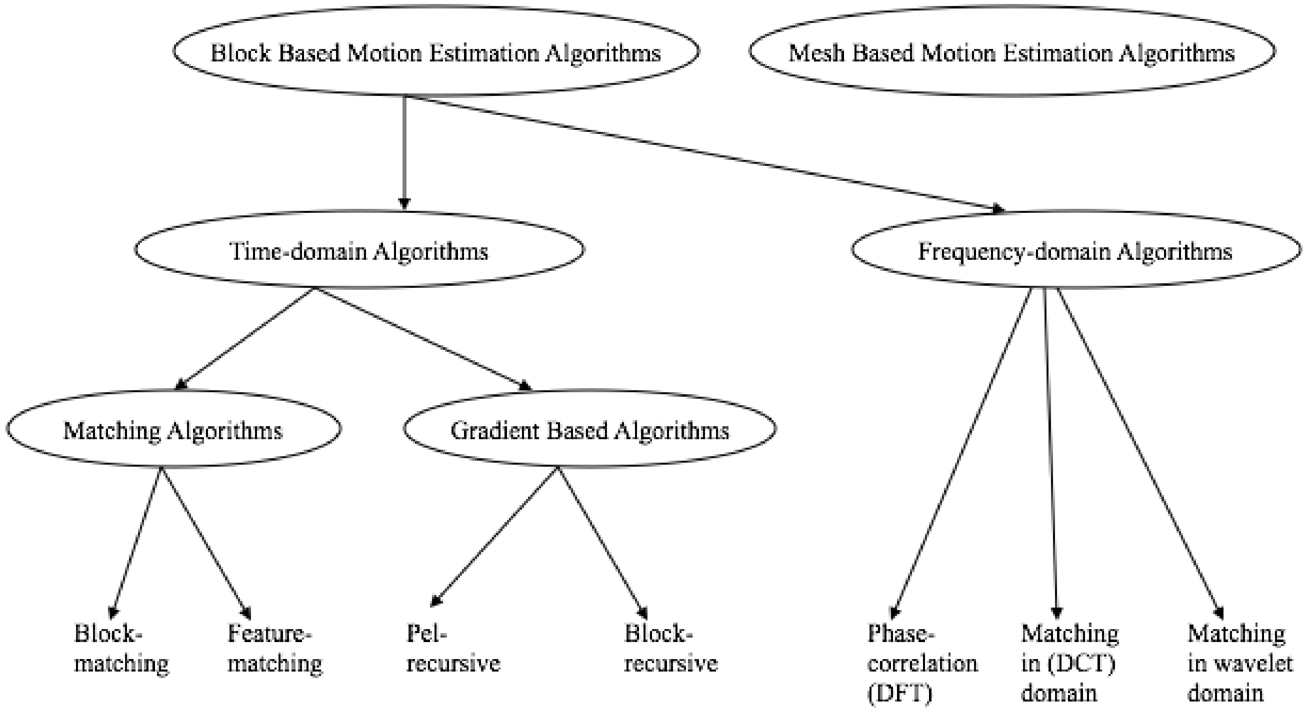
Inter prediction use :

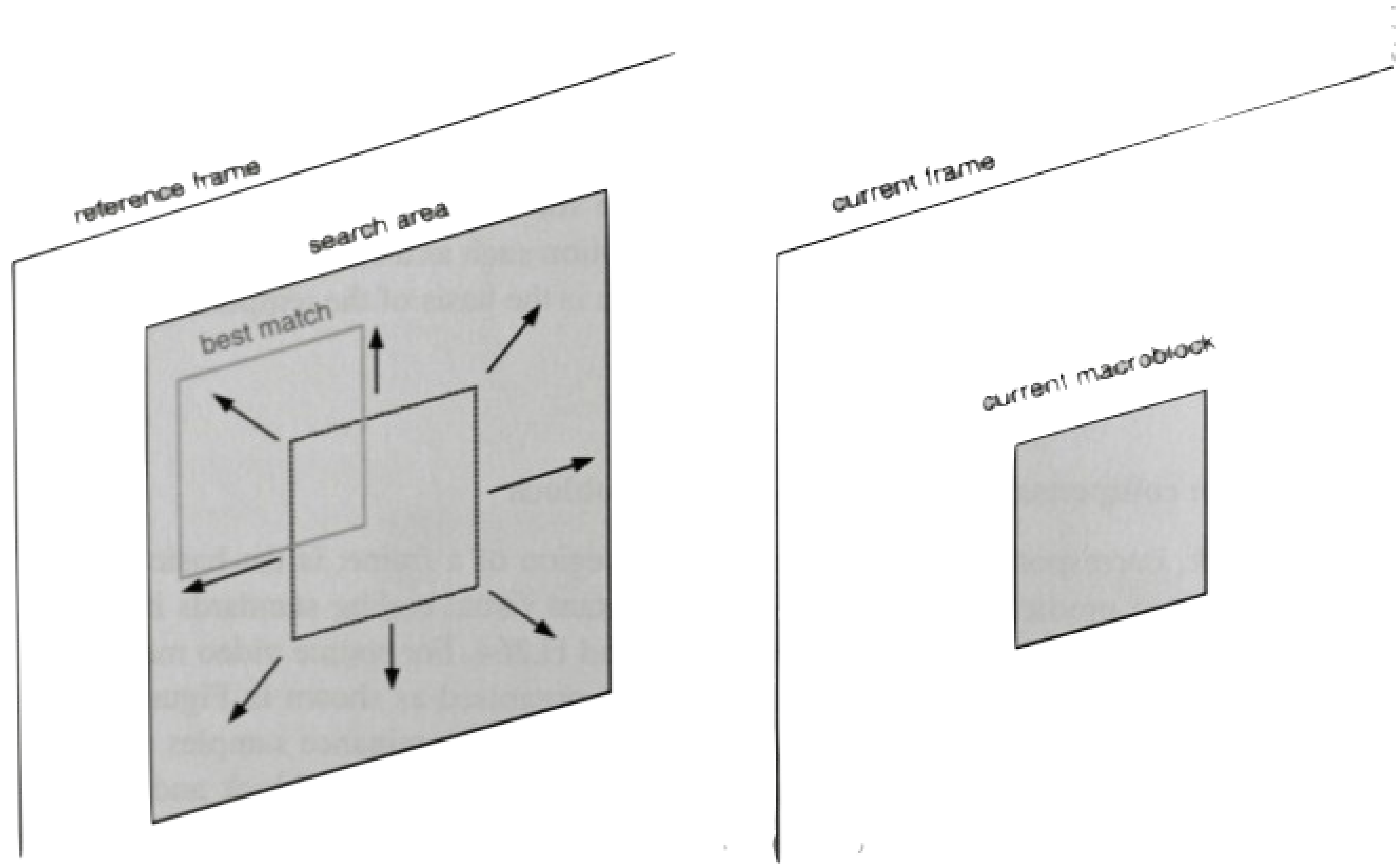
- Motion Estimation
  - Uses the previous / next frame to be predicted
    - Find the best match in a search region for a set of pixels.
  - Find the pixel displacement accross
    - Create a motion vector
- Motion Compensation
  - Corrects/compensate the motion prediction errors from the estimate (best match).
    - current macroblock substracted from best match macroblock -> Residual
    - relative best match coordinates correspond to the motion vector.
    - <u>Residual + motion vector are coded and transmitted</u>

Motion Estimation...

Several techniques exists :

- MPEG codecs use block based approach.
- Try to find the best match in previously decode frames
- Define a search area
  - usually neighboring blocks
- Compute difference
  - Residuals.
- Calculate displacement
  - MVDs - motion vectors.




reference frame
search area
best match
current frame
current macroblock

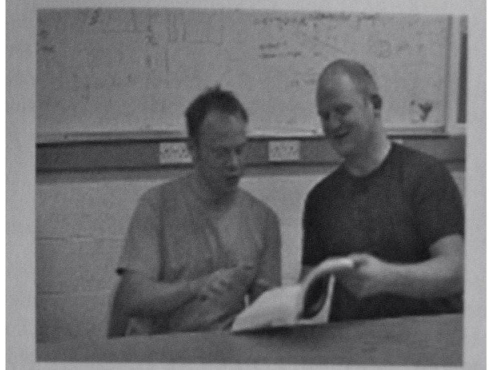


Figure 3.10 Frame 1

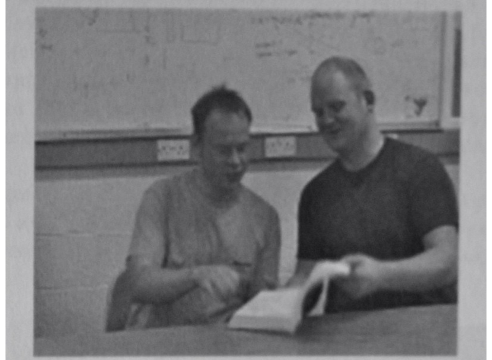


Figure 3.11 Frame 2

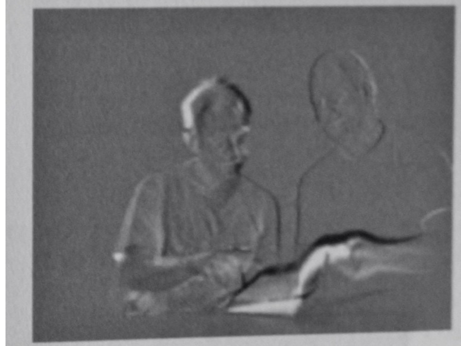


Figure 3.12 Residual : no motion compensation

CODEC - Entropy coding

Removes statistical redundancy in the provided samples

- Replaces data elements with coded representations
  - set of predefined codes stored in tables.
  - set of codes derives from the input data statistic
- Different entropy coders exists
  - Lempel Ziv Coding
  - Exp Golong Coding
  - **(Context adaptive) Variable Length Coding**
    - **Huffmann coding**
  - **(Context adaptive) Binary Arithmetic Coding (CABAC)**
    - **More efficient than Run lenght encoder**

CODEC - Entropy Coding - Huffman Coding

Used in MPEG-2, Theora

- established tree structure codes (binary strings)
  - short strings for most probable values
  - longer string for less probable values
- Codes are stored in the encoder and decoder in look up tables
  - Deterministic and decreases the complexity
  - does not take full advantage of the content specifics.

CODEC - Entropy Coding - Variable Length Coding (VLC)

Can be context adaptive (CAVLC)

- comparison with surrounding pixels codewords.

Coding procedure :

- Predefined codewords (tables) stored in the encoder and decoder
  - Does not take advantage of statistical redundancy in the content
- Mapping done according to DCT coefficient binary representation
  - No live computation of codes -> Low complexity.

CODEC - Entropy Coding -Variable Length Coding

4x4 block:

0	3	-1	0
0	-1	1	0
1	0	0	0
0	0	0	0

Reordered block:
0,3,0,1,-1,-1,0,1,0…

TotalCoeffs = 5 (indexed from highest frequency [4] to lowest frequency [0])
TotalZeros = 3
T1s = 3 (in fact there are 4 trailing ones but only 3 can be encoded as a "special case")

Encoding:

Element	Value	Code
coeff_token	TotalCoeffs=5, T1s=3	0000100
T1 sign (4)	+	0
T1 sign (3)	-	1
T1 sign (2)	-	1
Level (1)	+1 (use Level_VLC0)	1
Level (0)	+3 (use Level_VLC1)	0010
TotalZeros	3	111
run_before(4)	ZerosLeft=3; run_before=1	10
run_before(3)	ZerosLeft=2; run_before=0	1
run_before(2)	ZerosLeft=2; run_before=0	1
run_before(1)	ZerosLeft=2; run_before=1	01
run_before(0)	ZerosLeft=1; run_before=1	No code required; last coefficient.

The transmitted bitstream for this block is 000010001110010111101101 .

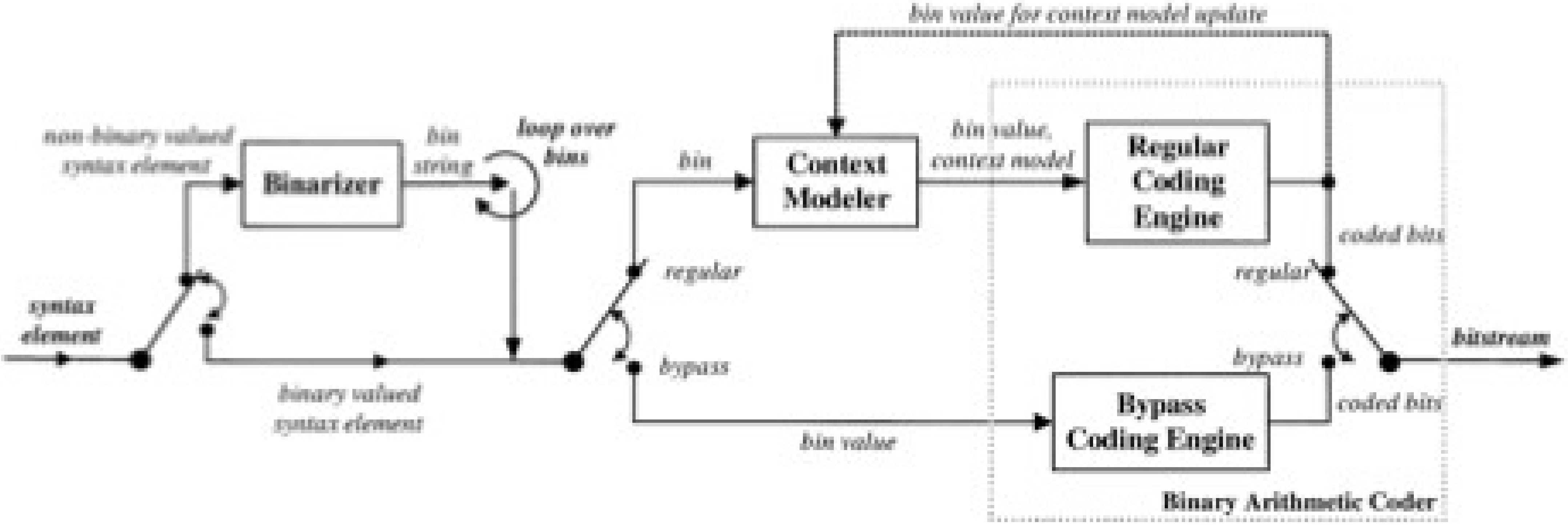
CODEC - Entropy Coding - Variable Length Coding

The output array is "built up" from the decoded values as shown below. Values added to the output array at each stage are underlined.

Code	Element	Value	Output array
0000100	coeff_token	TotalCoeffs=5, T1s=3	Empty
0	T1 sign	+	1
1	T1 sign	-	-1, 1
1	T1 sign	-	-1, -1, 1
1	Level	+1	1, -1, -1, 1
0010	Level	+3	3, 1, -1, -1, 1
111	TotalZeros	3	3, 1, -1, -1, 1
10	run_before	1	3, 1, -1, -1, 0, 1
1	run_before	0	3, 1, -1, -1, 0, 1
1	run_before	0	3, 1, -1, -1, 0, 1
01	run_before	1	3, 0, 1, -1, -1, 0, 1

The decoder has inserted two zeros; however, TotalZeros is equal to 3 and so another 1 zero is inserted before the lowest coefficient, making the final output array:
0, 3, 0, 1, -1, -1, 0, 1

CODEC - Entropy Coding - (CA) Binary Arithmetic Coding

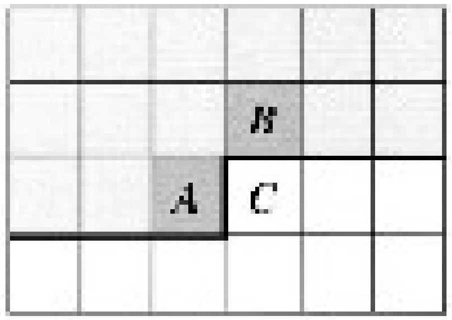


Entropy coding - *(CA) Binary Arithmetic Coding*

Only works on binary values...

1 - Binarization

- Change into binary codes the elements which are not binary values (mvd, dct coeff., ...)
  - Using Exp Golomg (VLC)



2 - Selection of context model

- always based when applicable on the left and top neighbor previous context.

3 - Arithmetic Coding

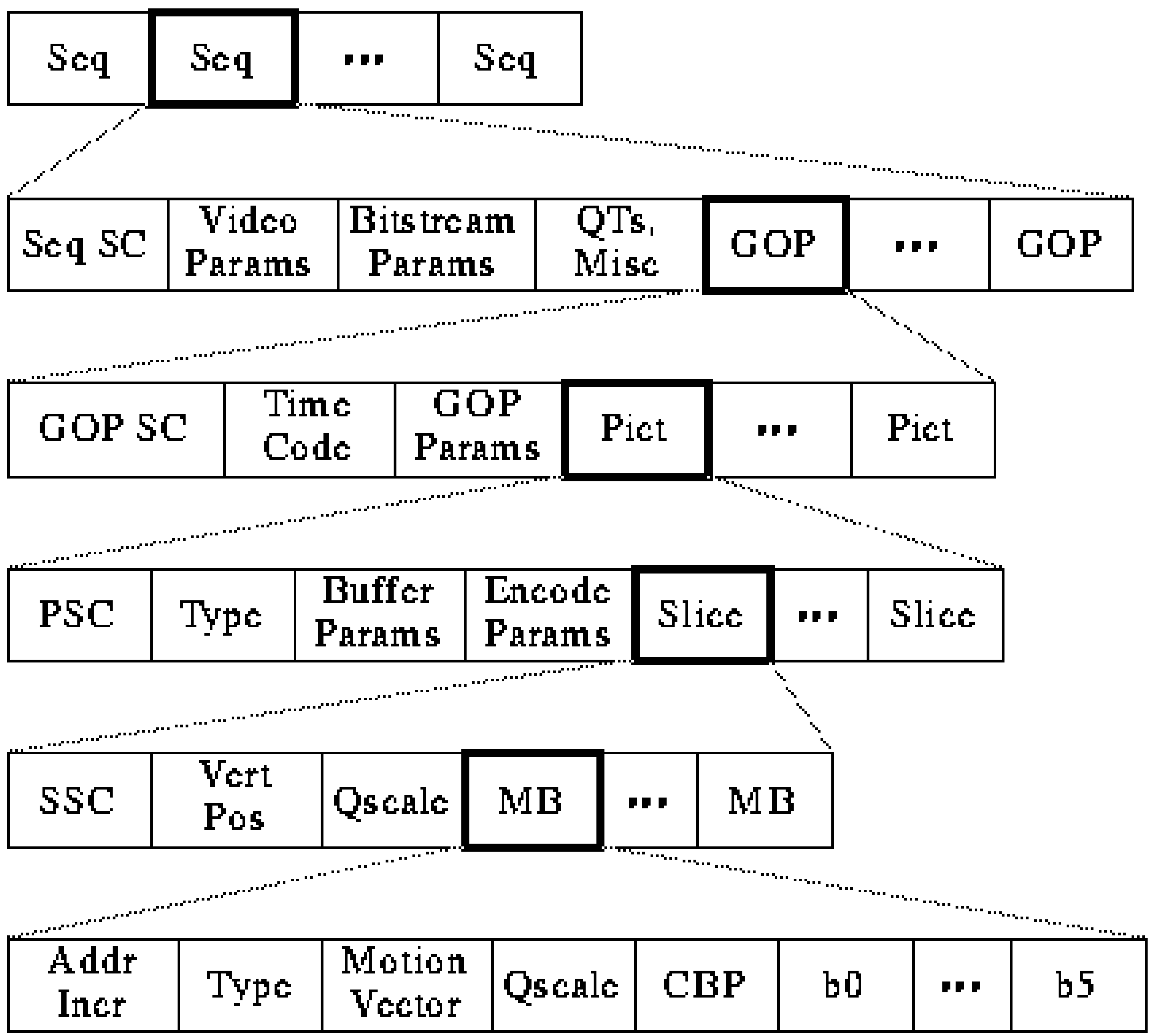
4 - Update selected context model

Provides up to 15% gain on CAVLC.

Not available in H.264/AVC Baseline profile.

- High complexity

MPEG - 2 - Video Data Hierarchy



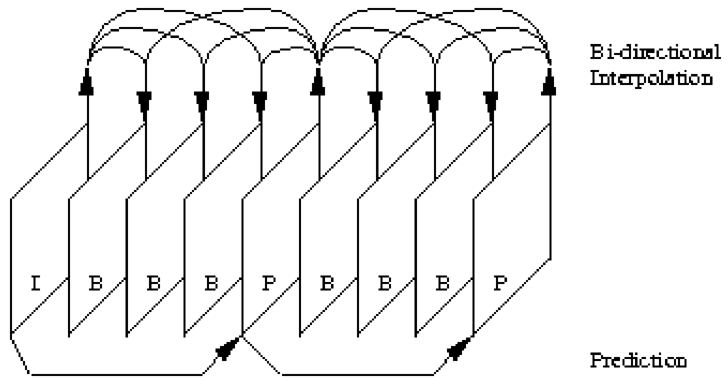
MPEG 2 - Video Data Hierarchy

- Pixel :
- Block :
  - 8x8 pixels block (smallest prediction unit in MPEG-2)
- Macroblock :
  - 16x16 pixels block
- Slice :
  - Set of adjacent macroblocks in a picture (usually a row of macroblocks). used as error resilience mechanism (skip if error).
- Picture :
  - Y, Cb, Cr rectangular matrices
- Group of Picture (GOP) :
  - Group of one or more pictures allowing for random stream access.
- Sequence:
  - Includes one or more groups of pictures.
- Sequence(GOPs(Pictures(slices(16x16_Macroblock(Block(pixel))))))

MPEG-2 - Picture structure

Intra Pictures (I - Pictures)

- coded using only information present in the picture itself,
- provides potential random access points into the compressed videc
- uses only transform coding and provide moderate compression.



Predicted Pictures (P - Pictures)

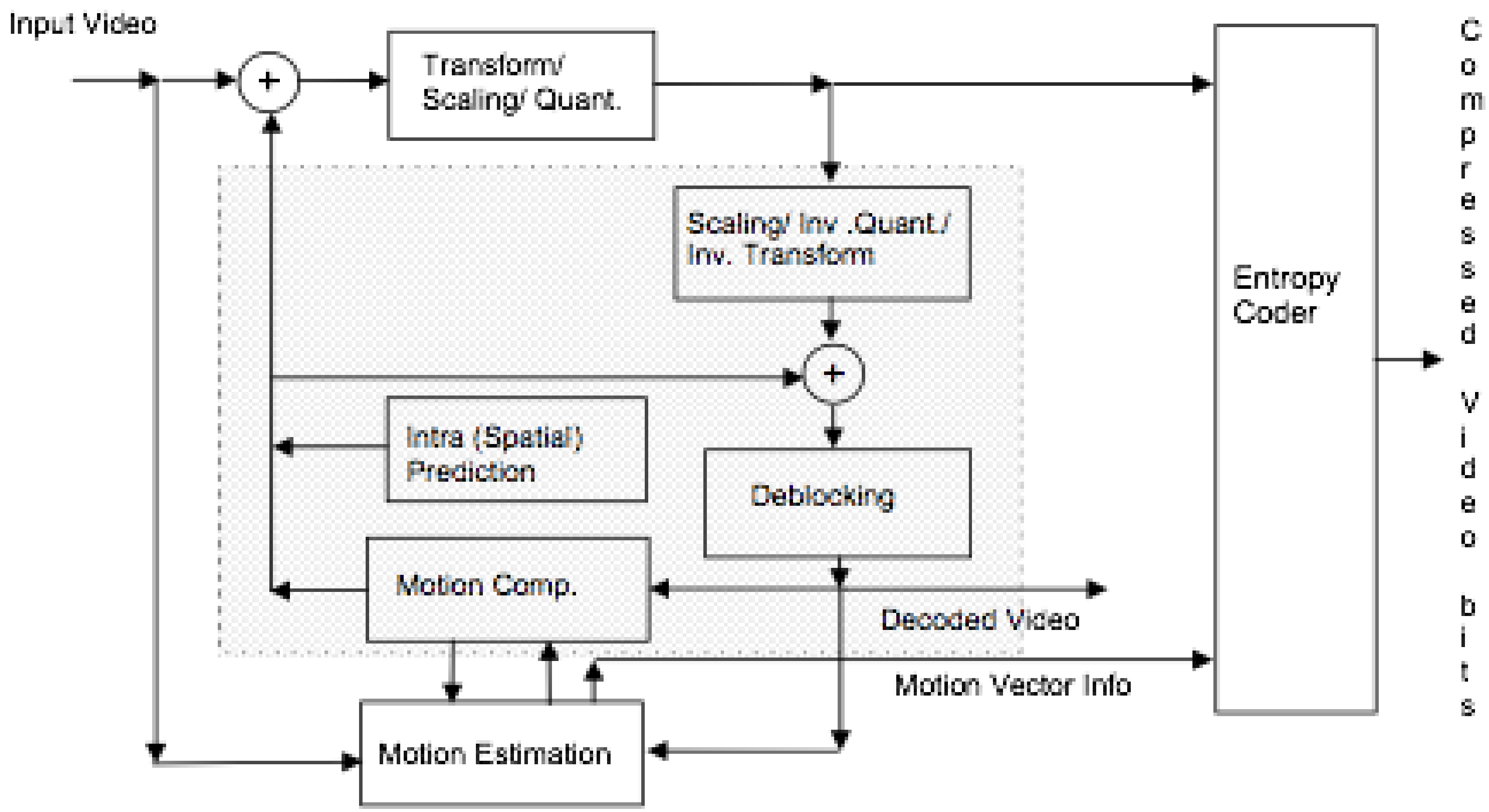
- coded with respect to the nearest previous I- or P-pictures. (forward prediction)
- can serve as a prediction reference for B-pictures and future P-pictures.
- use motion compensation to provide more compression than is possible with I-pictures.

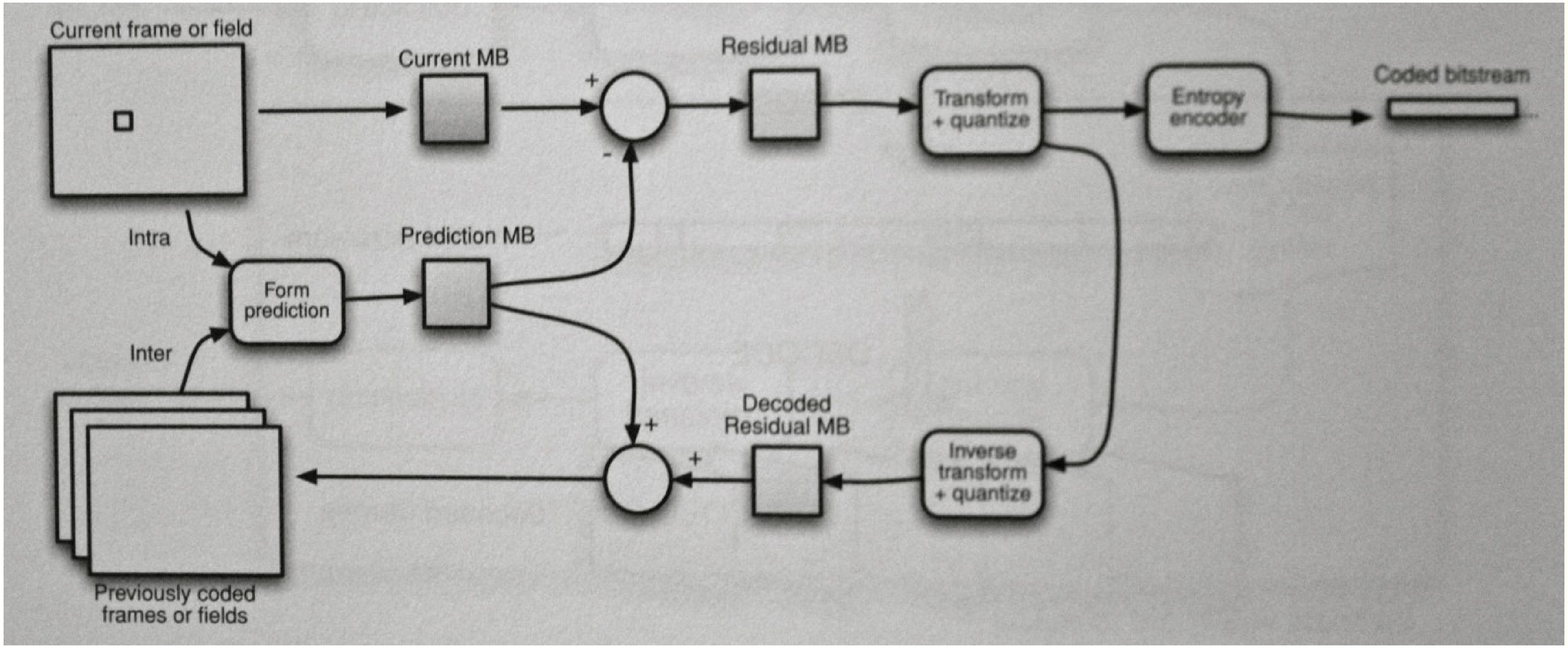
Bi-directional Pictures (B - Pictures)

- use both a past and future picture as a reference. (bidirectional prediction).
- B-pictures provide the most compression since it use the past and future picture as a regerence, however, the computation time is the largest.
- Can only be predicted from not more than 2 frames I or P.
- Can never be referenced.

H.264/AVC

H.264/AVC - Block Diagram (1/2)




Current frame or field
Current MB
Residual MB
Transform + quantize
Entropy encoder
Coded bitstream
+
-
Intra
Prediction MB
Form prediction
Inter
Decoded Residual MB
+
+
Inverse transform + quantize
Previously coded frames or fields

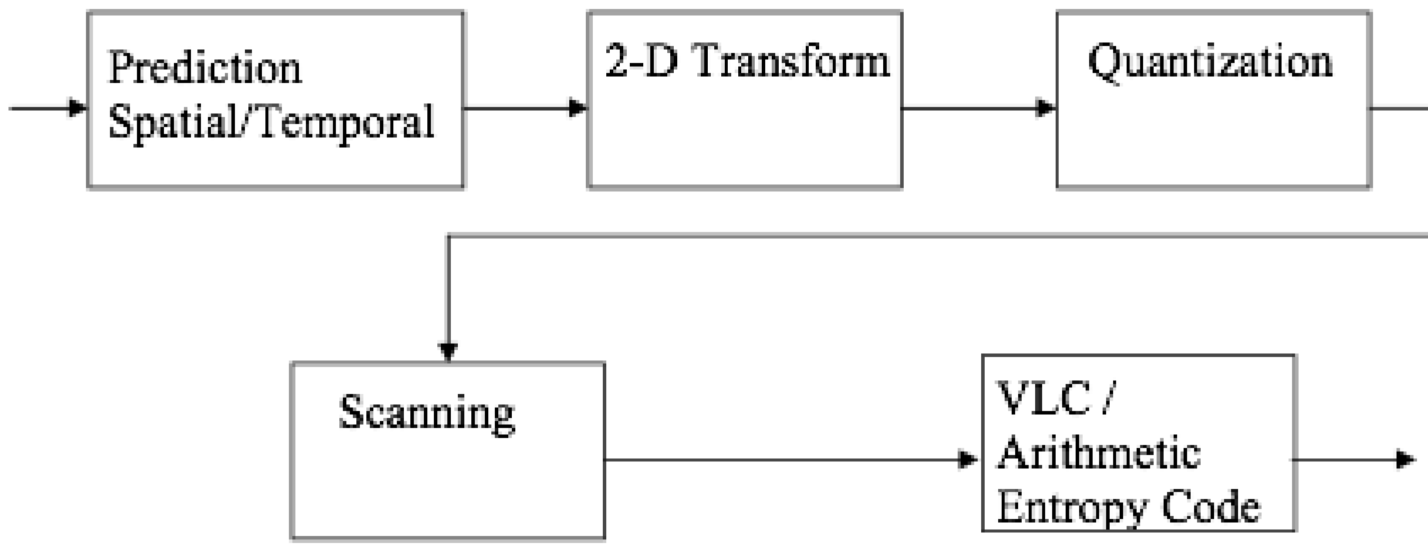
H.264/AVC - Block Diagram (2/2)

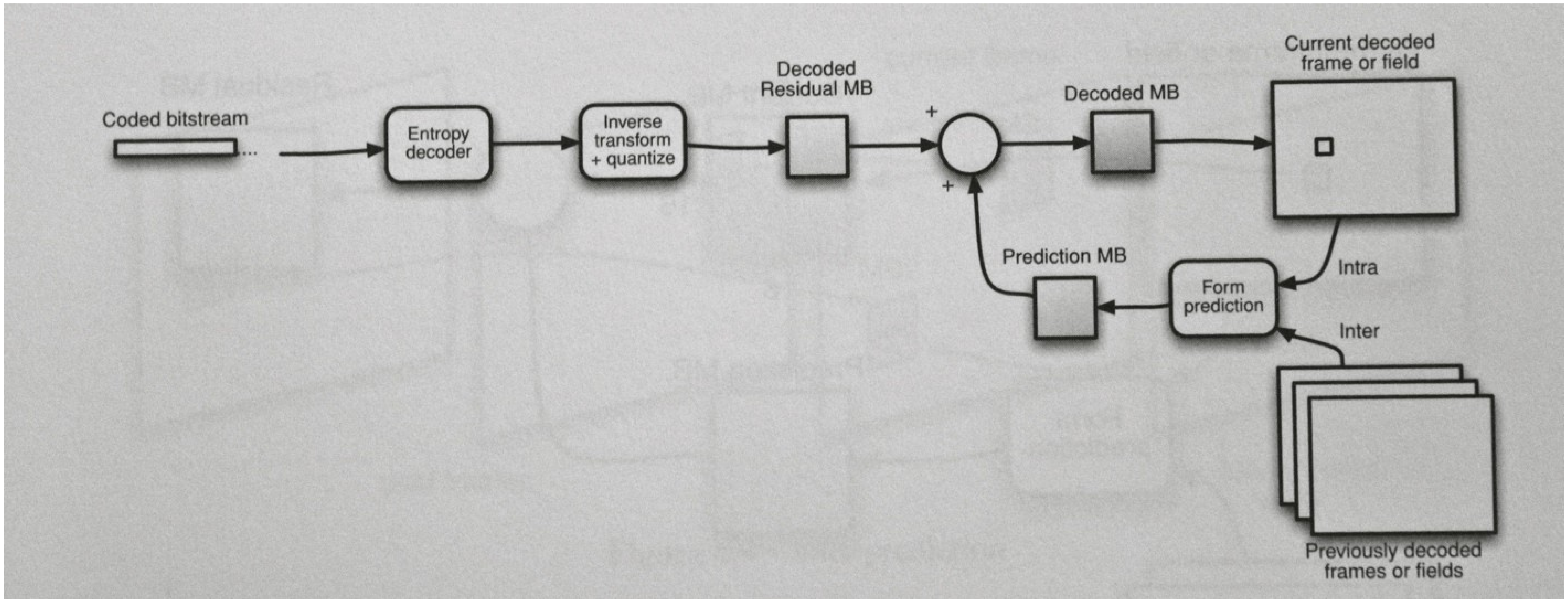
1 - Context based spatial prediction of 4x4 block or 16x16 (if intra)

2- 2-D tranform of residual difference between Input block and prediction

3 - Adaptive quantization of coefficients

4 - Scanning of coefficient according to format





Coded bitstream
Entropy decoder
Inverse transform + quantize
Decoded Residual MB
+
+
Decoded MB
Current decoded frame or field
Prediction MB
Form prediction
Intra
Inter
Previously decoded frames or fields

H.264/AVC - Tools - Coarse tools (1/2)

Adaptive In-loop /deblocking Filters

- Adapts to boundary strength and gradient of image

New video data structures

- Slices concept instead of Pictures and GOP

Hierarchical B frames

- Allows for Multi-frame reference in different order

Entropy coding

- Context adaptive
  - Variable length coding (CAVLC)
  - Arithmetic Coding (CABAC)

H.264/AVC - Additionnal Tools (2/2)

"Intra" spatial (block based) prediction

- Full-macroblock luma or chroma prediction – 4 modes (directions) for prediction
- 8x8 (FRExt-only) or 4x4 luma prediction – 9 modes (directions) for prediction "Inter" temporal prediction – block based motion estimation and compensation

Multiple reference pictures

- Reference B pictures
- Arbitrary referencing order
- Variable block sizes for motion compensation

Seven block sizes: 16x16, 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4

- 1/4-sample luma interpolation (1/4 or 1/8th-sample chroma interpolation)

Weighted prediction

Frame or Field based motion estimation for interlaced scanned video

Interlaced coding features

- Frame-field adaptation
- Picture Adaptive Frame Field (PicAFF)
- MacroBlock Adaptive Frame Field (MBAFF)
- Field scan

Lossless representation capability

- Intra PCM raw sample-value macroblocks
- Entropy-coded transform-bypass lossless macroblocks (FRExt-only)

8x8 (FRExt-only) or 4x4 integer inverse transform (conceptually similar to the well-known DCT)

Residual color transform for efficient RGB coding without conversion loss or bit expansion (FRExt-only)

Scalar quantization

Encoder-specified perceptually weighted quantization scaling matrices (FRExt-only)

Logarithmic control of quantization step size as a function of quantization control parameter

Deblocking filter (within the motion compensation loop)

Coefficient scanning

Zig-Zag (Frame)

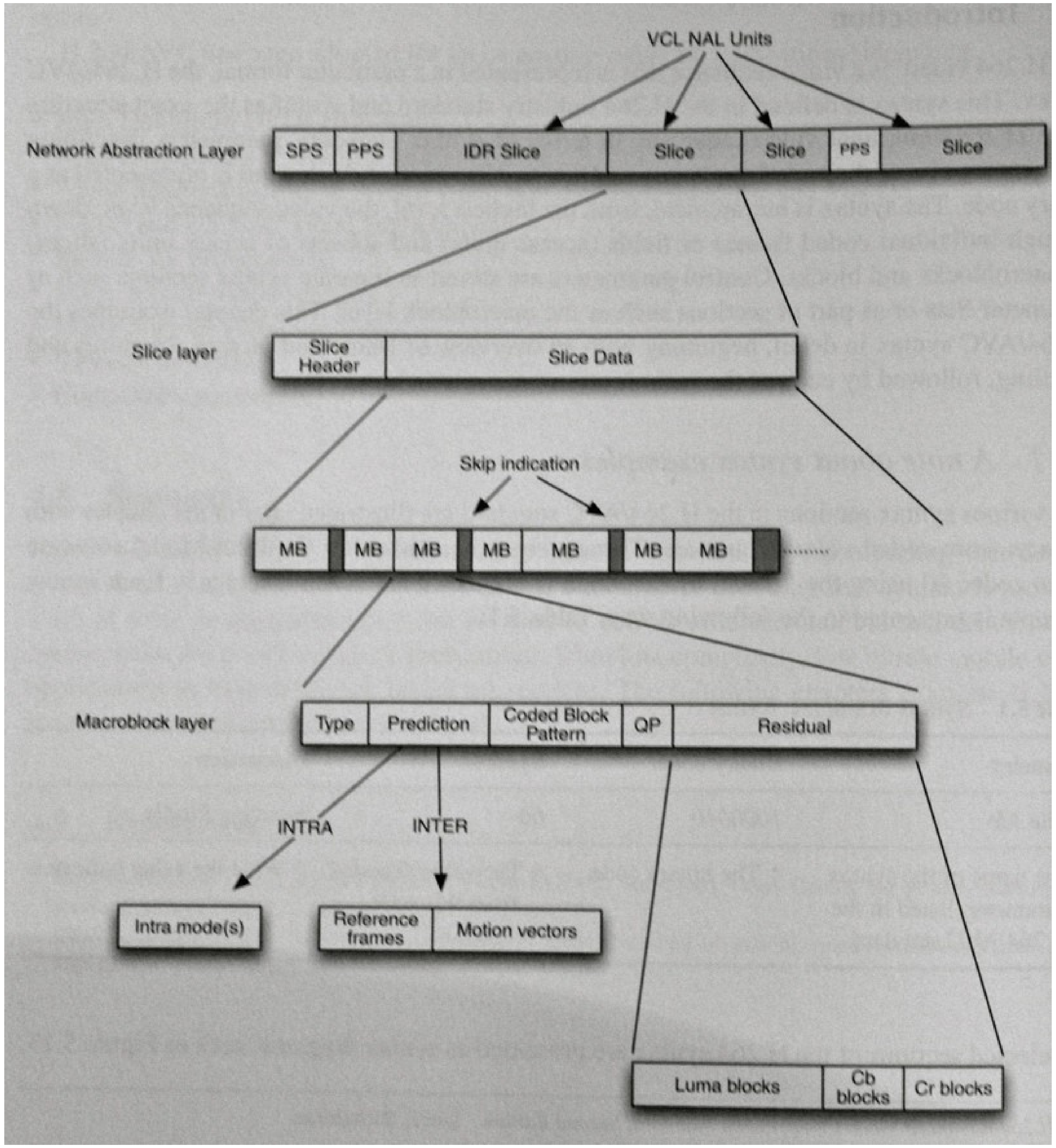
- Field
- Lossless Entropy coding
- oUniversal Variable Length Coding (UVLC) using Exp-Golomb codes o Context Adaptive VLC (CAVLC) o Context-based Adaptive Binary Arithmetic Coding (CABAC)

Error Resilience Tools o Flexible Macroblock Ordering (FMO) o Arbitrary Slice Order (ASO) o Redundant Slices

SP and SI synchronization pictures for streaming and other uses Various color spaces supported (YCbCr of various types, YCgCo, RGB, etc. – especially in FRExt) 4:2:0, 4:2:2 (FRExt-only), and 4:4:4 (FRExt-only) color formats Auxiliary pictures for alpha blending (FRExt-only)

H.264/AVC - Video data hierarchy

- Pixel
- Block :
  - 4x4 pixels block
- Submacroblock partitions (6)
  - depends on
- Macroblock :
  - 16x16 pixels block
- Slices (5 new types)
  - I, P, B, SP and SI slices
  - Set of adjacent macroblocks in a picture. used as error resilience mechanism (skip if error).
- Picture :
  - Y, Cb, Cr rectangular matrices
- Sequence:
  - Includes one or more groups of pictures.
- Sequence(Pictures(slices(16x16_Macroblock(SubMacroblock(Block(pixel))))))


VCL NAL Units
Network Abstraction Layer
SPS
PPS
IDR Slice
Slice
Slice
PPS
Slice
Slice layer
Slice Header
Slice Data
Skip indication
MB
MB
MB
MB
MB
MB
MB
Macroblock layer
Type
Prediction
Coded Block Pattern
QP
Residual
INTRA
INTER
Intra mode(s)
Reference frames
Motion vectors
Luma blocks
Cb blocks
Cr blocks

H.264/AVC - Slice Types

Slice

- Region of a picture that can be decoded independently

I - Slice

- All macroblocks of the slice are coded using Intra prediction.

P - Slice (predictive)

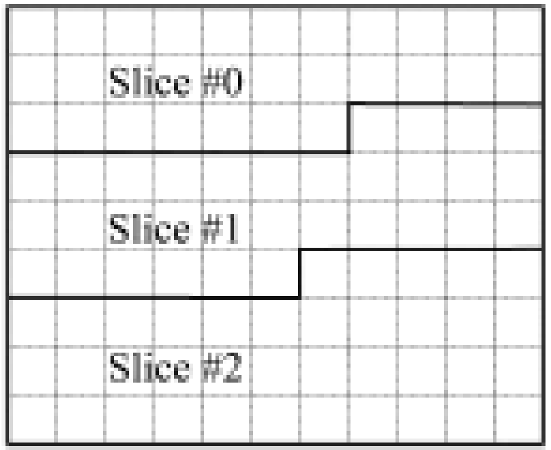
- Contains macroblock predicted by at most one motion compensation signal in addition to Intra-predicted macroblock.

B - Slice (Bi-predictive)

- can include coding types of P slices and mocroblock predicted from

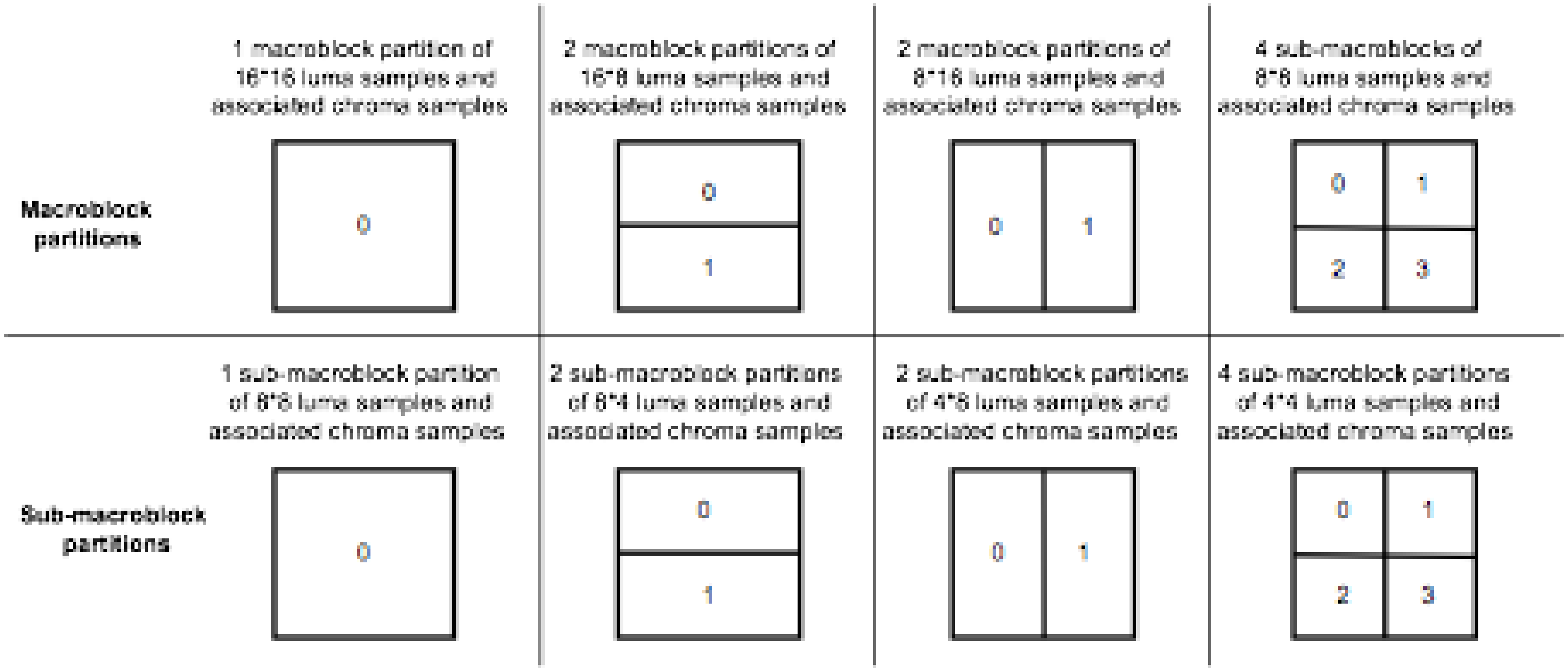
SP - Slice (Switching P-slice)

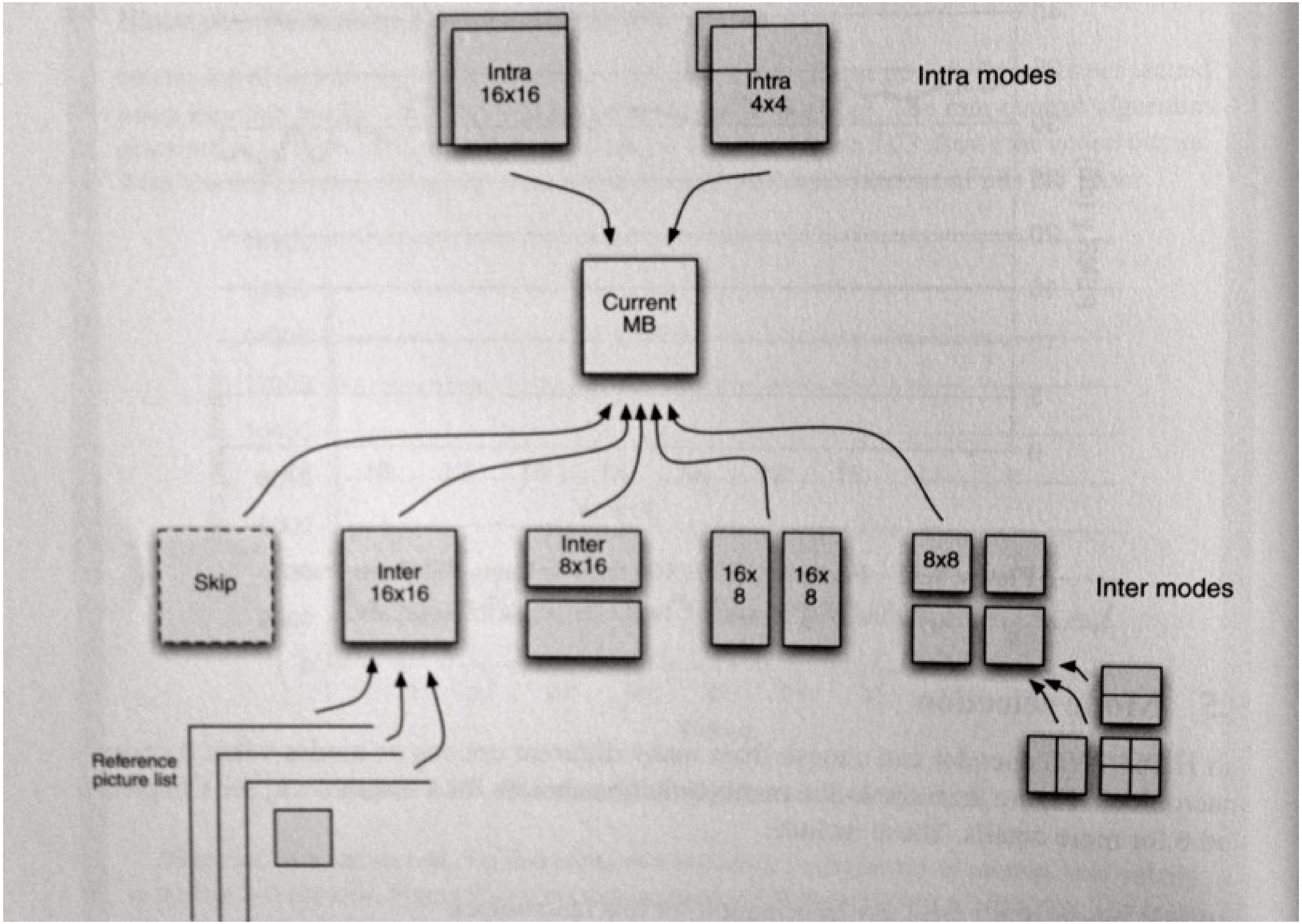
SI - Slice (Switching I-slice)



H.264/AVC - New (7) Macroblock partitions for predictions

- 16x16 Macroblock Partition
- 8x8 Sub macroblock Partition




Intra 16x16
Intra 4x4
Intra modes
Current MB
Skip
Inter 16x16
Inter 8x16
16x 8
16x 8
8x8
Inter modes
Reference picture list

H.264/AVC - Intra (spatial) Prediction

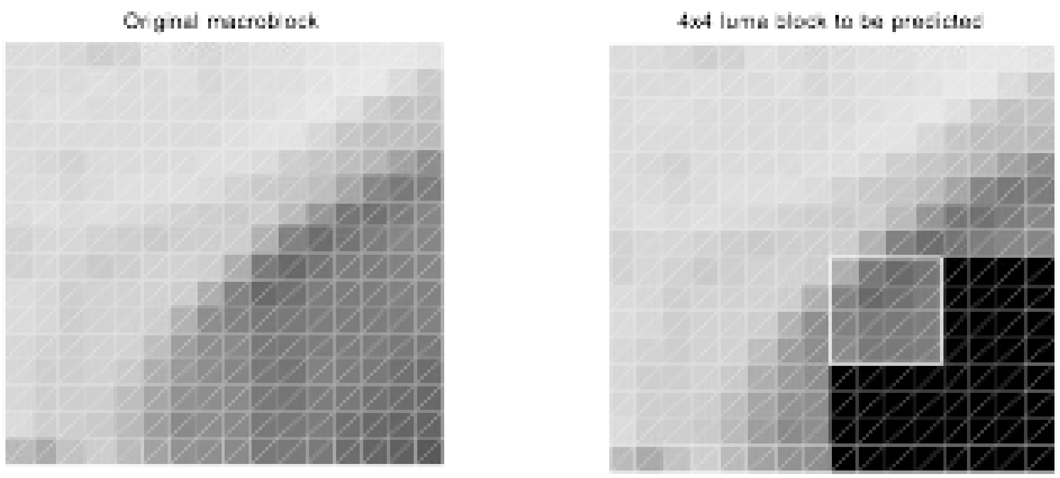
3 Modes:

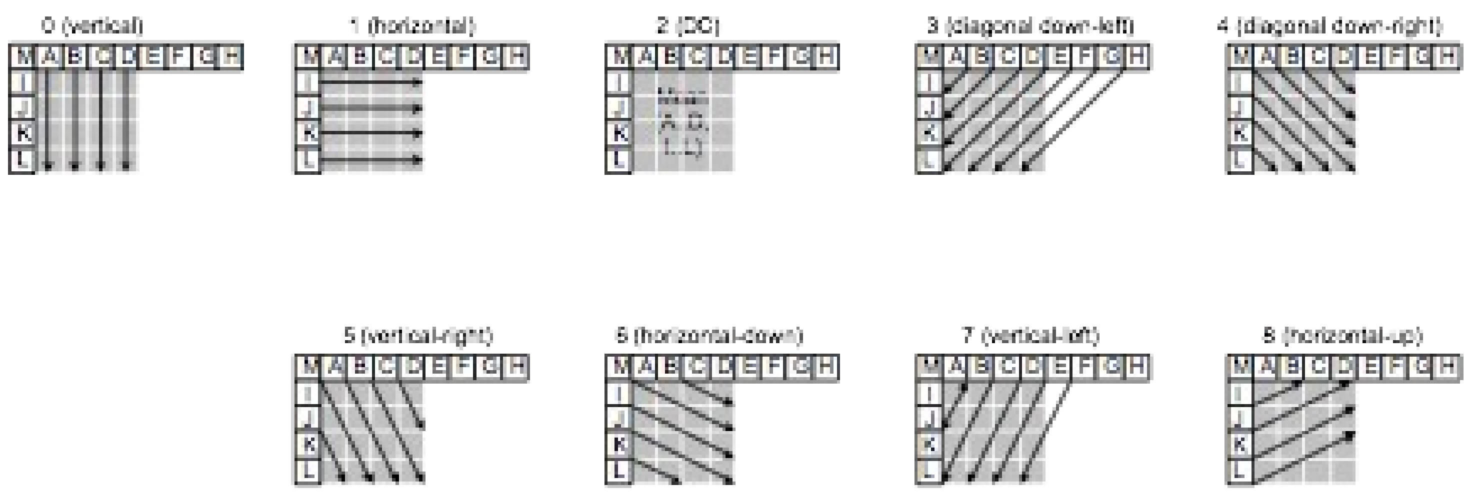
- Intra_4x4
  - The 4x4 block is predicted in 8 directions from 4x4 intra blocks.
  - weighted average value of surrounding pixel from previously predicted blocks.
  - used for coding high detail area.
- Intra_16x16
  - predictions only in 4 directions (4 first in 4x4 mode)
  - Best for smooth area coding
- Intra_PCM
  - Macroblock is not encoded but transmitted as is.
  - No residual signal sent.

Residual signal = difference between original and prediction + prediction_mode.

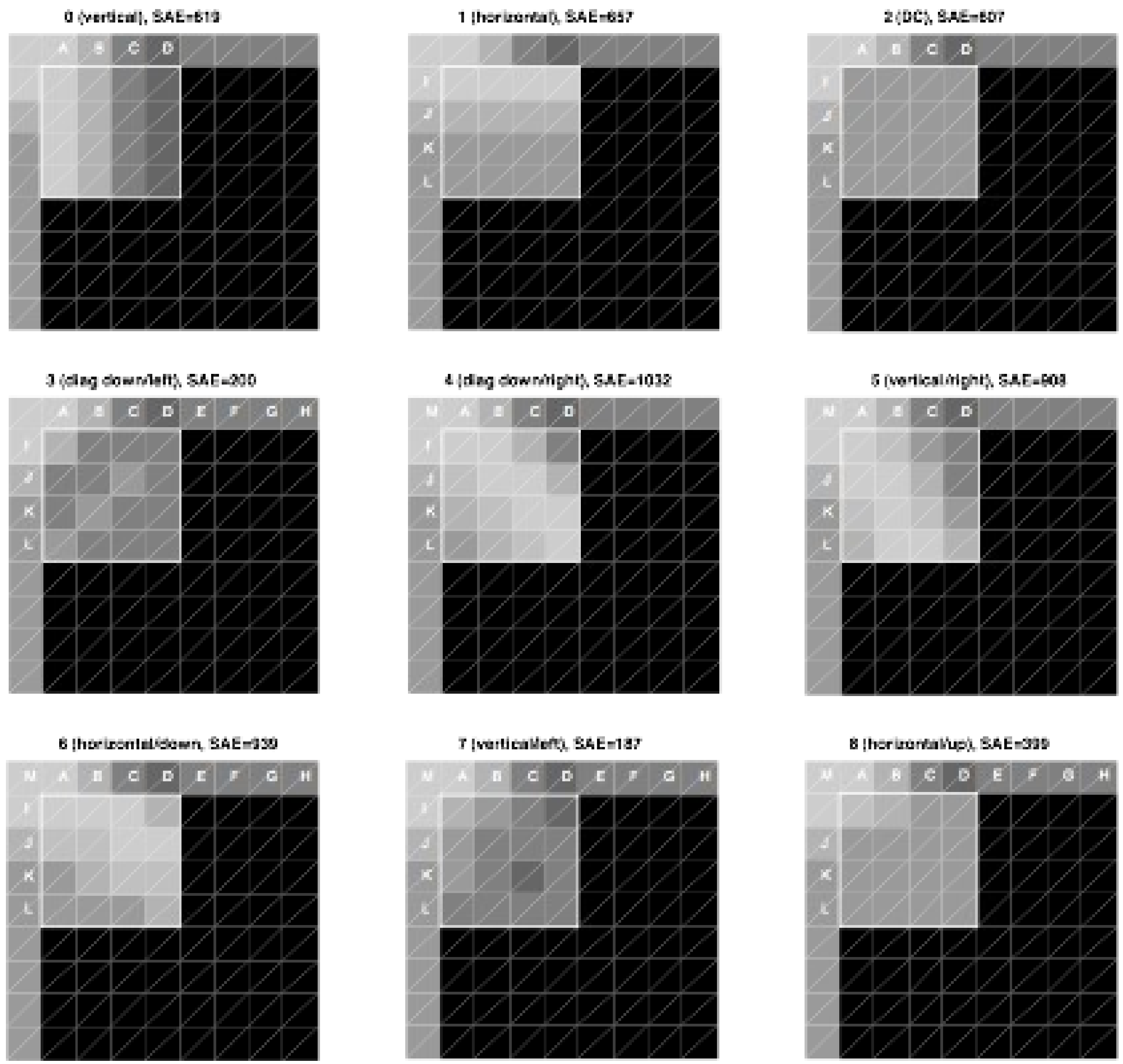
- sent to the transform module.

H.264/AVC - Intra Prediction 4x4 (1/2)





H.264/AVC - Intra Prediction 4x4 (2/2)



H.264/AVC - Inter Prediction

Multiple prediction references possible

- including B-Macroblocks previously predicted
- Larger frame store needed (Decoded Picture Buffer - DPB).
- Number of reference frames determined by the Levels
- Reference index is coded with the residual.

P-Predictive

- Results of at most one Motion compensation signal (predicted Mb + MV)

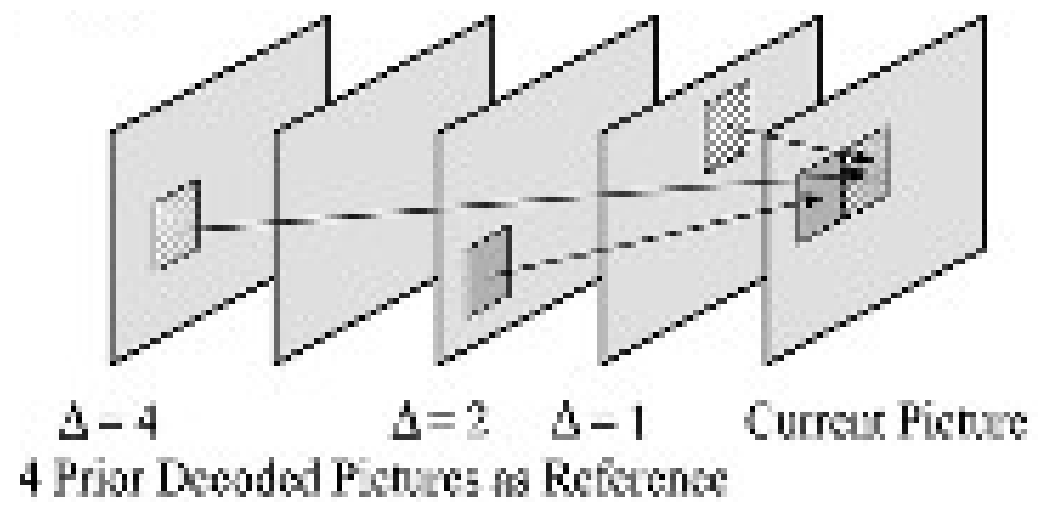
B-Predictive

- Result of weighted average of 2 MC signals (Mb and MVs)

Direct Mode

Skip Mode

- P Macroblock remaining same



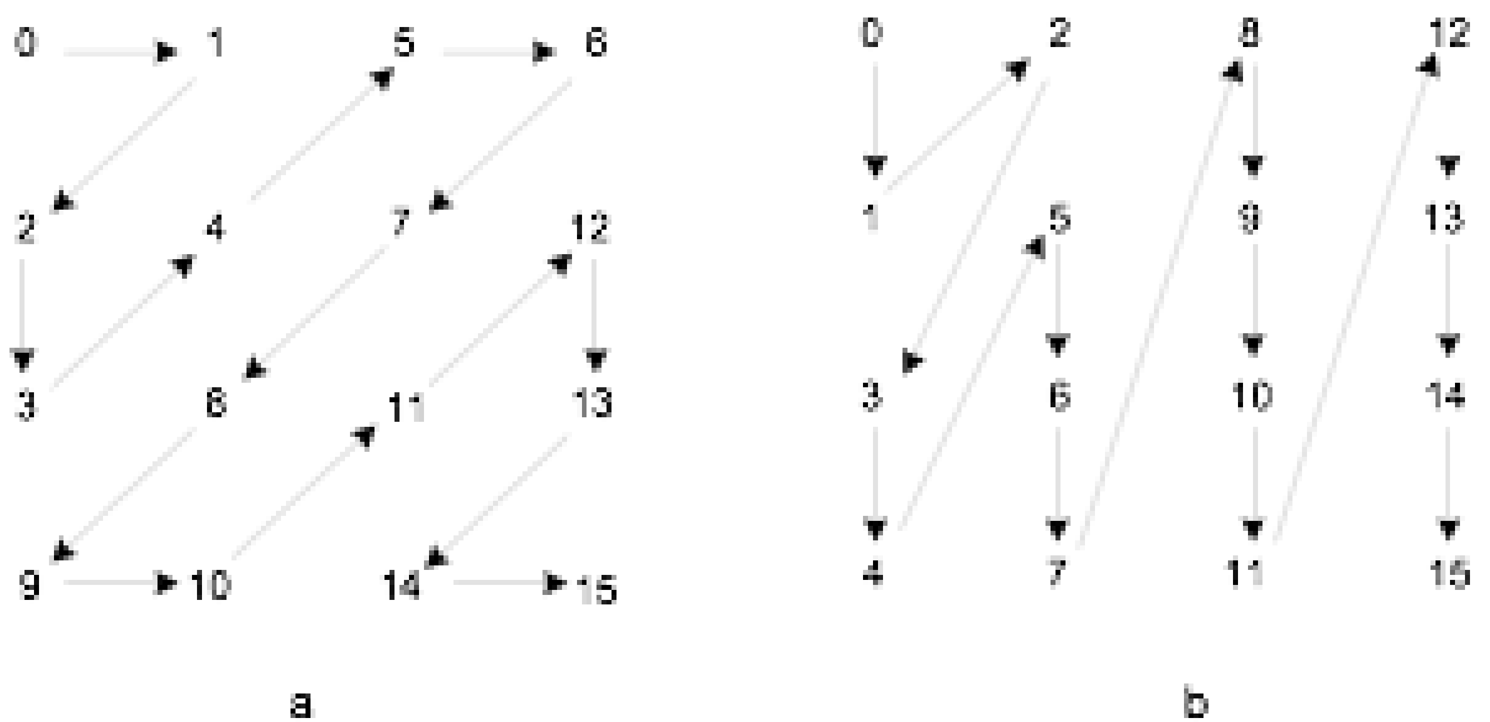
H.264/AVC - Syntax Elements

Syntax element	Slice type		
	SI/I	P/SP	B
mb_type	0/3-10	14-20	27-35
mb_skip_flag		11-13	24-26
sub_mb_type		21-23	36-39
mvd (horizontal)		40-46	40-46
mvd (vertical)		47-53	47-53
ref_idx		54-59	54-59
mb_qp_delta	60-63	60-63	60-63
intra_chroma_pred_mode	64-67	64-67	64-67
prev_intra4x4_pred_mode_flag	68	68	68
rem_intra4x4_pred_mode	69	69	69
mb_field_decoding_flag	70-72	70-72	70-72
coded_block_pattern	73-84	73-84	73-84
coded_block_flag	85-104	85-104	85-104
significant_coeff_flag	105-165, 277-337	105-165, 277-337	105-165, 277-337
last_significant_coeff_flag	166-226, 338-398	166-226, 338-398	166-226, 338-398
coeff_abs_level_minus1	227-275	227-275	227-275
end_of_slice_flag	276	276	276

H.264/AVC - Scanning

Scanning procedure for 4x4 macroblock sample after the Pseudo DCT transform

- a) for Progressive
- b) for interlaced content (due to reduced correlation between lines)



H.264/AVC - Interlaced tools PAFF vs MbAFF (1/2)

PAFF

- Picture Adaptive Field Frame coding
- Frame Mode
  - Combine the two fields together as a single frame and encode as a single picture.
- Field Mode
  - Encode each field separately as a separate picture

MbAFF

- Macroblock adaptive Field Frame Coding
- The entire frame is considered to be one single picture.
  - Frame mode
    - Macroblocks of the two fields can be encoded together as one single macroblock
    - both macroblocks are kept in pairs in the slice/frame.
  - Field mode
    - Macroblocks are encoded separately. each containing own field values.

MbAFF Provides 15% gain over PAFF.

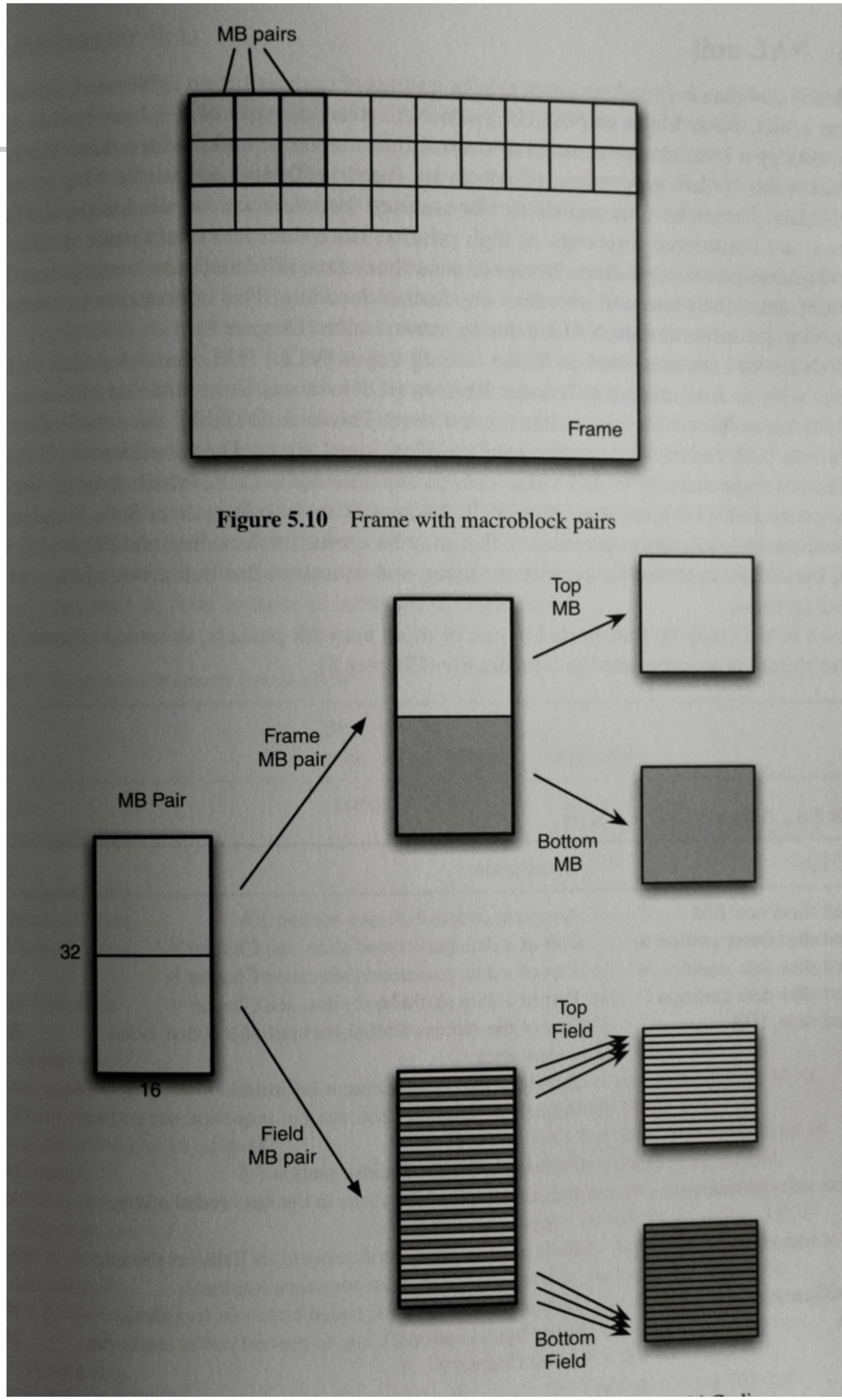
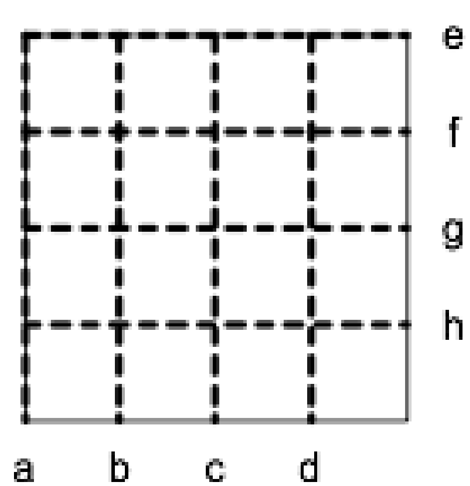


Figure 5.10 Frame with macroblock pairs

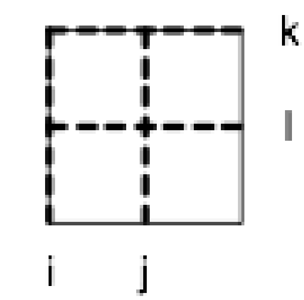
H.264/AVC - Adaptive In-loop filters / deblocking filters

Applied on Macroblock 16x16 edges

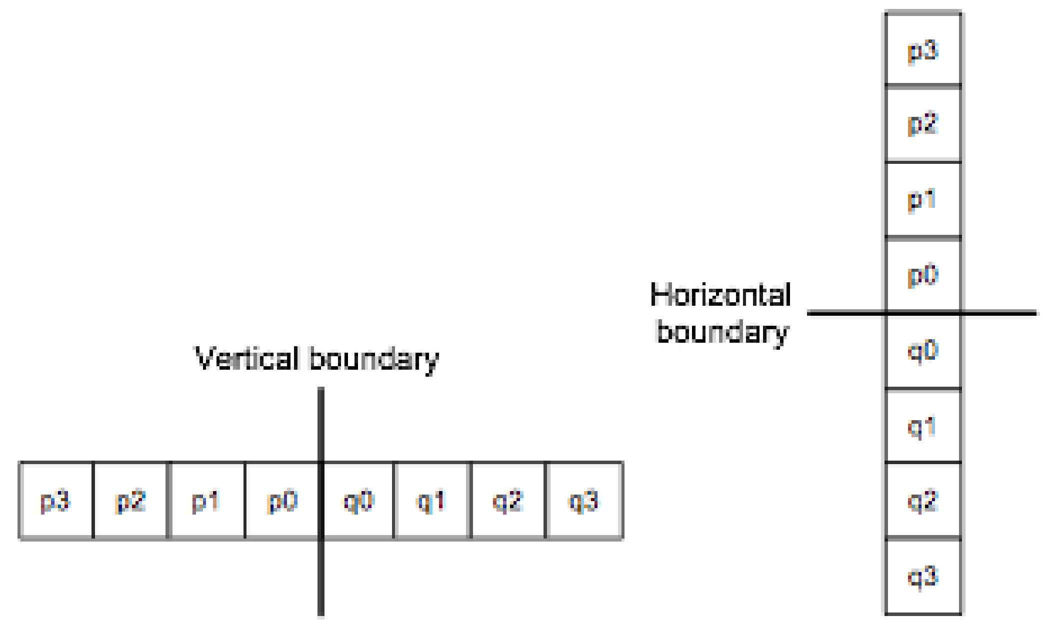
- Designed to remove blocky artefacts
- Applied on macroblock borders according to context
- Filter strength adapted to content
- Strong tool against mosquito noise



Boundary filtering: 16x16 luma



8x8 chroma



H.264/AVC - Adaptive In-loop filters / deblocking filters

In loop filtering depends on

- the boundary strength between macroblocks
- the gradient between adjacent pixels accross the boundary
  - magnitude of pixel difference is the indicator e.g. |p0 -q0| .

The filter gets longer th stronger and the larger the gradient

- from 3-tap to a 5 tap filter.

Provides 5-10% bit-rate saving on non filtered video.

p or q is intra coded **and** boundary is a macroblock boundary	Bs=4 (strongest filtering)
p or q is intra coded and boundary is **not** a macroblock boundary	Bs=3
neither p or q is intra coded; p or q contain coded coefficients	Bs=2
neither p or q is intra coded; neither p or q contain coded coefficients; p and q have different reference frames **or** a different number of reference frames **or** different motion vector values	Bs=1
neither p or q is intra coded; neither p or q contain coded coefficients; p and q have same reference frame and identical motion vectors	Bs=0 (no filtering)

H.264/AVC - The GOP and IDR question...

An IDR picture is a

- Instantaneously Decoded Refresh frame
- Usually an I-frame
- Forbids that any frame after it can use any frame before it as a prediction reference.

Closed GOP

- Self contained GOP
  - Frames / Macroblock inside the GOP do not have references outside the GOP.
  - Useful for fast browsing of the stream (DVD, Blu Ray etc)
- GOP start I-frames are IDR frames

Open GOP

- references outdide the GOP are permitted.

In H.264

- GOP structure not anymore relevant (Macroblock types prevail) but still considered for rate control.

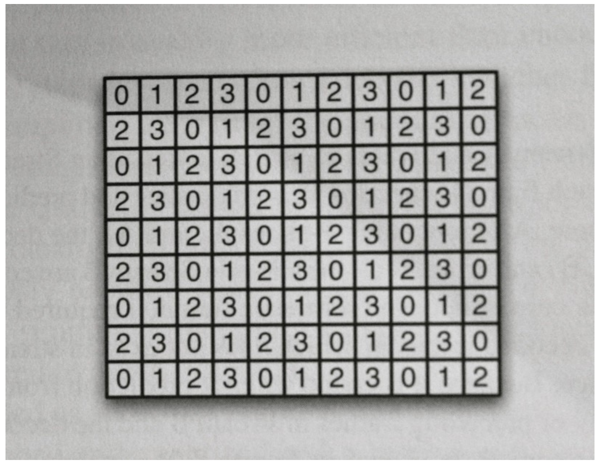
H.264/AVC - Error Resilience Tools

FMO - Flexible Macroblock Ordering

- Reorders the Macroblocks according to predefine patterns to increase robustness to transmission losses.
- Only available in the constrained profile.

ASO - Arbitrary Slice Order

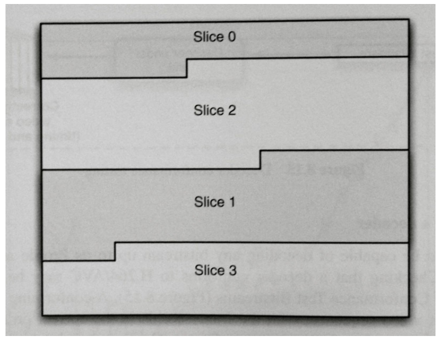
- Same as FMO at slice level



Data Partitioning

Redundant slice

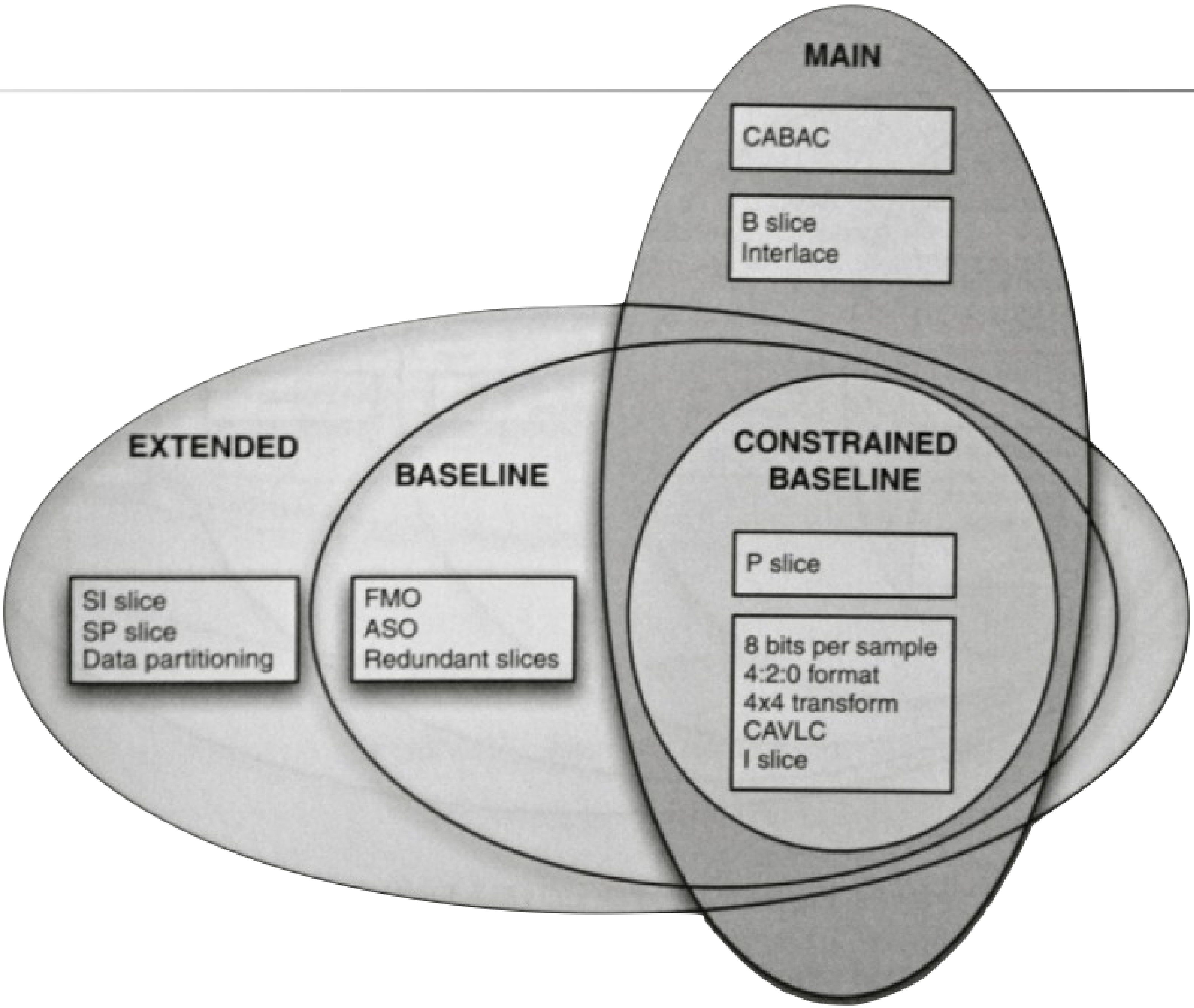
- Duplicate slices within the stream.
  - skipped by the decoder when received
  - Increases the stream error resilience but reduces available bandwidth.

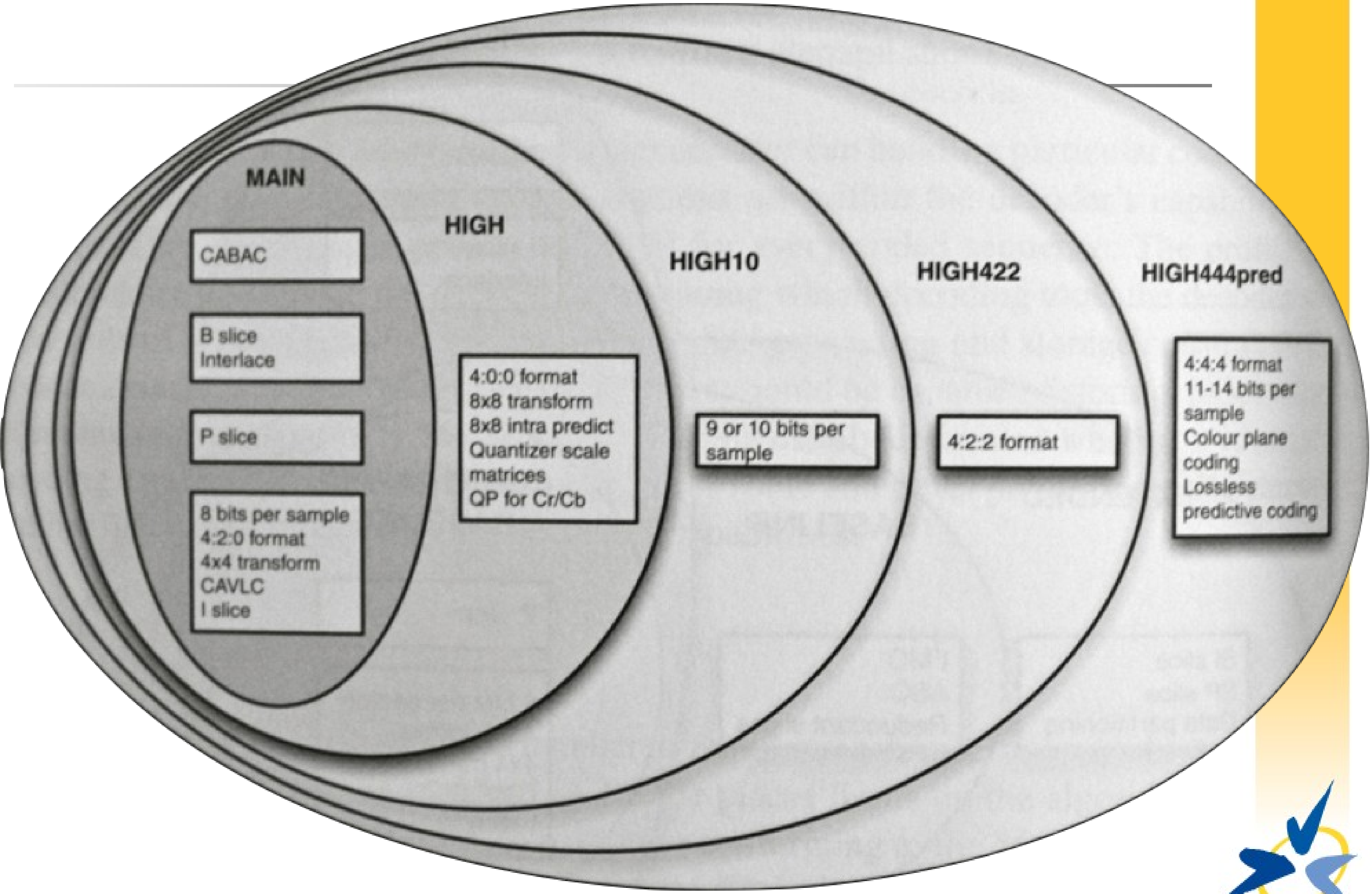


H.264/AVC - Profiles

Coding Tools	Baseline	Main
I and P Slices	X	X
CAVLC	X	X
CABAC		X
B Slices		X
Interlaced Coding (PicAFF, MBAFF)		X
Enh. Error Resil. (FMO, ASO, RS)	X	
Further Enh. Error Resil (DP)		
SP and SI Slices		

Coding Tools	High	High 10	High 4:2:2	High 4:4:4
Main Profile Tools	X	X	X	X
4:2:0 Chroma Format	X	X	X	X
8 Bit Sample Bit Depth	X	X	X	X
8x8 vs. 4x4 Transform Adaptivity	X	X	X	X
Quantization Scaling Matrices	X	X	X	X
Separate Cb and Cr QP control	X	X	X	X
Monochrome video format	X	X	X	X
9 and 10 Bit Sample Bit Depth		X	X	X
4:2:2 Chroma Format			X	X
11 and 12 Bit Sample Bit Depth				X
4:4:4 Chroma Format				X
Residual Color Transform				X
Predictive Lossless Coding				X


MAIN
CABAC
B slice
Interlace
EXTENDED
BASELINE
CONSTRAINED BASELINE
SI slice
SP slice
Data partitioning
FMO
ASO
Redundant slices
P slice
8 bits per sample
4:2:0 format
4x4 transform
CAVLC
I slice


MAIN
CABAC
B slice
Interlace
P slice
8 bits per sample
4:2:0 format
4x4 transform
CAVLC
I slice
HIGH
4:0:0 format
8x8 transform
8x8 intra predict
Quantizer scale matrices
QP for Cr/Cb
HIGH10
9 or 10 bits per sample
HIGH422
4:2:2 format
HIGH444pred
4:4:4 format
11-14 bits per sample
Colour plane coding
Lossless predictive coding

H.264/AVC - Levels

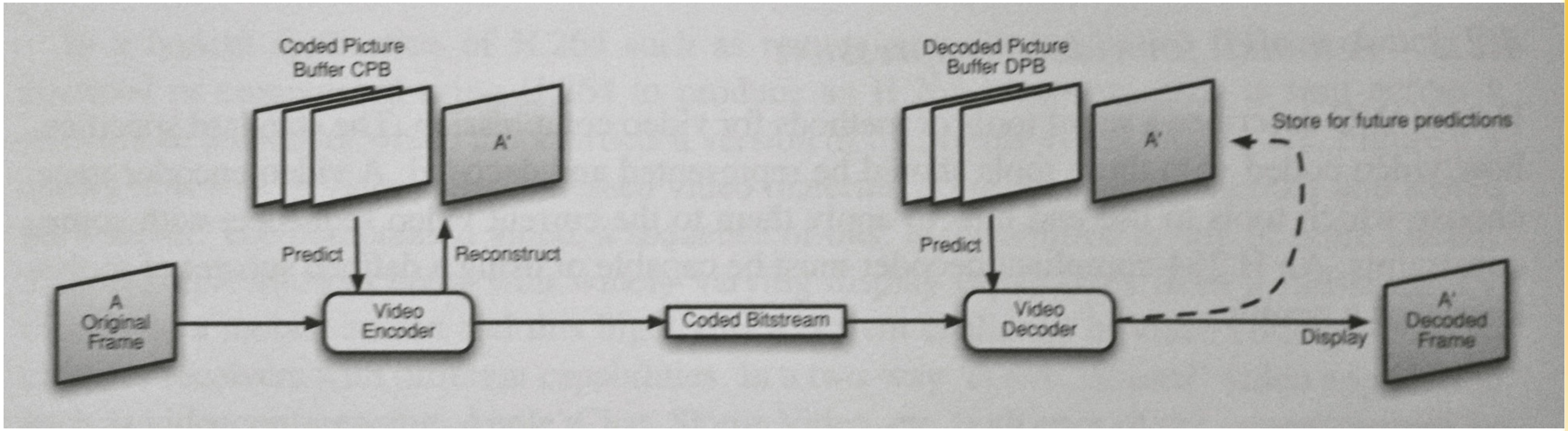
Level Number	Typical Picture Size	Typical frame rate	Maximum compressed bit rate (for VCL) in Non-FRExt profiles	Maximum number of reference frames for typical picture size
1	QCIF	15	64 kbps	4
1b	QCIF	15	128 kbps	4
1.1	CIF or QCIF	7.5 (CIF) / 30 (QCIF)	192 kbps	2 (CIF) / 9 (QCIF)
1.2	CIF	15	384 kbps	6
1.3	CIF	30	768 kbps	6
2	CIF	30	2 Mbps	6
2.1	HHR (480i or 576i)	30 / 25	4 Mbps	6
2.2	SD	15	4 Mbps	5
3	SD	30 / 25	10 Mbps	5
3.1	1280x720p	30	14 Mbps	5
3.2	1280x720p	60	20 Mbps	4
4	HD Formats (720p or 1080i)	60p / 30i	20 Mbps	4
4.1	HD Formats (720p or 1080i)	60p / 30i	50 Mbps	4
4.2	1920x1080p	60p	50 Mbps	4
5	2kx1k	72	135 Mbps	5
5.1	2kx1k or 4kx2k	120 / 30	240 Mbps	5

H.264/AVC - Conformance

HRD - Hypothetical Reference Decoder

- Insures that any decoder supporting the appropriate level and profile will be able to decode the encoded stream.
- Constraints on the input and output buffer models ensures implementation agnostic systems.
- Specific constraint on the DPB an CPB buffers:
  - DPB Decoded Picture Buffer
    - Stores the decoded pictures to be displayed and used as reference for coded pictures.
  - CPB Coded Picture Buffer
    - stores the coded pictures in the decoder received through the transmission channel.

The coding delay might suffer a little.


Coded Picture
Buffer CPB
A'
Predict
Reconstruct
A
Original
Frame
Video
Encoder
Coded Bitstream
Decoded Picture
Buffer DPB
A'
Store for future predictions
Predict
Video
Decoder
Display
A'
Decoded
Frame

H.264/AVC - Transport

Raw Bitstream

NAL units (Network abstraction layer) = Header + VCL payload.

- separated the coded video information and the information needed for proper video decoding
- 2 Types of NAL
    - VCL (Video Coding Layer)
        - encoded video content (syntax elements): motion vectors, coefficients etc...
    - Non VCL
        - Set of NAL units without video coded information
            - contains data necessary for proper/enhanced decoding of a picture or sequence of pictures.
            - contains SEI messages etc.

Access Unit (AU)

- Set of non-VCL and VCL NAL units associated to a fully decodable picture
- Smallest unit in a PES packet

Patents…

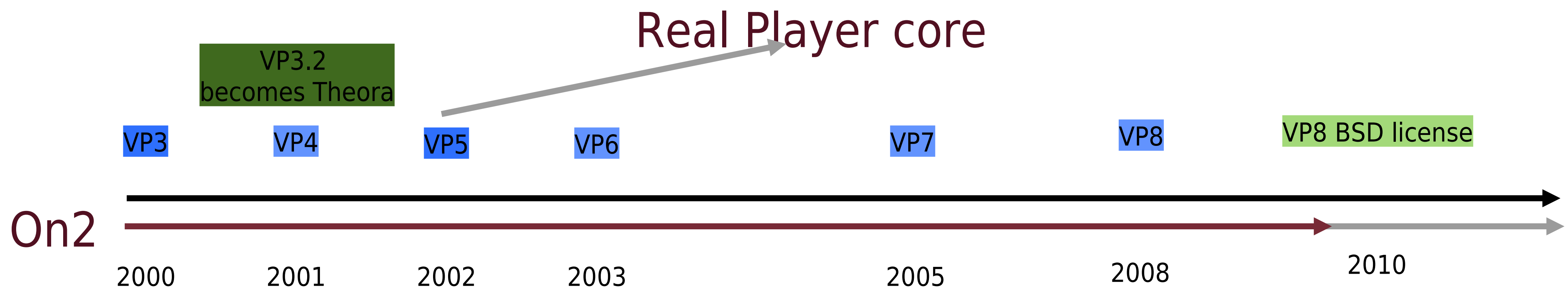
MPEG-LA

- MPEG-LA manages the IP portfolio for H.264/AVC and defines the licensing terms. New terms are valid 2011-2015.
- For companies that manufacture and sell encoders and decoders:
  - No royalty for the first 100,000 units of a licensed product;
  - 20 cents per unit up to 5 million and 10 cents per unit above 5 million.
- For video content or service providers:
  - For individual videos, the royalty is the lower of 2% of the price paid to the licensee or $0.02 per title.
  - For subscription video services, the royalty is an annual fee based on the number of subscribers. A sliding scale goes from 0 (for up to 250,000 subscribers) to $100,000 (for more than 1 million subscribers) per year.

No royalties for ad-supported videos delivered over the Internet.

The maximum annual royalty for an Enterprise is 6.5 M$.
http://www.mpegla.com/main/programs/AVC/Pages/Agreement.aspx

VP8



- Codec developped by On2 technologies acquired by google in 2010
- core part of the WebM project launched by google
  - create an open file format with open source license free codecs.
    - Audio : Ogg Vorbis - Xiph.org
    - Video : VP8 codec.

VP8

- *Google hereby grants to you a perpetual, worldwide, non- exclusive, **no-charge, royalty-free**, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, transfer, and otherwise run, modify and propagate the contents of this implementation of VP8 [...] If you or your agent or exclusive licensee institute or order or agree to the institution of patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that this implementation of VP8 or any code incorporated within this implementation of VP8 constitutes direct or contributory patent infringement, or inducement of patent infringement, then **any patent rights granted to you under this License for this implementation of VP8 shall terminate as of the date such litigation is filed**. »*
- http://www.webmproject.org/license/additional/

H.264/AVC vs VP8

	H.264	VP8
License	pay	Free (until...)
Video Performance	Similar ?	Similar?
HW acceleration	Widely supported	coming (Anthill)

- Risk of market fragmentation.
- slow-down adoption of HTML5.

In the Open source World…

- Several codecs open source (sometimes license free) exist :
  - X.264/AVC (open source implementation)
  - JASPER
  - Dirac (SMPTE VC-1)
  - Theora

X.264 (H.264/AVC)

- Best and most complete H.264/AVC open source implementation
- Core of several (most) video services on the web
  - Cloud Transcoding
  - Dvd and bluRay RIP software
  - Open Broadcast encoder http://www.ob-encoder.com/

Dirac (VC-2)

- Wavelet based codec developed by BBC.
- Standardised by SMPTE as VC-2
- 2 versions :
    - Dirac - for distribution, streaming etc. involves inter-predictions.
    - Dirac Pro - Intra only codec fro professional broadcast apps.
- supported by most available video libraries and players
- http://diracvideo.org/specifications/

Theora

- Former On2 VP3 codec freed and provided to Xipg.org license free.
- DCT (block) based with huffmann coding
- Improved coding efficiency but forward compatible with old VP3 content
- lower quality than existing codecs (H.264/AVC, VP8)
- www.xiph.org
- Download : http://www.theora.org/downloads/

Acceleration GPU/Hardware

- GPU is an dedicated Graphic Processor unit that reduces the CPU computation load by implementing some of the graphic operations.
- Started with 3D graphics (OpenGL)
  - Now common for normal 2D video rendering and video decoding.
  - Not all codec can benefit form GPU acceleration (e.g. VP8 not yet supported by some cards)