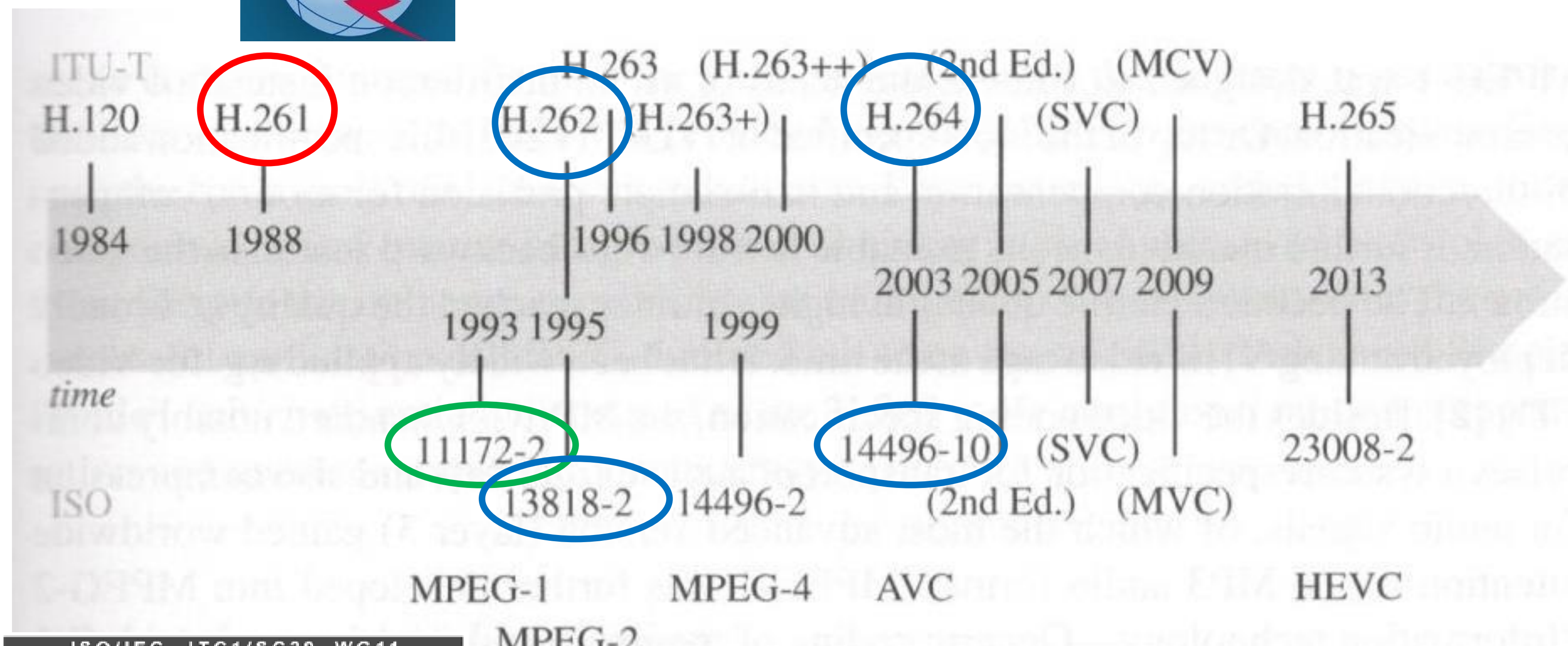
MULTI-APPLICATION VIDEO CODING: TARGETING HD RESOLUTION

Fernando Pereira

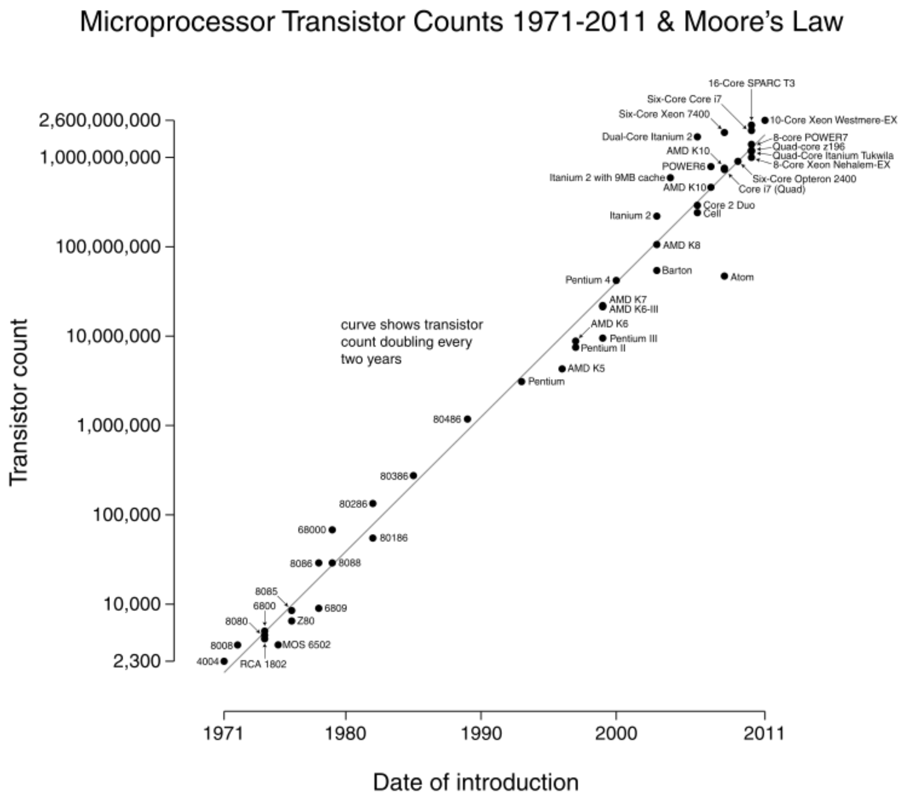
Standards Over Time …



Moore’s Law at Work …

- **“Moore's law” is the observation that, over the history of computing hardware, the number of transistors in a dense integrated circuit doubles approximately every two years (some say 18 months).**
- **The observation is named after Gordon E. Moore, co-founder of the Intel Corporation, who described the trend in 1965.**
- **His prediction has proven to be accurate, in part because the law now is used in the semiconductor industry to guide long-term planning and to set targets for research and development.**

The bottom line is that ‘the affordable complexity increases every day’ …



Video Coding in MPEG-4

There are two Parts in the MPEG-4 standard dealing with video coding:

First version in 1998

- **Part 2: Visual (1998)** – Specifies several coding tools targeting the efficient and error resilient of video, including both arbitrarily and rectangular shaped video; it also includes coding of 3D faces and bodies.

First version in 2003

- **Part 10: Advanced Video Coding (AVC) (2003)** – Specifies more efficient (about 50%) and more resilient frame based video coding tools; this Part has been jointly developed by ISO/IEC MPEG and ITU-T through the Joint Video Team (JVT) and it is often known as H.264/AVC.

Each of these two Parts specifies several profiles with different video coding functionalities and compression efficiency versus complexity trade-offs. H.264/AVC only addresses rectangular frames !

MPEG-4 Advanced Video Coding (AVC), also ITU-T H.264

H.264/AVC: The Objective

2003 !

Coding of frame-based video with increased efficiency: about 50% less rate for the same perceptual quality regarding existing standards such as H.263, MPEG-2 Video and MPEG-4 Visual.

This standard (joint between ISO/IEC MPEG and ITU-T) offers also good flexibility in terms of compression efficiency-complexity trade-offs as well as good performance in terms of error resilience for mobile environments and fixed and wireless Internet (both progressive and interlaced formats).

What Does the Sucessive 50% Gain Mean ?

Quality for same bitrate ...



MPEG-2 Video

MPEG-4 Visual
or Part 2

H.264/AVC
Also MPEG-4 Part 10

TÉCNICO LISBOA

H.264/AVC: Main Functionalities

- **Coding efficiency (about 50% compared to previous standards)**
  - Enhanced motion compensation, multiple reference frames, hierarchical B frames, improved de-blocking filter, smaller blocks for transform, enhanced entropy coding
- **Error resilience (improving error robustness)**
  - Data partitioning, FMO, slices, redundant slices, resynchronization markers, multiple reference pictures, parameter sets, etc.
- **Temporal scalability**
  - Enabled by flexible reference picture management via hierarchical predictions
- **Network friendliness**
  - Enabled through NAL design, bitrate adaptation

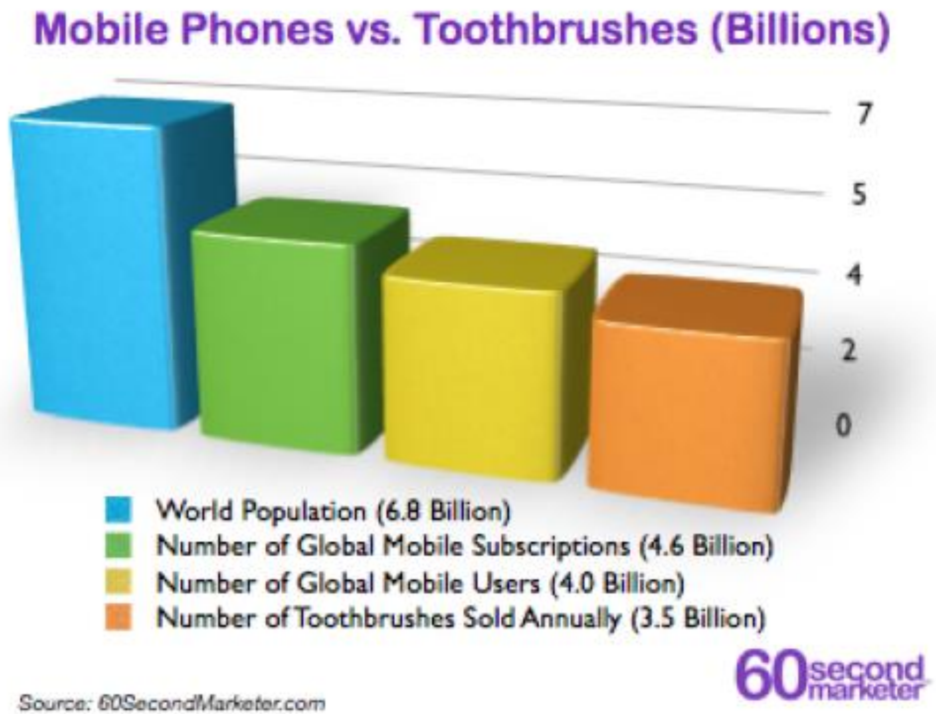
H.264/AVC: Applications

H.264/AVC is the recommended codec for all 3GPP video services.

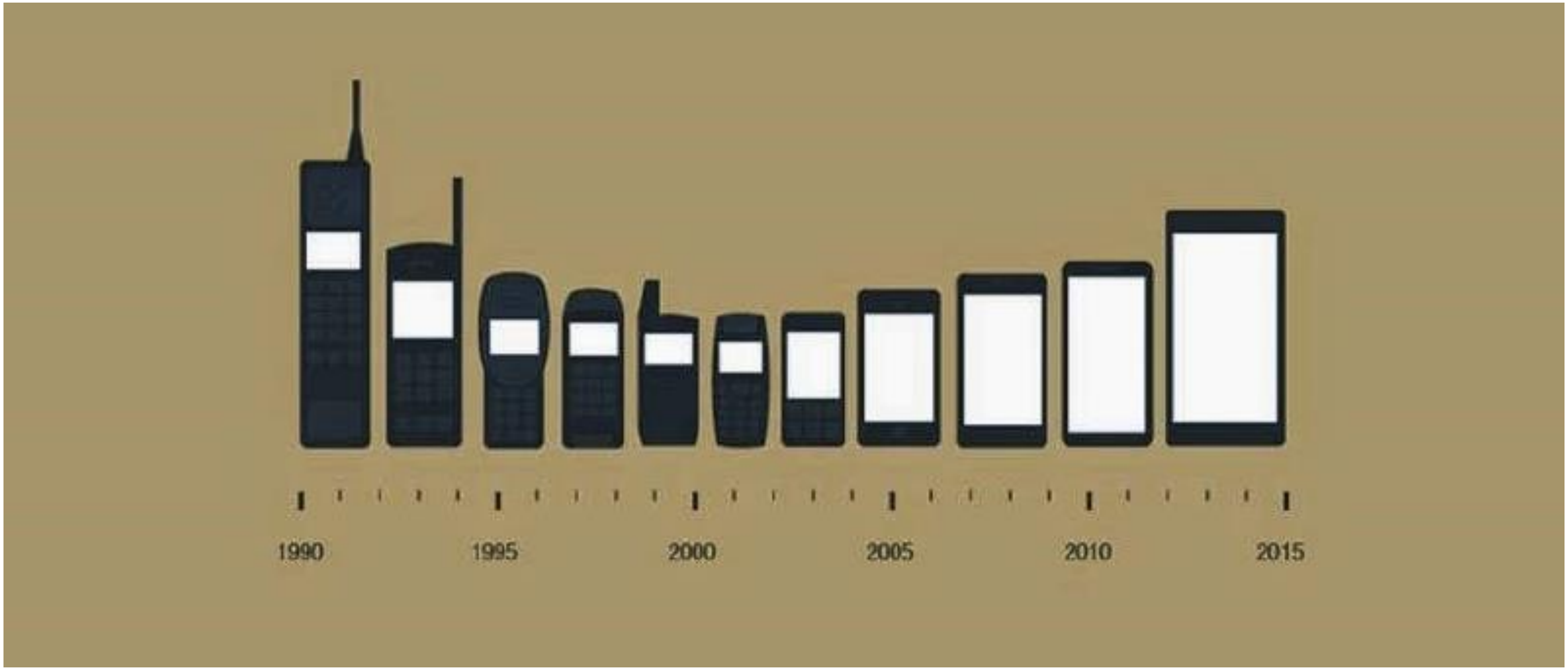
- **Entertainment Video (1-8+ Mbps, higher latency)**
  - Broadcast / Satellite / Cable / DVD / VoD / FS-VDSL / …
  - DVB/ATSC/SCTE, DVD Forum, DSL Forum
- **Conversational Services (usually <1 Mbps, low latency)**
  - H.320 Conversational
  - 3GPP Conversational H.324/M
  - H.323 Conversational Internet/best effort IP/RTP
  - 3GPP Conversational IP/RTP/SIP
- **Streaming Services (usually lower bitrate, higher latency)**
  - 3GPP Streaming IP/RTP/RTSP
  - Streaming IP/RTP/RTSP (without TCP fallback)
- **Other Services**
  - 3GPP Multimedia Messaging Services

A GLOBAL INITIATIVE

Mobile Has Really Exploded …



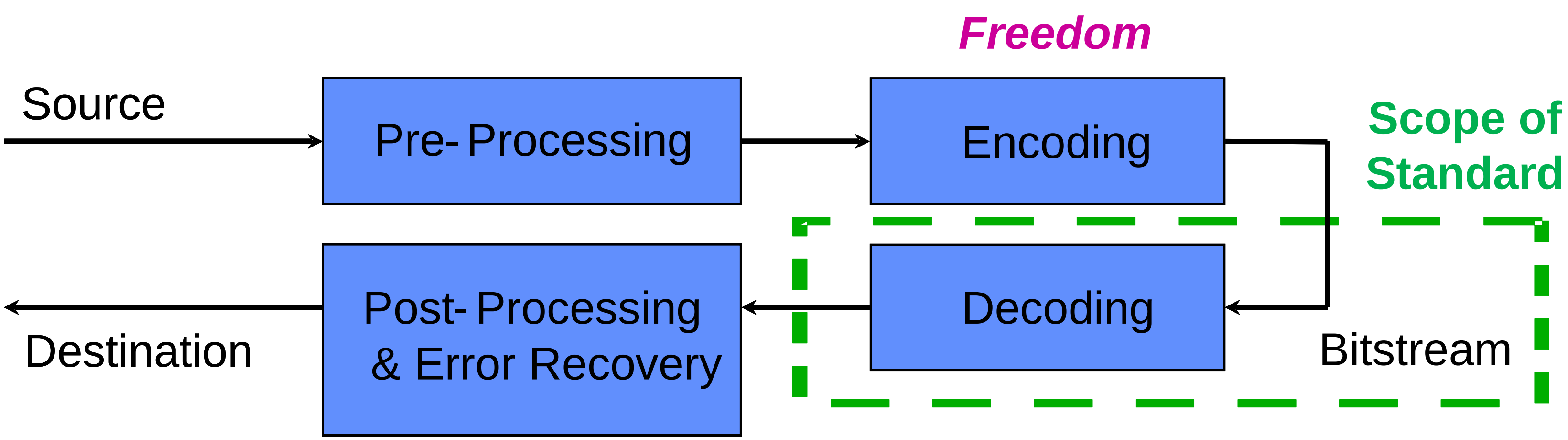
Evolution of Celular Phones ... Video has Won !



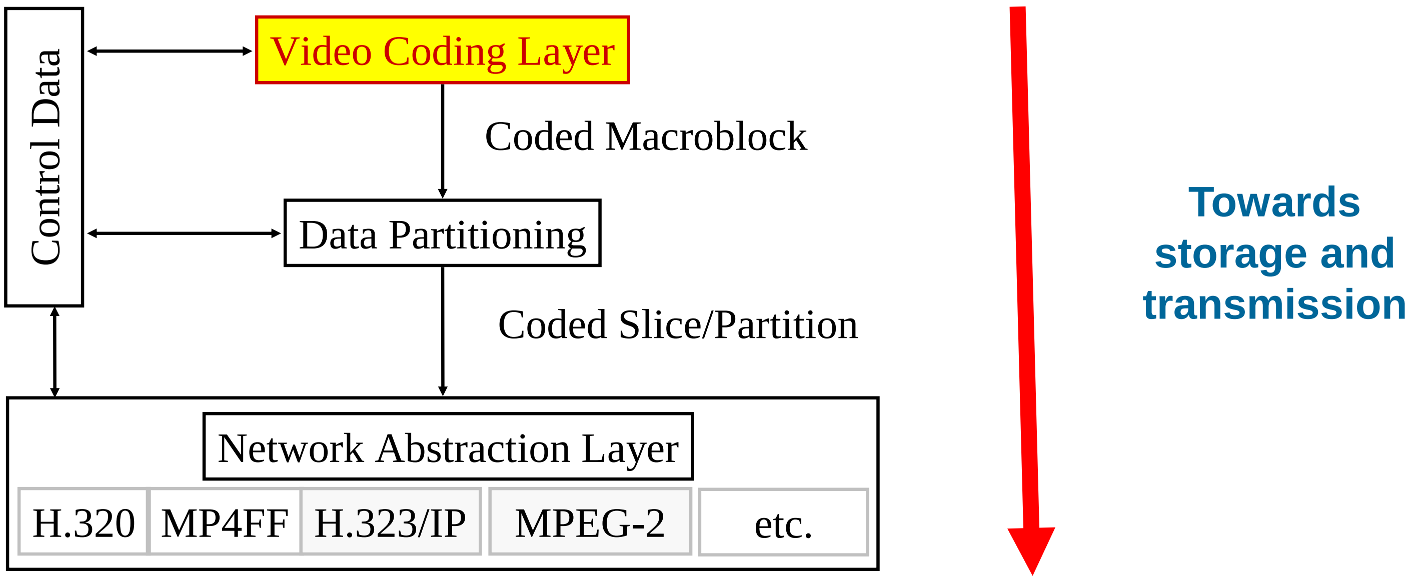
H.264/AVC: The Scope of the Standard

As usual for MPEG standards, H.264/AVC specifies only the bitstream syntax and semantics as well as the decoding process:

- **Allows several types of encoder optimizations**
- **Allows to reduce the encoding implementation complexity (at the cost of some quality)**
- **Does NOT allow to guarantee any minimum level of quality (which depends on the encoder implementations) !**



H.264/AVC Layer Structure



To address the need for flexibility and customizability, the H.264/AVC design covers:

- **A Video Coding Layer (VCL), which is designed to efficiently represent the video content**
- **A Network Abstraction Layer (NAL), which packages the VCL representation and provides header information in a manner appropriate for conveyance by a variety of transport layers or storage media**

The Network Abstraction Layer (NAL)

Streams as NAL Units

Network integration

- **NAL Unit Definition -** The elementary unit processed by an H.264/AVC codec is called the network abstraction layer (NAL) unit. **The coded video data is organized into NAL units, which are packets containing an integer number of bytes**.
- **NAL Unit Encapsulation -** NAL units can be easily encapsulated into different transport protocols and file formats, such as MPEG-2 Systems Transport Stream, Real-Time Transfer Protocol (RTP), and MPEG-4 file format**.**
- **NAL Unit Composition -** A NAL unit starts with a one-byte header, which signals the type of contained data; the remaining bytes represent *payload data*. The header indicates the type of NAL unit and whether a VCL NAL unit is part of a reference or non-reference picture.
- **NAL Unit Classification:**
  - **VCL NAL units** - contain data representing the samples of video pictures in the form of slices or slice data partitions.
  - **Non-VCL NAL units** - contain additional information such as enhancement information (SEI), parameter sets, picture delimiter, or filler data.

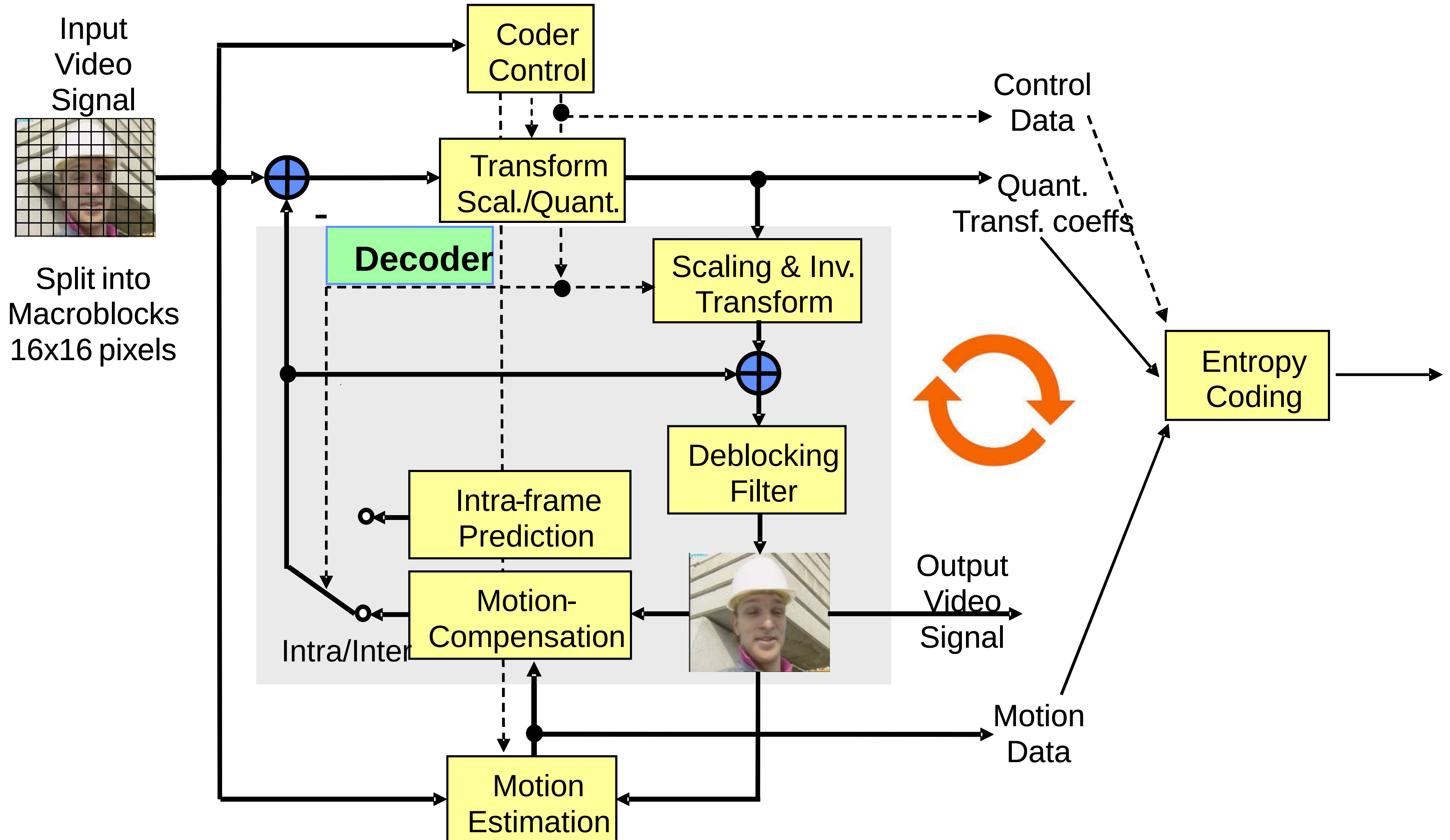
Also Access Units

Network integration

- **Access Unit -** Set of consecutive *NAL units* with specific properties; the decoding of an access unit results in exactly one decoded picture.
- **Coded Video Sequence -** Set of consecutive *access units* with certain properties**.**
- **The most important non-VCL NAL units are parameter sets (PS) and Supplemental Enhancement Information (SEI).**
  - The parameter sets allow sending infrequently changing Sequence (SPS) and Picture (PPS) level information reliably, asynchronously, and in advance of the media stream that contains the VCL NAL units.
  - To be able to change picture parameters, such as picture size, without the need to transmit parameter set updates synchronously to the slice packet stream, the encoder and decoder can maintain a list of more than one SPS and PPS. Each slice header contains a codeword that indicates the SPS and PPS in use.
  - SEI messages are not required for decoding the samples of a video sequence; they rather provide additional information which can assist the decoding process or related processes like bitstream manipulation or display.

The Video Coding Layer (VCL)

H.264/AVC Basic Encoding Architecture: Again Hybrid Coding



Many Similarities to Previous Standards

- Macroblocks: 16×16 luma + 2 × 8×8 chroma samples (starting with 4:2:0 MBs)
- Input: Association of luma and chroma and conventional sub-sampling of chroma (starting with 4:2:0, more later)
- Block motion displacement
- Motion vectors over picture boundaries
- Variable block-size motion
- Block transforms
- Scalar quantization
- I, P, and B coding types

H.264/AVC Compression Gains: Why ?

The H.264/AVC standard is based on the same hybrid (*temporal prediction* + *spatial transform*) coding architecture used for previous video coding standards with some important differences:

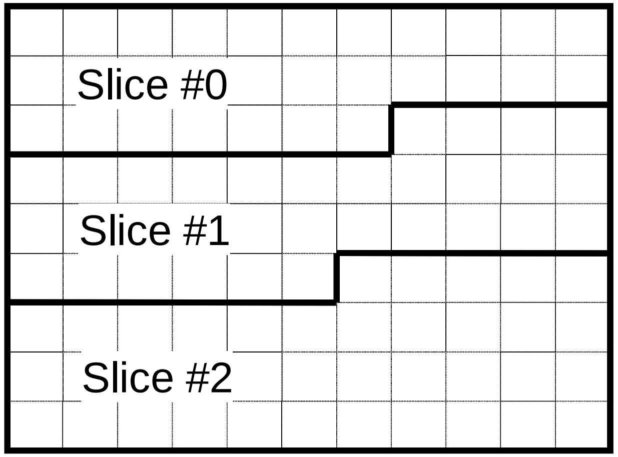
- Variable (and smaller) block size motion compensation
- Multiple reference frames
- Hierarchical transform with smaller block sizes
- Deblocking filter in the prediction loop
- Improved, adaptive entropy coding

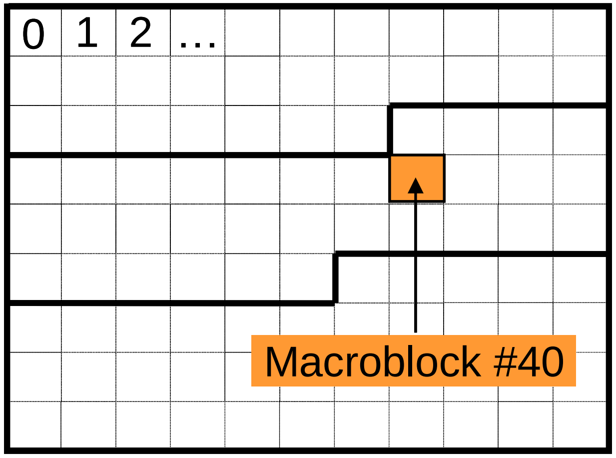
which <u>all together allow achieving substantial gains</u> regarding the bitrate needed to reach a certain quality level.

The H.264/AVC standard addresses a vast set of applications, from personal communications to storage and broadcasting, at various qualities and resolutions.

Partitioning of the Picture

- **Picture (Y,Cr,Cb; 4:2:0 and later more; 8 bit/sample):**
  - A picture (frame or field) is split into 1 or several slices
- **Slice:**
  - Slices are a sequence of macroblocks
  - Slices are self-contained (no predictions between slices)
- **Macroblock (MB):**
  - Basic and most important syntax & processing unit
  - Contains 16×16 luminance samples and 2 × 8×8 chrominance samples (for 4:2:0 content)
  - MBs within a slice depend on each other
  - MBs within a slice are transmitted in raster scan order
  - MBs can be further partitioned

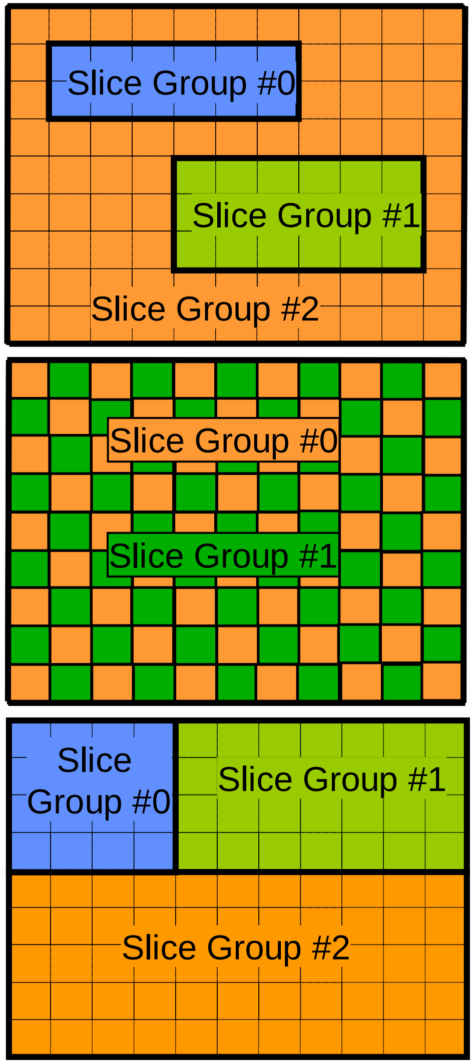




Slices and Slice Groups

Flexible Macroblock Ordering (FMO) allows mapping the MBs into slice groups, where a slice group itself may contain several slices. Therefore, MBs might be transmitted out of raster scan order in a flexible and efficient way.

- **Slice Group:**
  - Pattern of macroblocks defined by a *Macroblock Allocation Map*
  - A slice group may contain 1 to several slices
- **Macroblock Allocation Map Types:**
  - Interleaved slices
  - Dispersed macroblock allocation
  - One or more "foreground" slice groups and a "leftover" slice group
  - Explicitly assign a slice group to each macroblock location in raster scan order



Lossless Macroblock Coding Mode

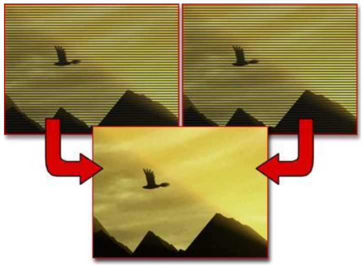
H.264/AVC lossless macroblock coding features include:

- **PCM Mode -** A lossless PCM representation mode in which video samples are represented directly, allowing the perfect representation of specific regions and a strict limit to be placed on the quantity of coded data for each macroblock (*guaranteeing that no MB will ever spend more bits than PCM*).

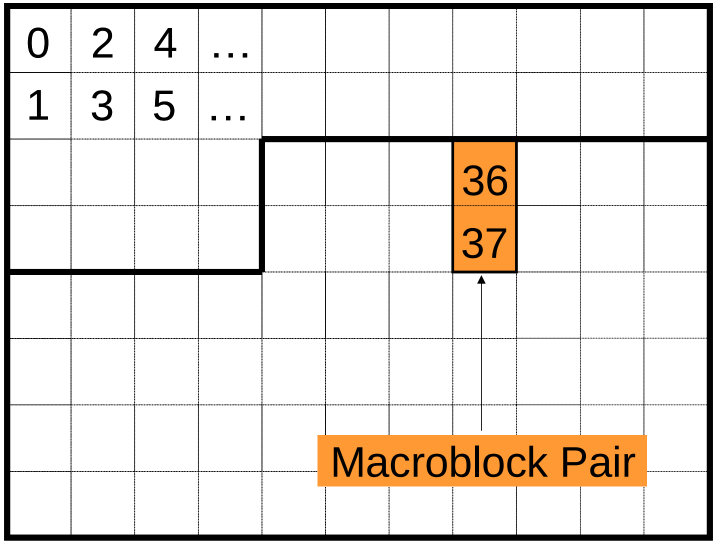
- **Lossless Coding Mode -** An enhanced lossless macroblock representation mode allowing perfect representation of specific regions while ordinarily using substantially fewer bits than the PCM mode.

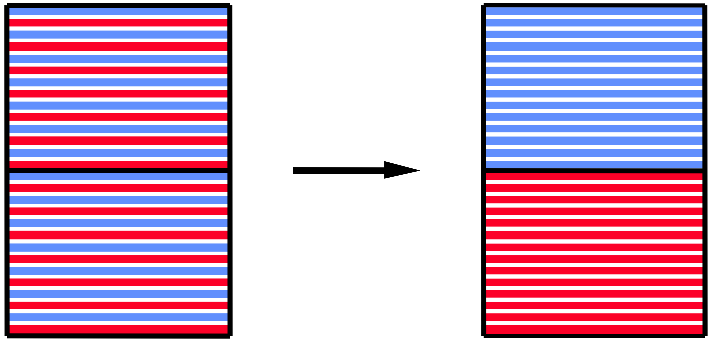
Interlaced Processing

- **Field coding**
  - each field is coded as a separate picture using fields for motion compensation



- **Frame coding**
  - Type 1: the complete frame is coded as a separate picture/frame (as in MPEG-2 Video)
  - Type 2: the frame is scanned as macroblock pairs, for each macroblock pair a switch between frame and field coding is used

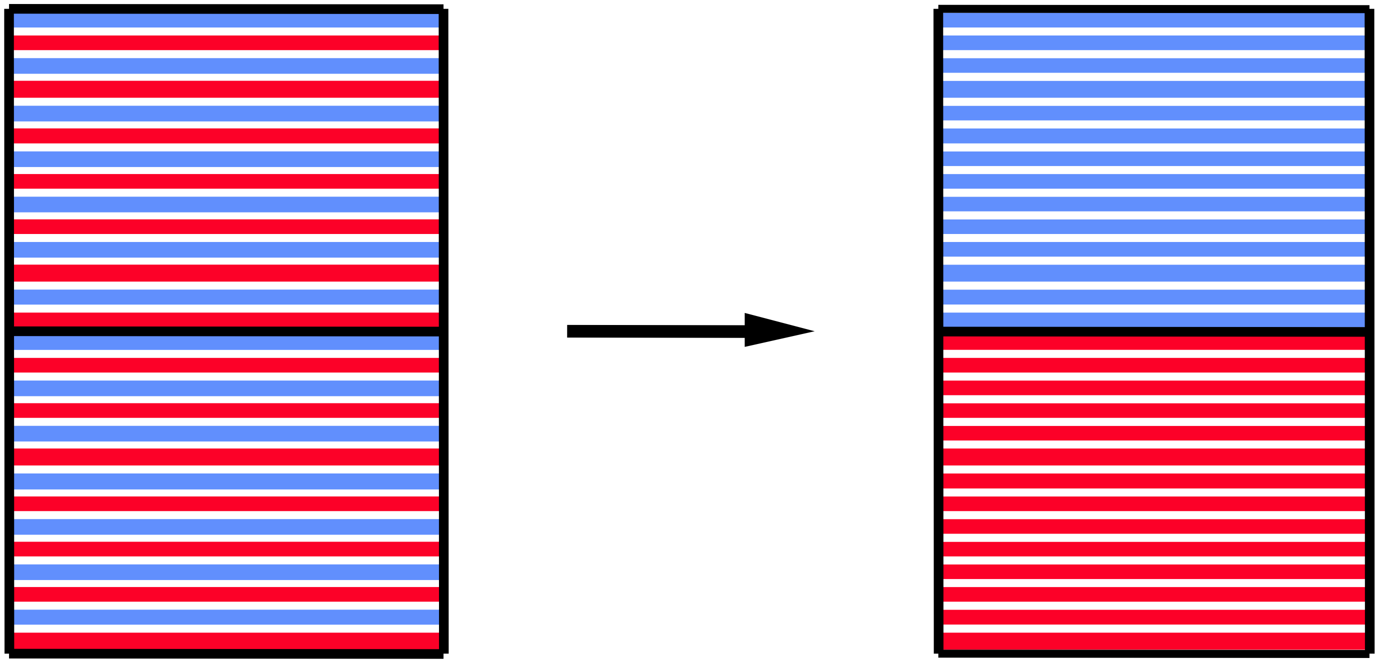




Macroblock-Based Frame/Field Adaptive Coding

Better for more static content

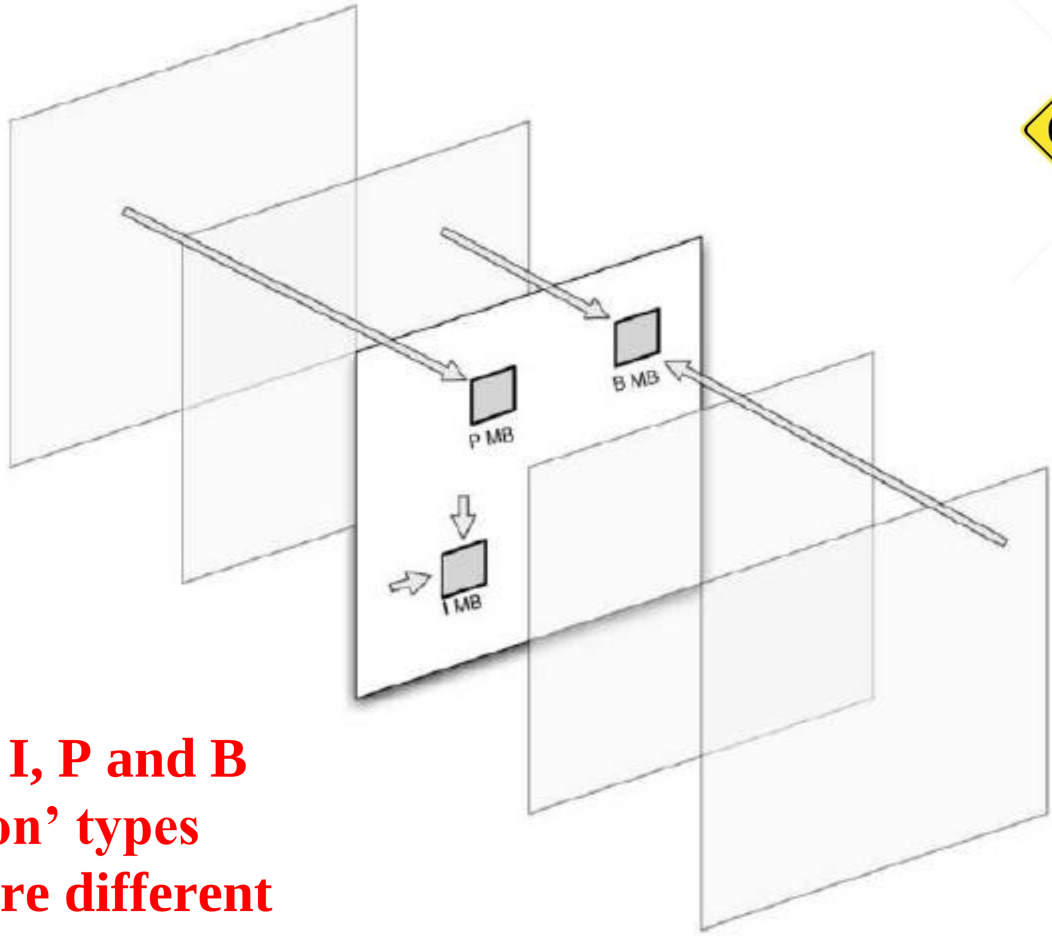
Better for content with motion



A Pair of Macroblocks
in Frame Mode

Top/Bottom Macroblocks
in Field Mode

The ‘Prediction’ Types Trio: I, P and B



Warning: the I, P and B ‘prediction’ types definitions are different from the past !

The New H.264/AVC Coding Tools

TÉCNICO LISBOA

H.264/AVC Main New Tools

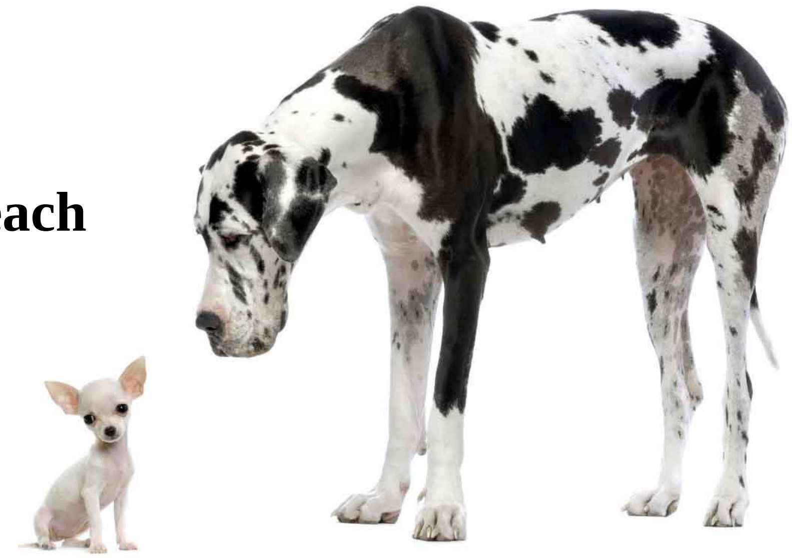
- **Temporal Redundancy**
  - Variable (and smaller) block size motion compensation
  - Multiple reference frames
  - Generalized B frames
- **Spatial Redundancy**
  - Intra prediction
  - Hierarchical transform with smaller (4×4) block sizes
  - *Deblocking filter in the prediction loop (also temporal)*
- **Irrelevancy**
  - Quantization
- **Statistical Redundancy**
  - Improved, adaptive entropy coding: CAVL and CABAC

Exploiting Temporal Redundancy

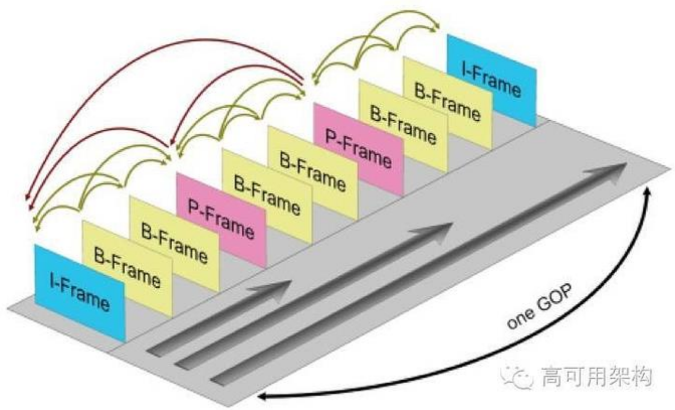
Temporal Redundancy Reduction Tools

The main target of the temporal redundancy reduction tools is to

- Minimize the temporal prediction error, this means
- Minimizing the information to transmit, thus
- Minimizing the rate necessary to reach a target quality



Slice Inter Coding Modes …
Same Names, Different Definitions



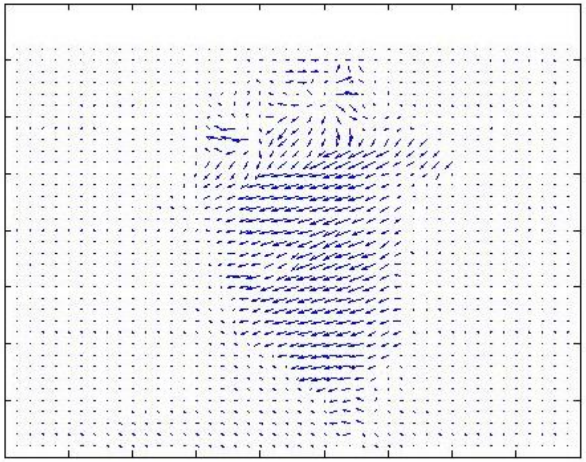
Past

Now

- **P Slices -** A slice in which some macroblocks can also be coded using Inter (temporal) prediction with at most one motion-compensated prediction signal per prediction block (past or future).

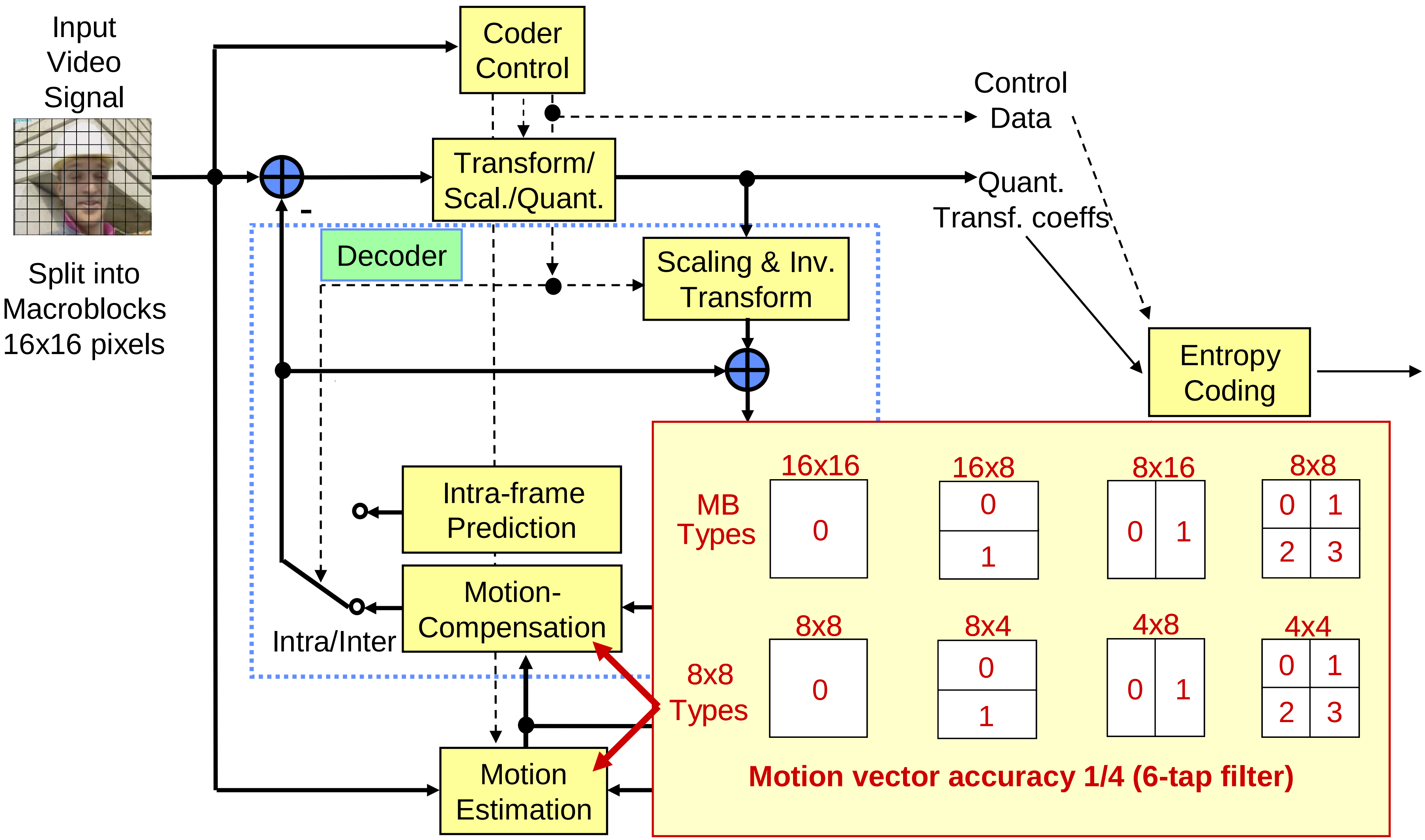
- **B Slices** - A slice in which some macroblocks can also be coded using Inter (temporal) prediction with two motion-compensated prediction signals per prediction block (both past, both future or one past and one future).

Motion is Important ... But Irregular ...

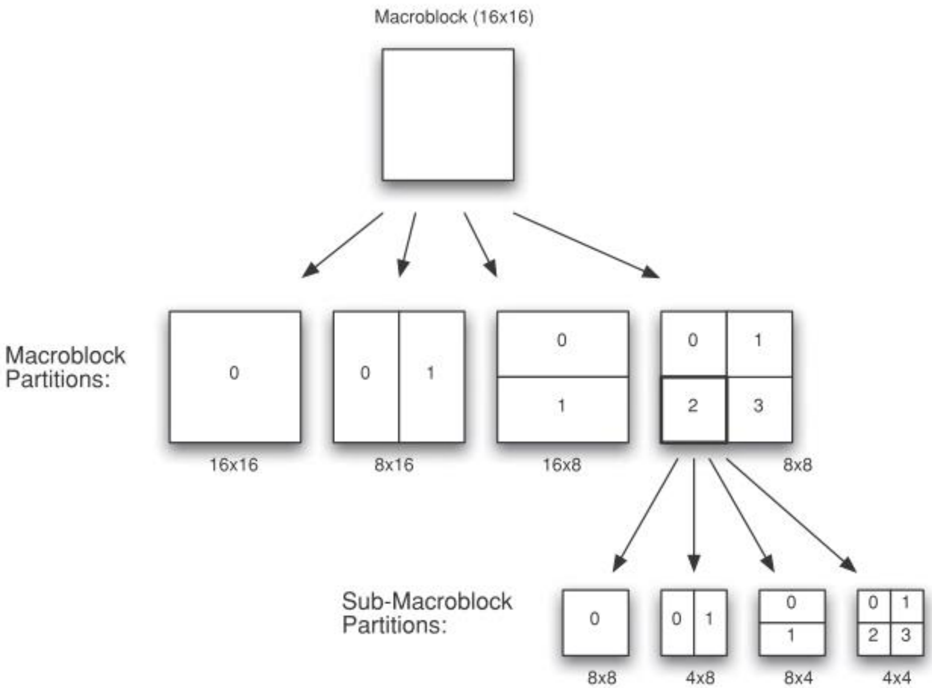


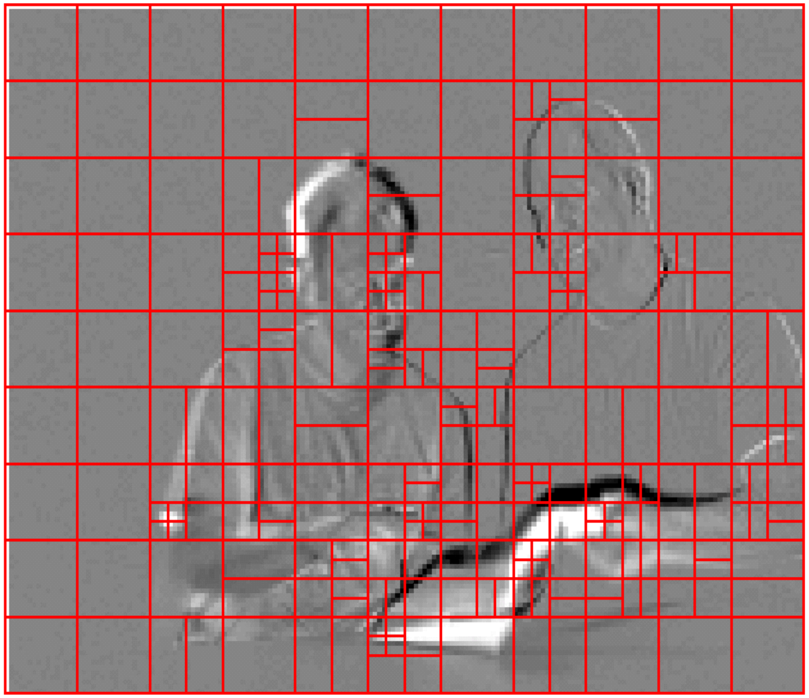
- Motion may be very irregular within a frame ...
- Motion is represented with a translational motion vector, thus unable to represent efficiently non-translational motion !
- Real motion is very diverse ! It is essential to extract the motion field as reliably as possible ...
- Some large parts have rather constant motion while other parts have highly varying motion ...

Variable Block- Size Motion Compensation



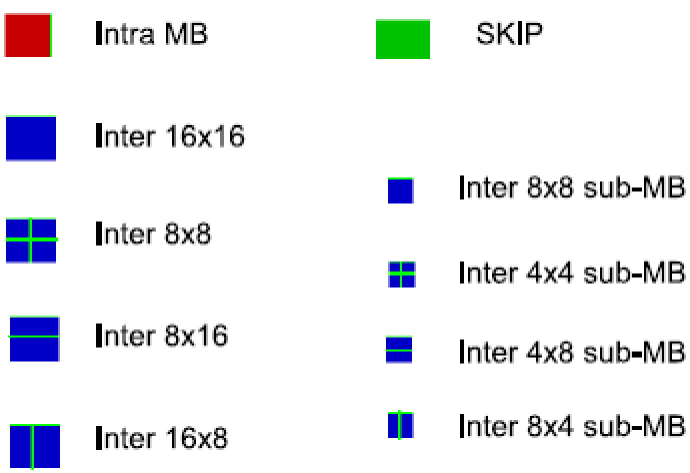
MBs and sub-MBs Partitioning for Motion Compensation

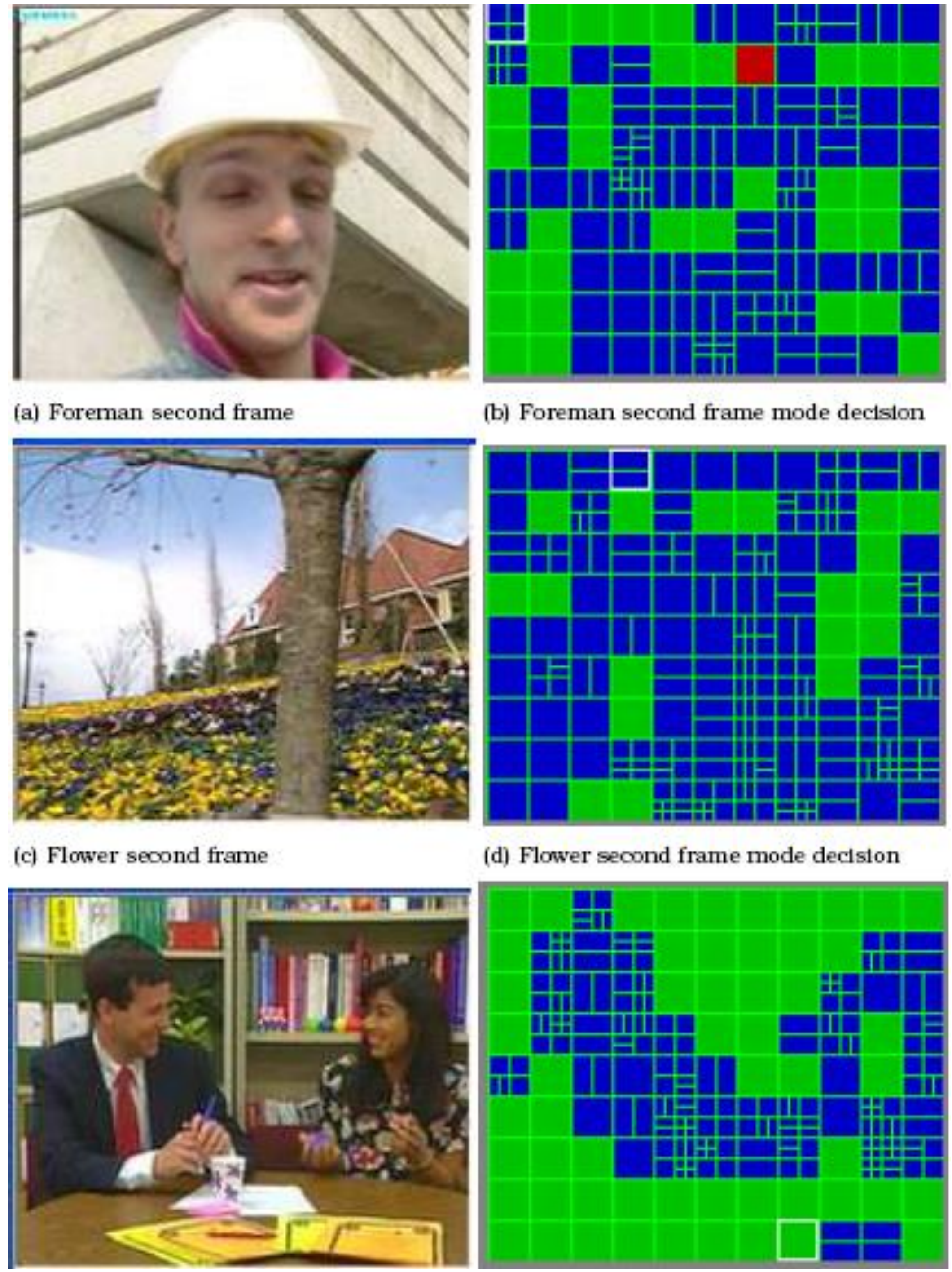




Adaptively investing bits on motion vectors for each macroblock <u>if the prediction residue reduction returns the investment</u> !

Motion vectors are differentially coded but not across slices.





(a) Foreman second frame (b) Foreman second frame mode decision

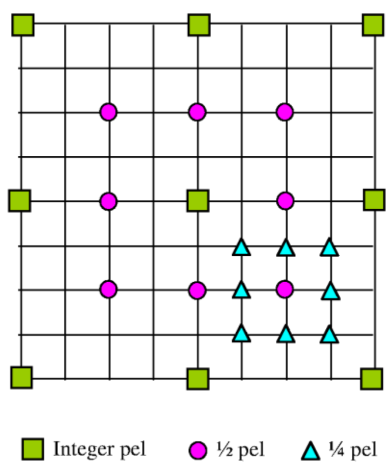
(c) Flower second frame (d) Flower second frame mode decision

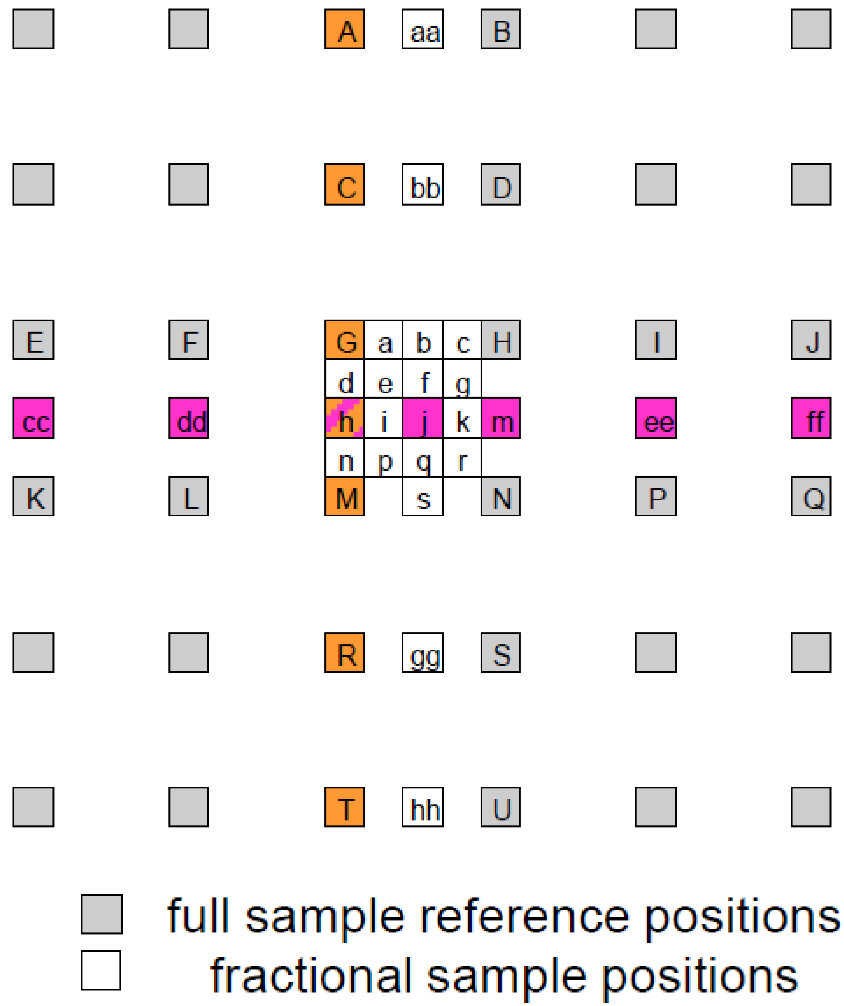
Half- and Quarter-Sample Luma Interpolations

- **Half-sample positions are obtained by applying a 6-tap filter with tap values:**

  (1, -5, 20, 20, -5, 1)

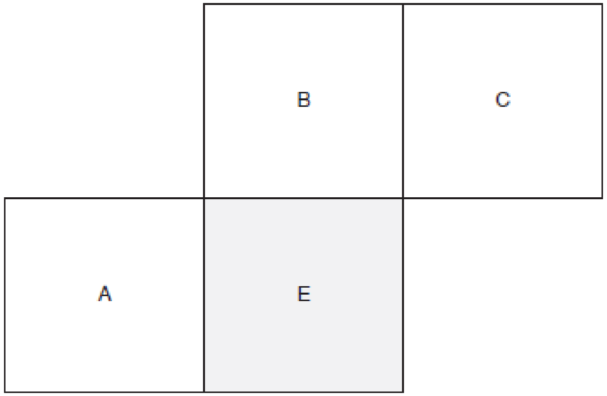
- **Quarter sample positions are obtained by averaging samples at integer and half sample positions**

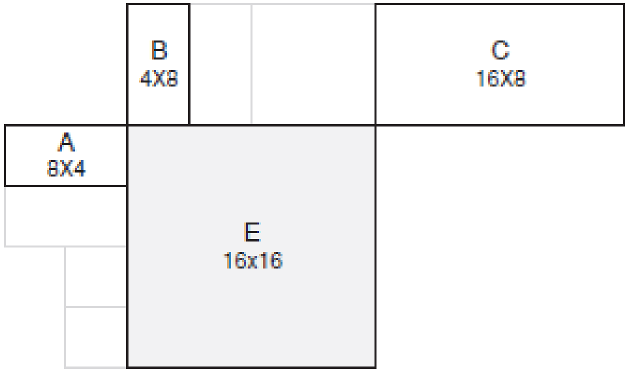




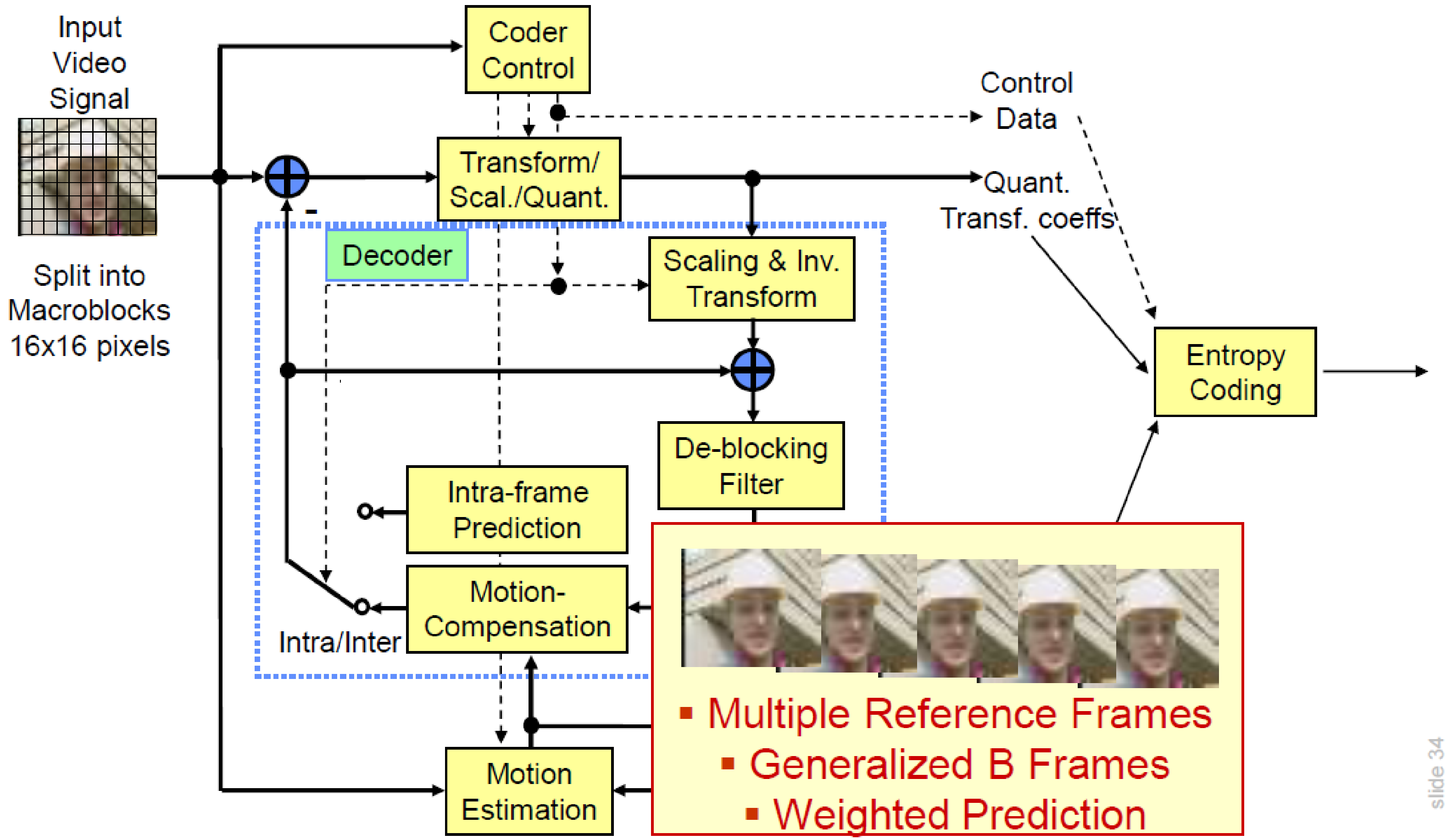
Motion Vector Coding: Predictions and Residues

- **Motion vectors for neighbouring partitions are often highly correlated and so each motion vector is predicted from vectors of nearby, previously coded partitions.**
- **A predicted vector, *MVp*, is formed based on previously calculated motion vectors and after *MVD*, the difference between the current vector and the predicted vector, is encoded and transmitted.**
- **The method of forming the prediction *MVp* depends on the motion compensation partition size and the availability of nearby vectors.**



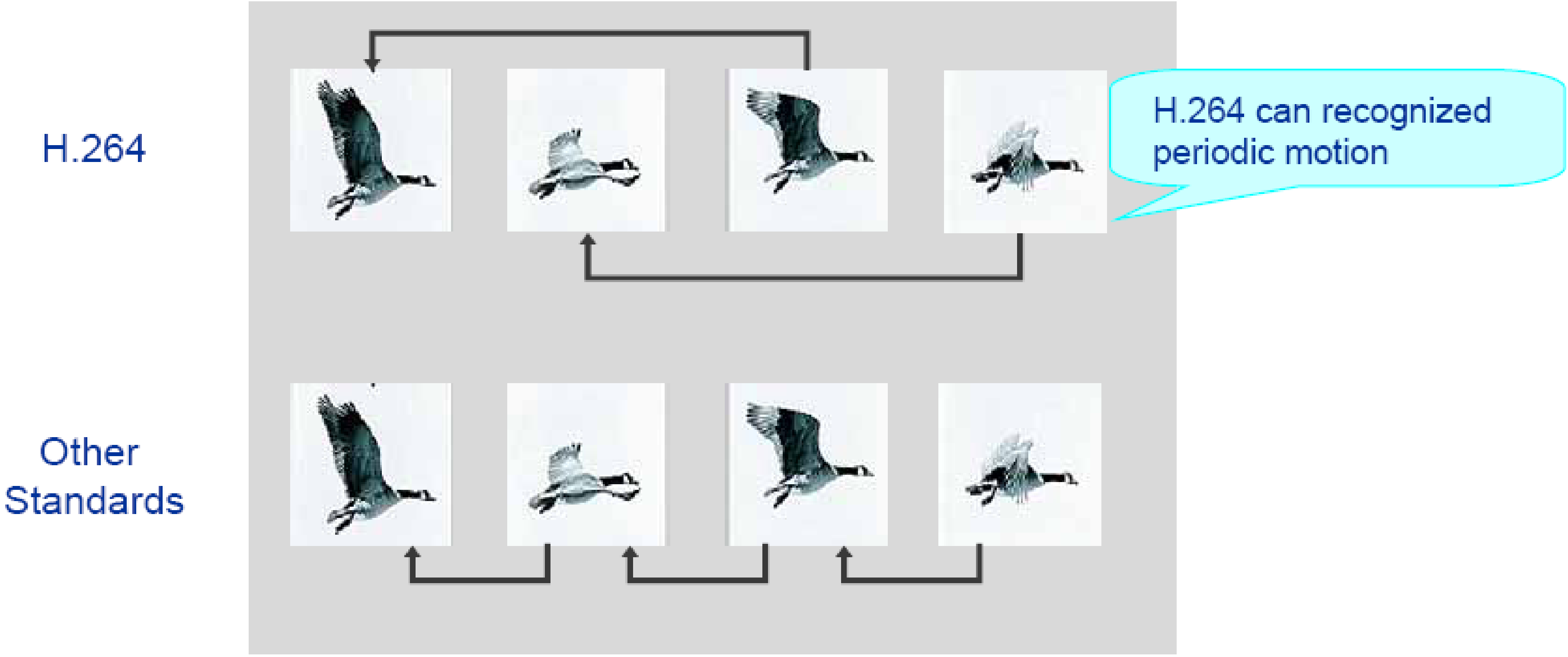


Multiple Reference Frames …

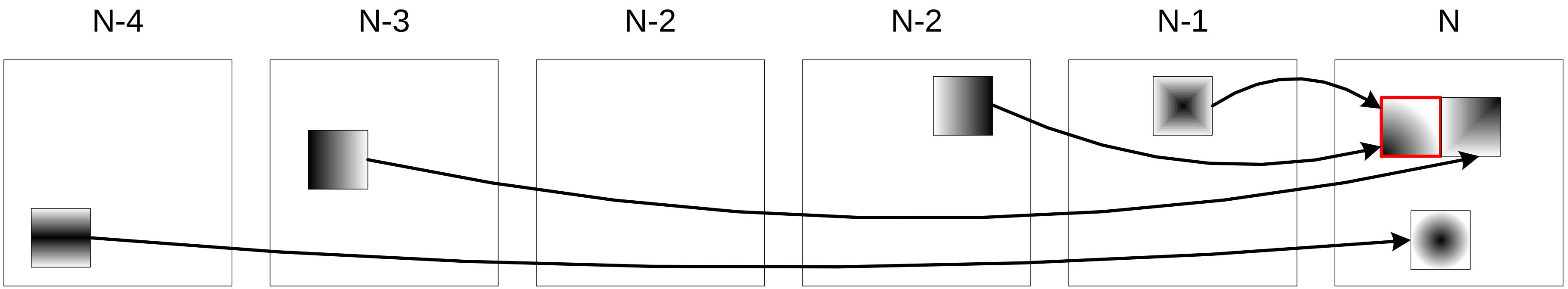


slide 34

The Benefits of Multiple Reference Frames



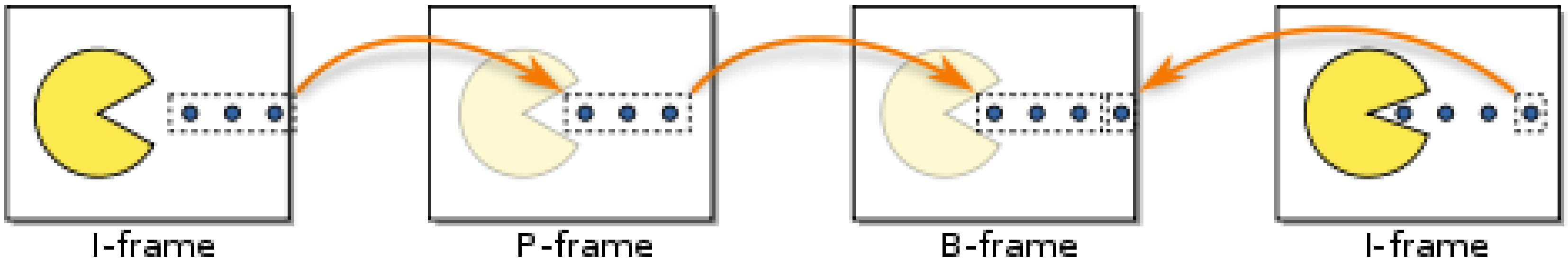
Multiple Reference Frames

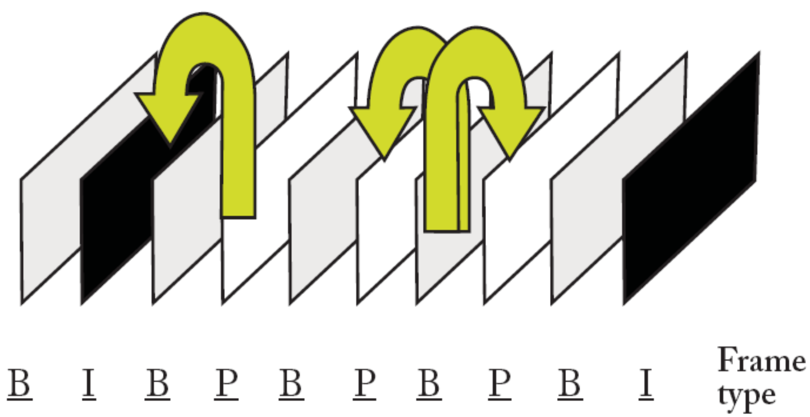


The H.264/AVC standard supports motion compensation with multiple reference frames, meaning that more than one previously (past or future) coded picture may be simultaneously used as prediction reference for the motion compensation of the MBs in a picture (at the cost of memory and computation).

- **Both the encoder and the decoder store the reference frames in a memory with multiple frames; up to 16 reference frames are allowed.**
- **<u>The decoder stores in the memory the same frames as the encoder</u>; this is guaranteed by means of memory control commands which are included in the coded bitstream.**

Conventional B-Frames: the Past ...





The rigid prediction rules in MPEG-1 and MPEG-2 (*notably B-frames cannot provide predictions*) prevent the usage of many successive B frames without damaging the compression efficiency.

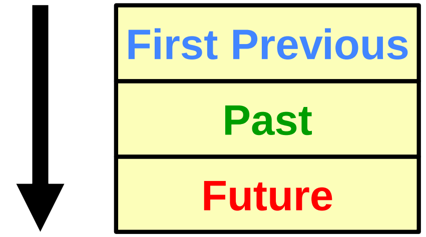
Generalized B Frames

The B frame concept is generalized in H.264/AVC since now any frame may be used as prediction reference for motion compensation (<u>also the B frames</u>); this means the selection of the prediction frames only depends on the memory management performed by the encoder.

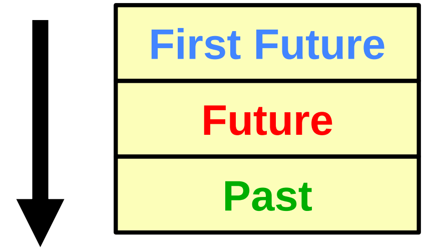
- **<u>B-slices may use two reference frames</u>, referred as the first and second reference frames, picked from the so-called *list 0* and *list 1*.**
- **The selection of the two reference frames to use <u>depends on the encoder</u>.**
- **Blocks or MBs may be coded using <u>a weighted prediction of two blocks or MBs</u> in two reference frames, both in the past, both in the future, or one in the past and another in the future.**
- **The weighted prediction allows to reach a <u>more efficient Inter coding</u> this means with a lower prediction error.**

List 0 and List 1 Elements: B-Slices

- **<u>List 0 gives preference to the past</u> and contains:**
  - First, the <u>first previous frame</u> in display order
  - After, the <u>following past frames</u>
  - The <u>future frames</u>



- **<u>List 1 gives preference to the future</u> and contains:**
  - First, the <u>first future frame</u> in display order
  - After, the <u>following future frames</u>
  - The <u>past frames</u>



- **As the position of the selected reference frame within each list has to be signaled, the positioning above is critical in terms of rate spending.**

- **It is also possible to leave a certain frame for an undetermined time in List 0.**

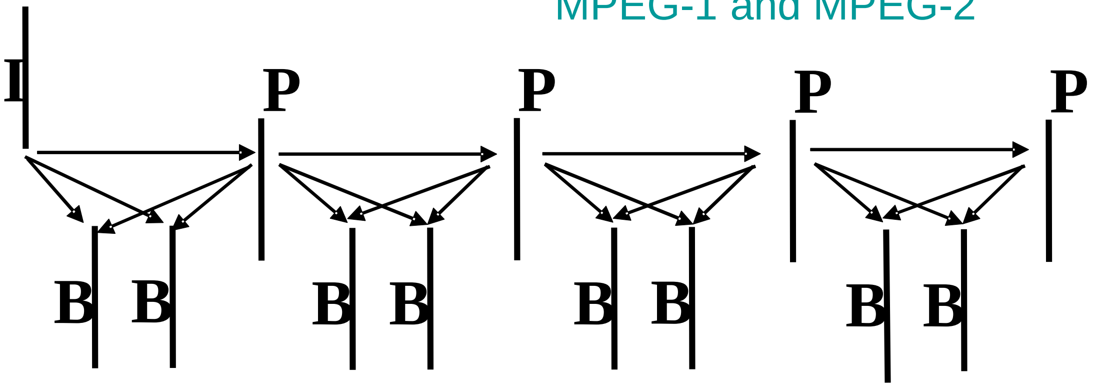
New Types of Temporal Referencing

1. ***Rigid*** **dependencies, e.g. MPEG-1 Video, MPEG-2 Video, etc.**

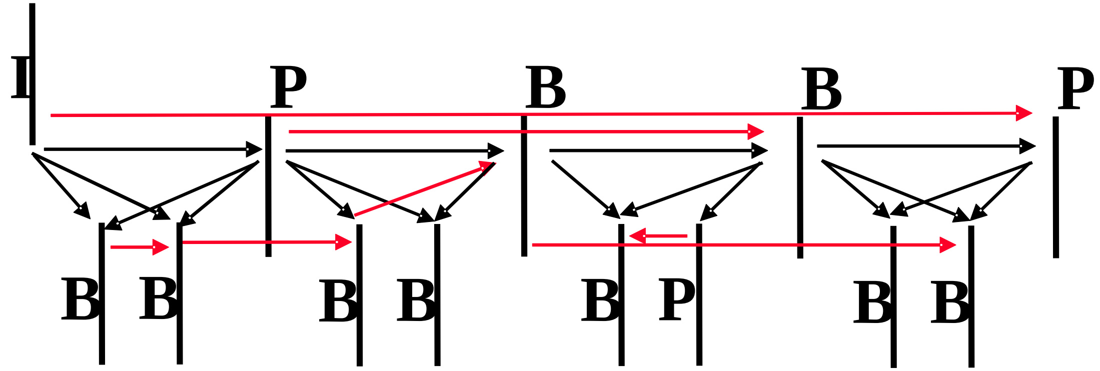
2. ***H.264/AVC new types of dependencies:***

- Referencing order and display order are decoupled, e.g. a P frame may not use as its prediction the previous I/P frame
- Referencing ability and picture type are decoupled, e.g. it is possible to use a B frame as reference

Conventional B-frames, e.g. MPEG-1 and MPEG-2

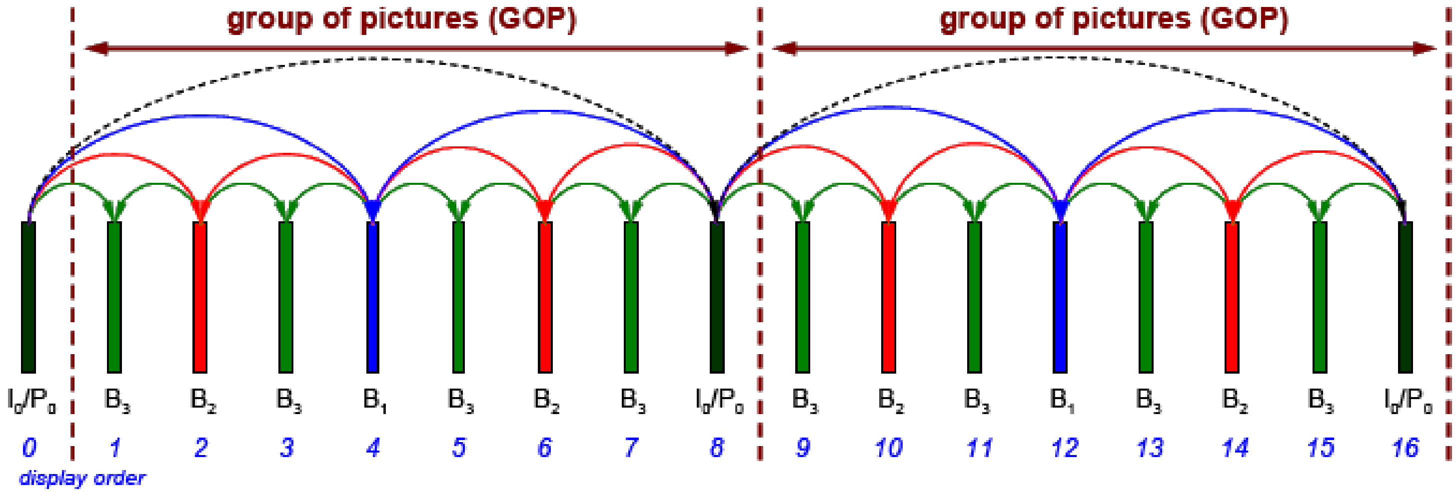


Free-style B-slices, e.g. H.264/AVC

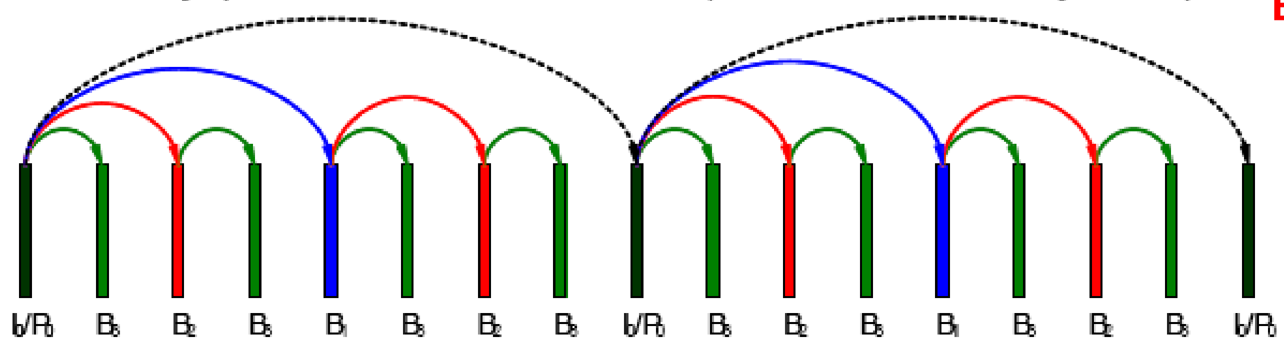


Hierarchical Prediction Structures

- Dyadic temporal scalability

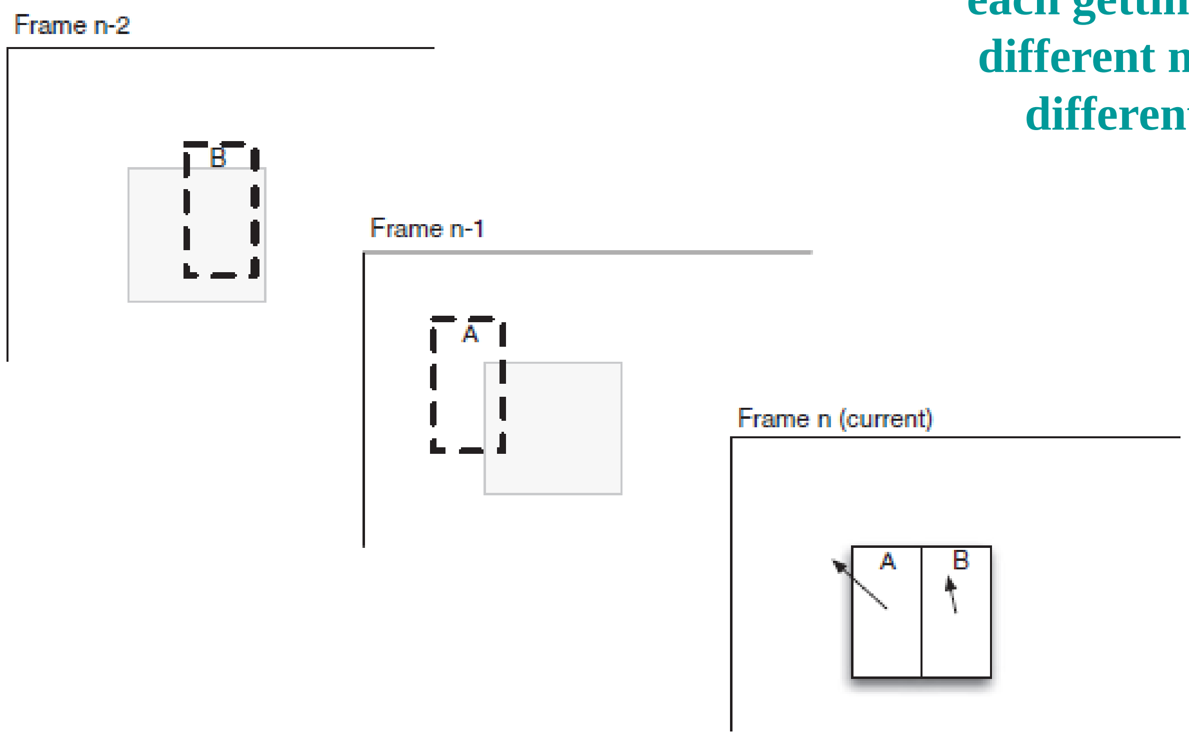


- Low-delay prediction structure (structural delay is 0)



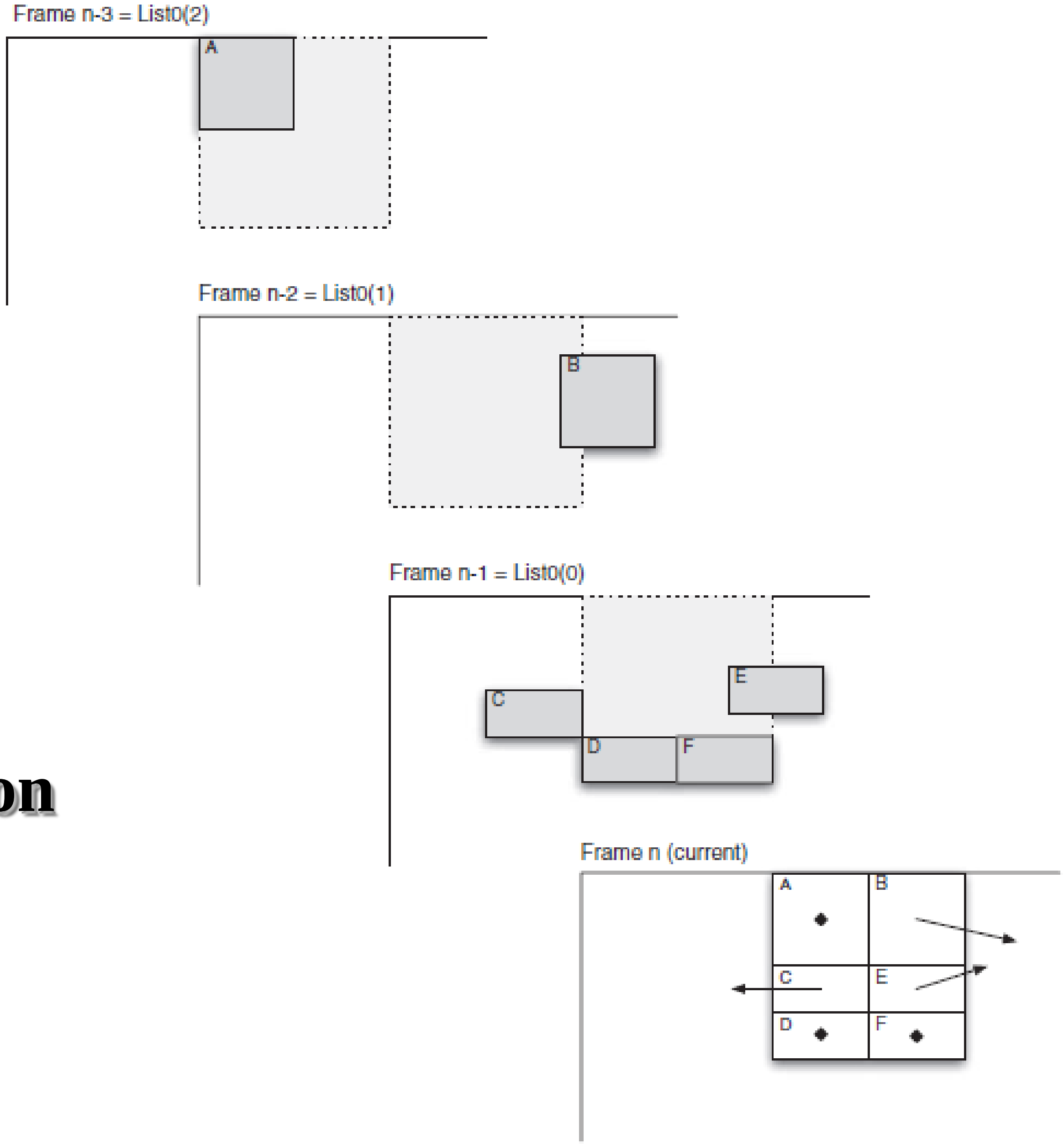
B-MBs may use 2 predictions, eventually both from the past !

P-MB Prediction Example

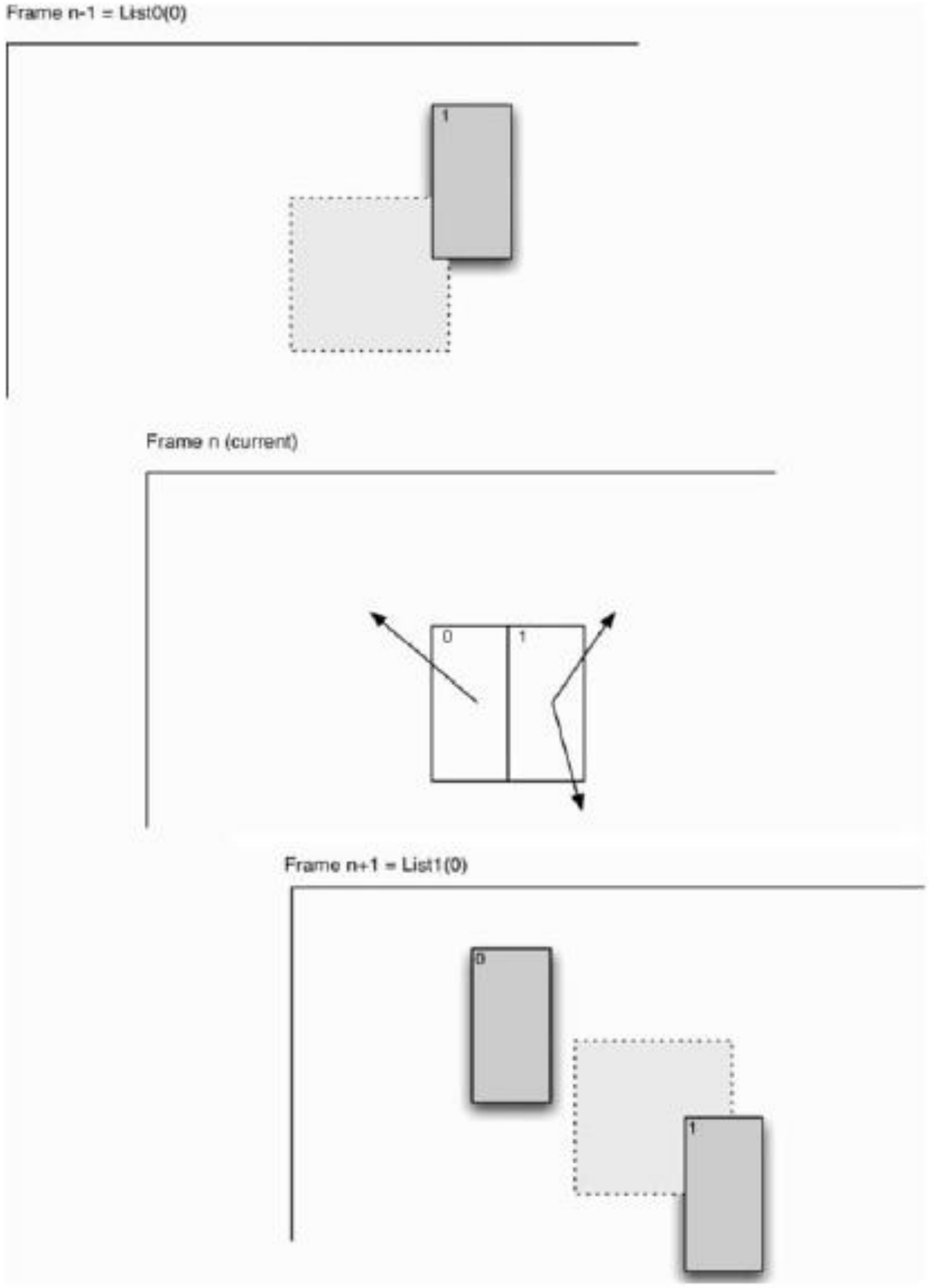


In this case, the P-MB is divided into 2 vertical halves, each getting a prediction with a different motion vector from a different reference frame.

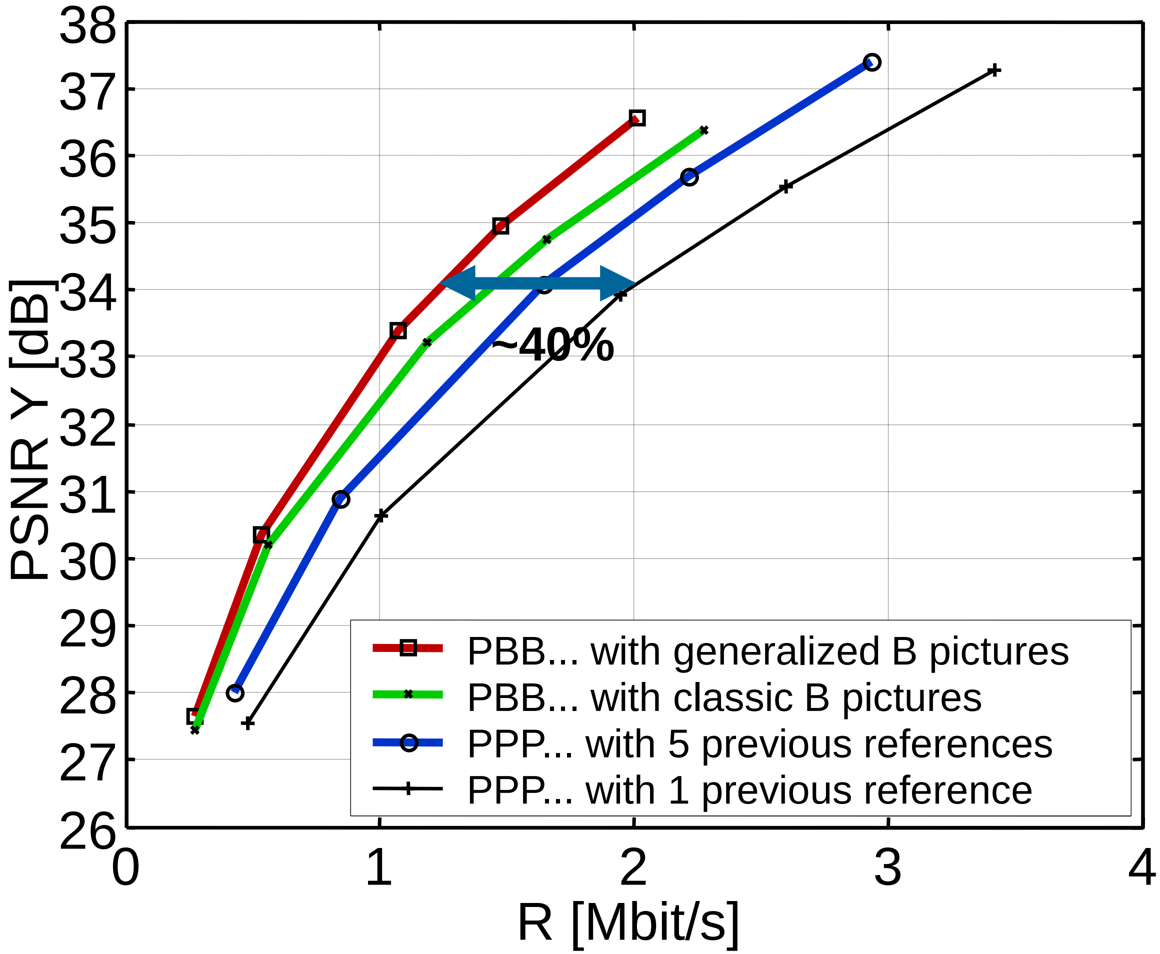
P-MB Prediction Example



B-MB Prediction Example



Comparative Performance: Mobile & Calendar, CIF, 30 Hz



Exploiting Spatial Redundancy

Spatial Redundancy Reduction Tools

The main target of the spatial redundancy reduction tools is to exploit the spatial correlation in

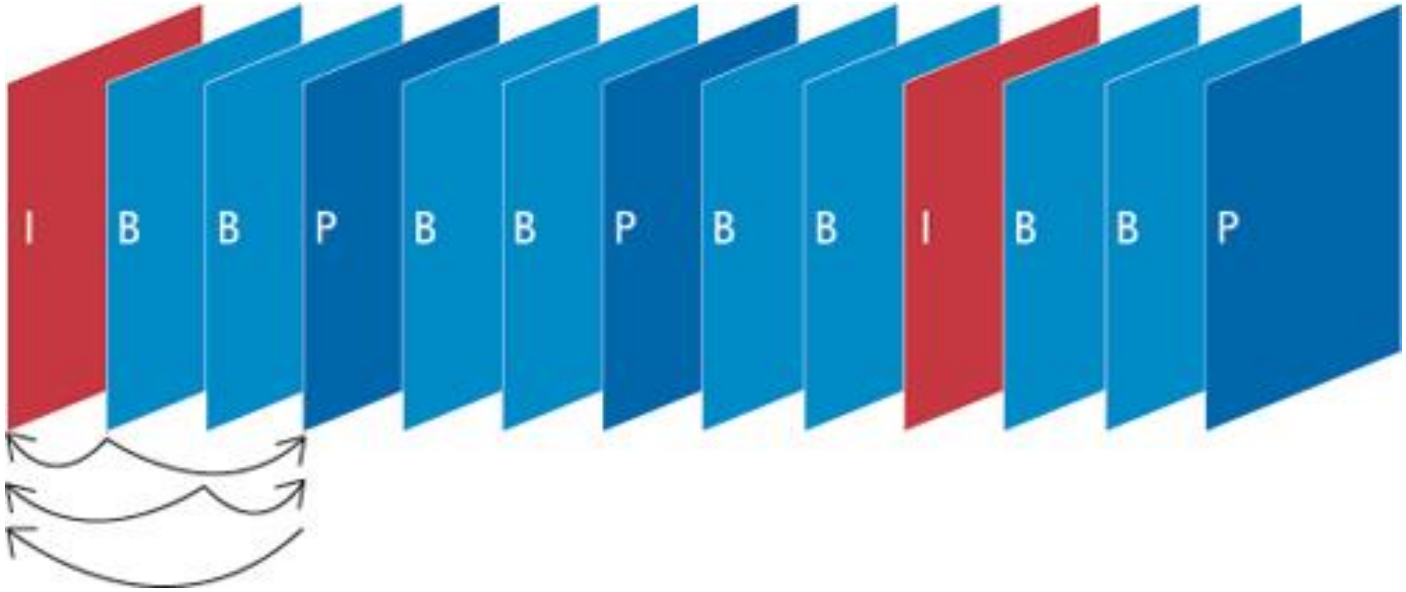
1. Intra macroblocks neighborhoods
2. Intra and Inter macroblocks prediction error

thus

- Further minimizing the information to transmit, and thus
- Minimizing the rate necessary to reach a target quality

TÉCNICO LISBOA

The Weigth of Intra Frames ...



- **Intra frames are essential for random access ...**
- **Intra macroblocks are rate expensive as they don't exploit as much redundancy as they should ... notably not the temporal redundancy ...**

Intra Coding in Two Flavors …

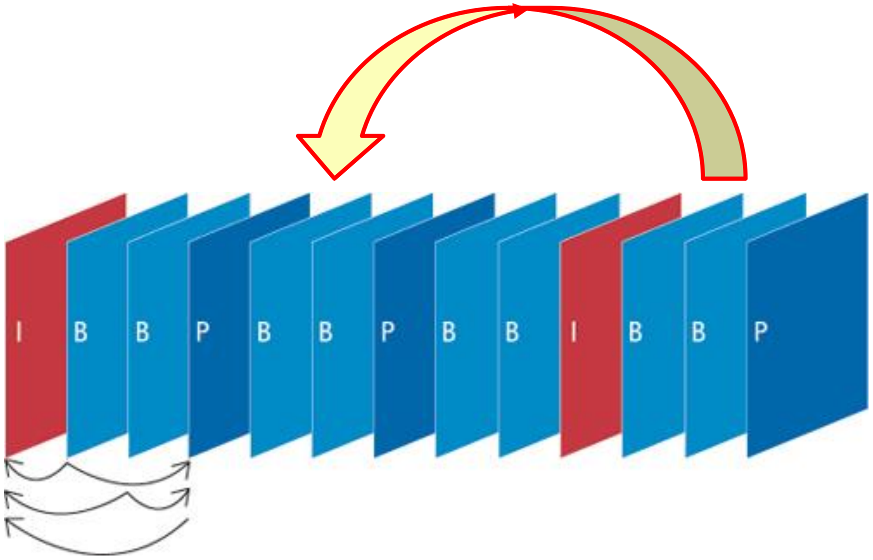
Intra (I) Slices - A slice in which all macroblocks are coded using Intra coding this means exploiting only spatial redundancy:

- **Intra pictures –** Pictures where all slices include only Intra coded macroblocks. Intra pictures do not necessarily provide some forms of random access, e.g. zapping, as in H.264/AVC pictures before Intra pictures may be used as reference for predictively coded Inter pictures coming after them.

- **Instantaneous Decoding Refresh (IDR) pictures -** Pictures where all slices include only Intra coded macroblocks AND the reference pictures buffer is emptied when they happen. This guarantees that after the decoding of an IDR picture all following coded pictures in decoding order can be decoded without access to any picture decoded prior to the IDR picture.

Regular Intra versus IDR

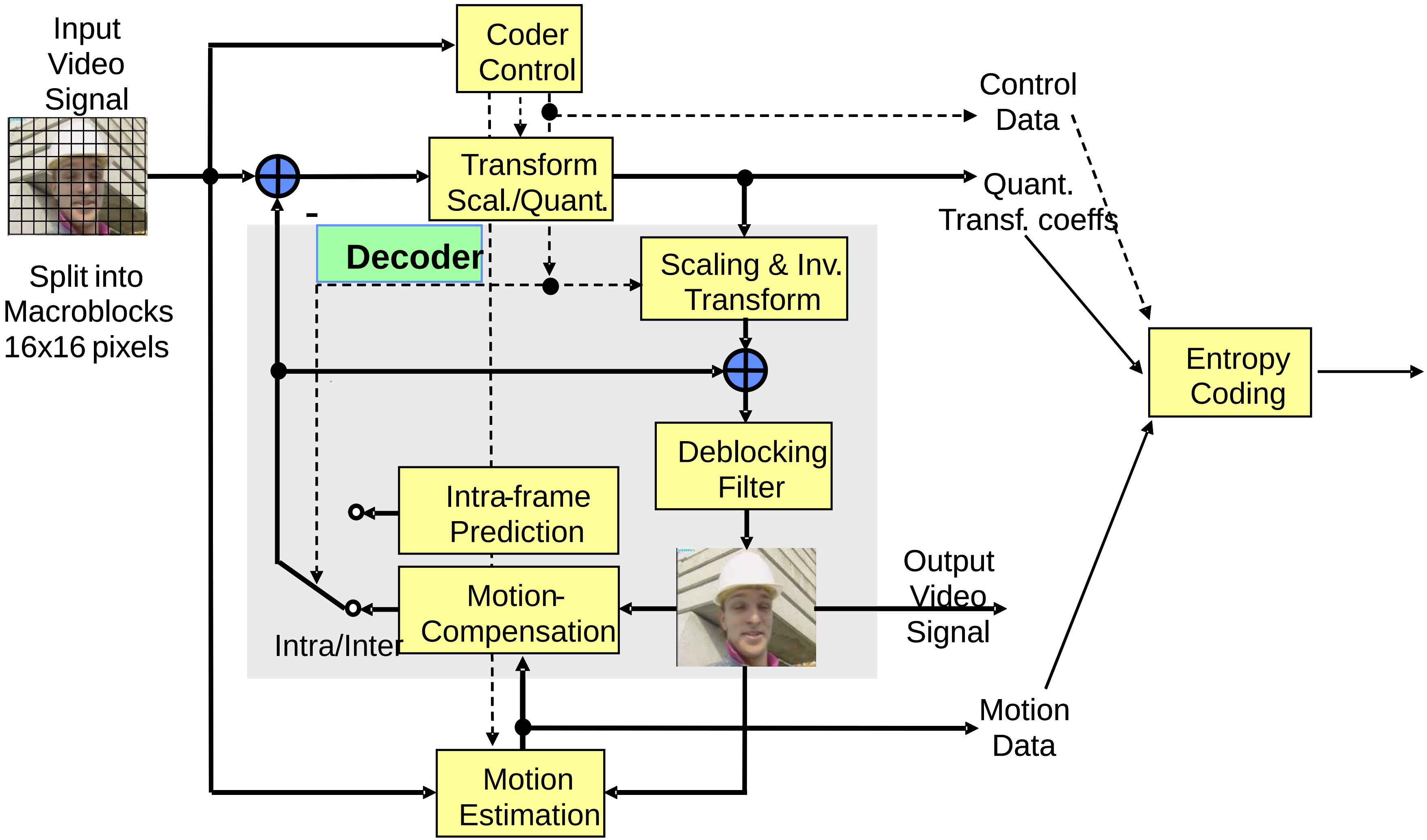
Regular H.264/AVC I-frames



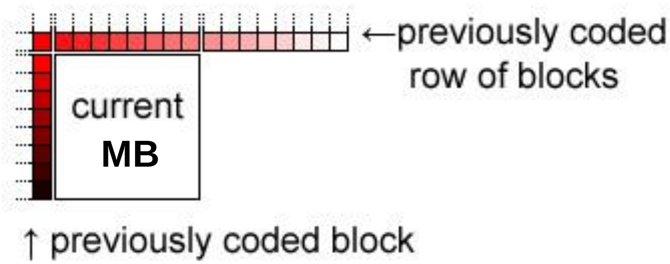
Instantaneous Decoding Refresh Frame
(as I-frames in MPEG-1 and MPEG-2)

Intra Prediction

H.264/AVC Basic Encoding Architecture



H.264/AVC: Intra Prediction



To increase the Intra coding compression efficiency, it is possible to exploit, for each macroblock (MB), the correlation with adjacent information in the same picture.

- If a MB is Intra coded, a prediction MB is built based on the previously coded and decoded blocks or MBs in the same picture.
- The prediction MB is subtracted from the MB currently being coded to create the **Intra prediction residue**.
- To guarantee slice independency, only samples from the same slice can be used to form the Intra prediction.

This type of Intra coding may imply error propagation if the prediction uses adjacent MBs which have been Inter coded; this may be solved by using the so-called *Constrained Intra Coding Mode* where only adjacent Intra coded MBs are used to form the Intra prediction.

H.264/AVC: Intra Prediction Types

Intra predictions may be performed in several ways:

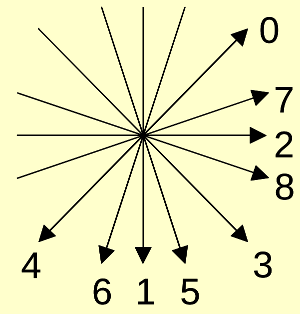
1. **Luminance Intra16×16 -** Single prediction for the whole MB: four modes are possible (vertical, horizontal, DC e planar) -> *more appropriate for uniform areas !*

2. **Luminance Intra4×4 -** Different predictions for the 16 samples of the several 4×4 blocks in a MB: nine modes (DC and 8 directional modes -> *more appropriate for textured areas !*

3. **Chrominance -** Single prediction with four modes (vertical, horizontal, DC and planar)

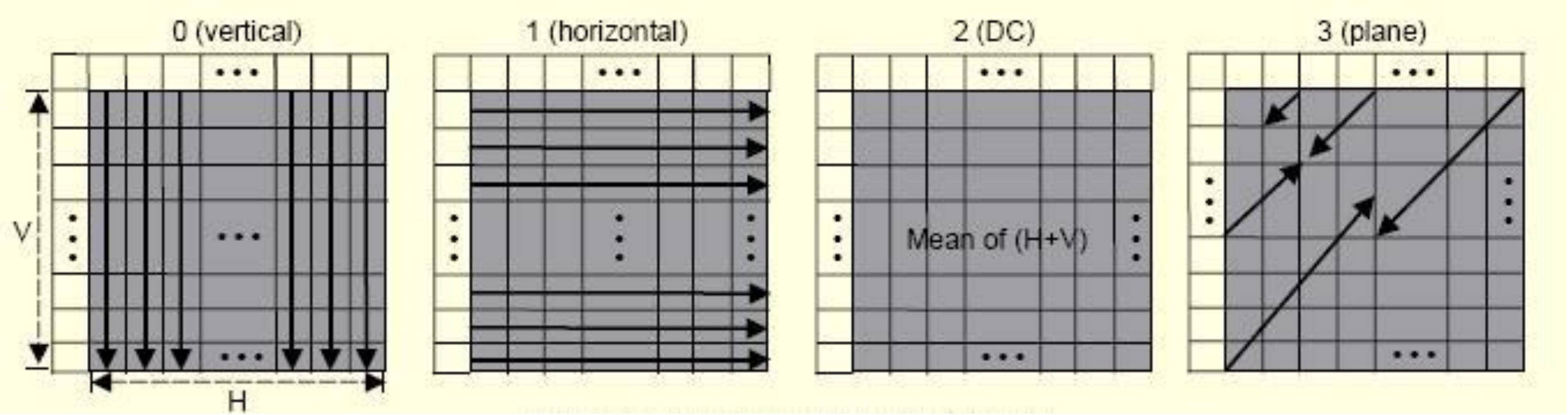
- Directional spatial prediction (9 types for luma, 1 chroma)

Q	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				



- e.g., Mode 3:
  diagonal down/right prediction
  a, f, k, p are predicted by
  (A + 2Q + I + 2) >> 2

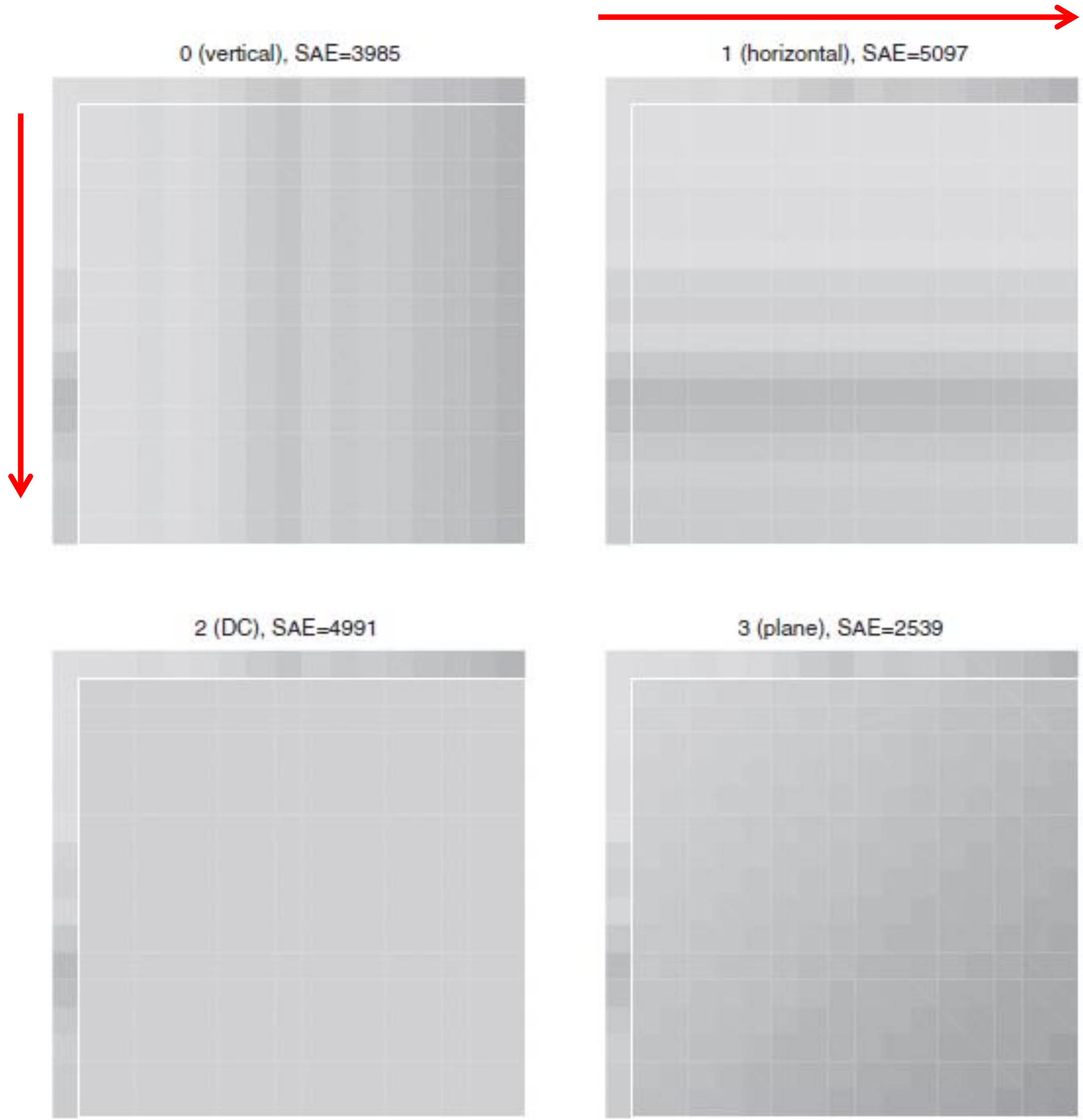
16×16 Blocks Intra Prediction Modes

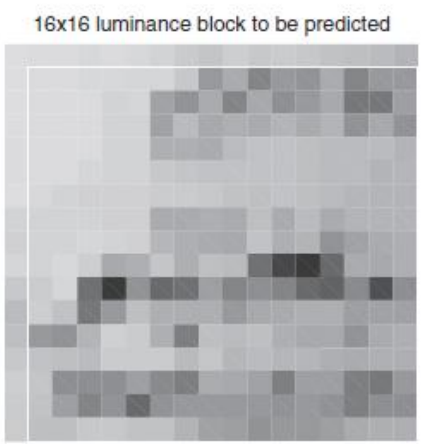


Average of all
neighbouring samples

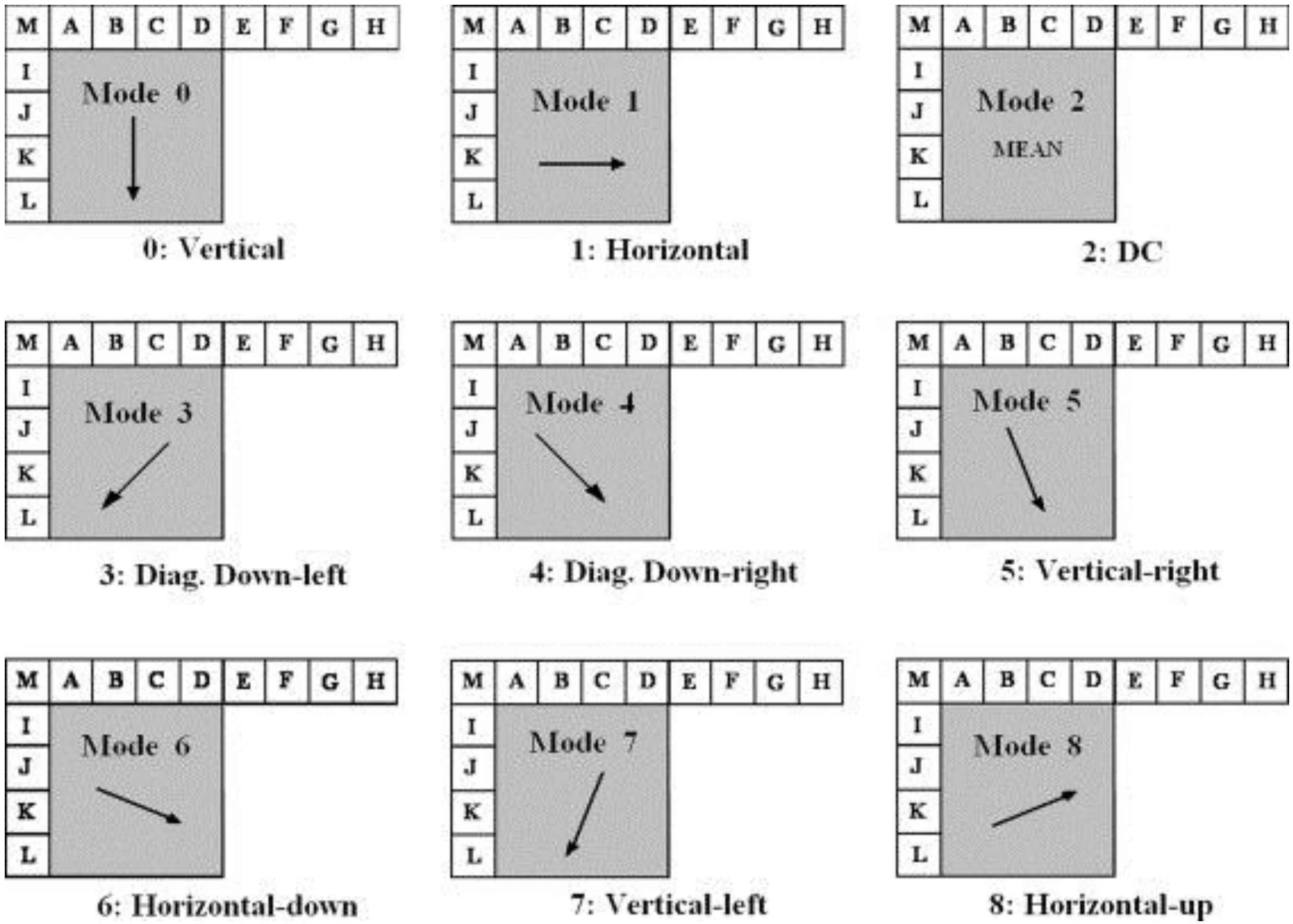
- **The luminance is predicted in the same way for all samples of a 16×16 MB (Intra16×16 modes).**
- **This coding mode is adequate for the image areas which have a smooth variation.**

16×16 Intra Prediction Modes at Work

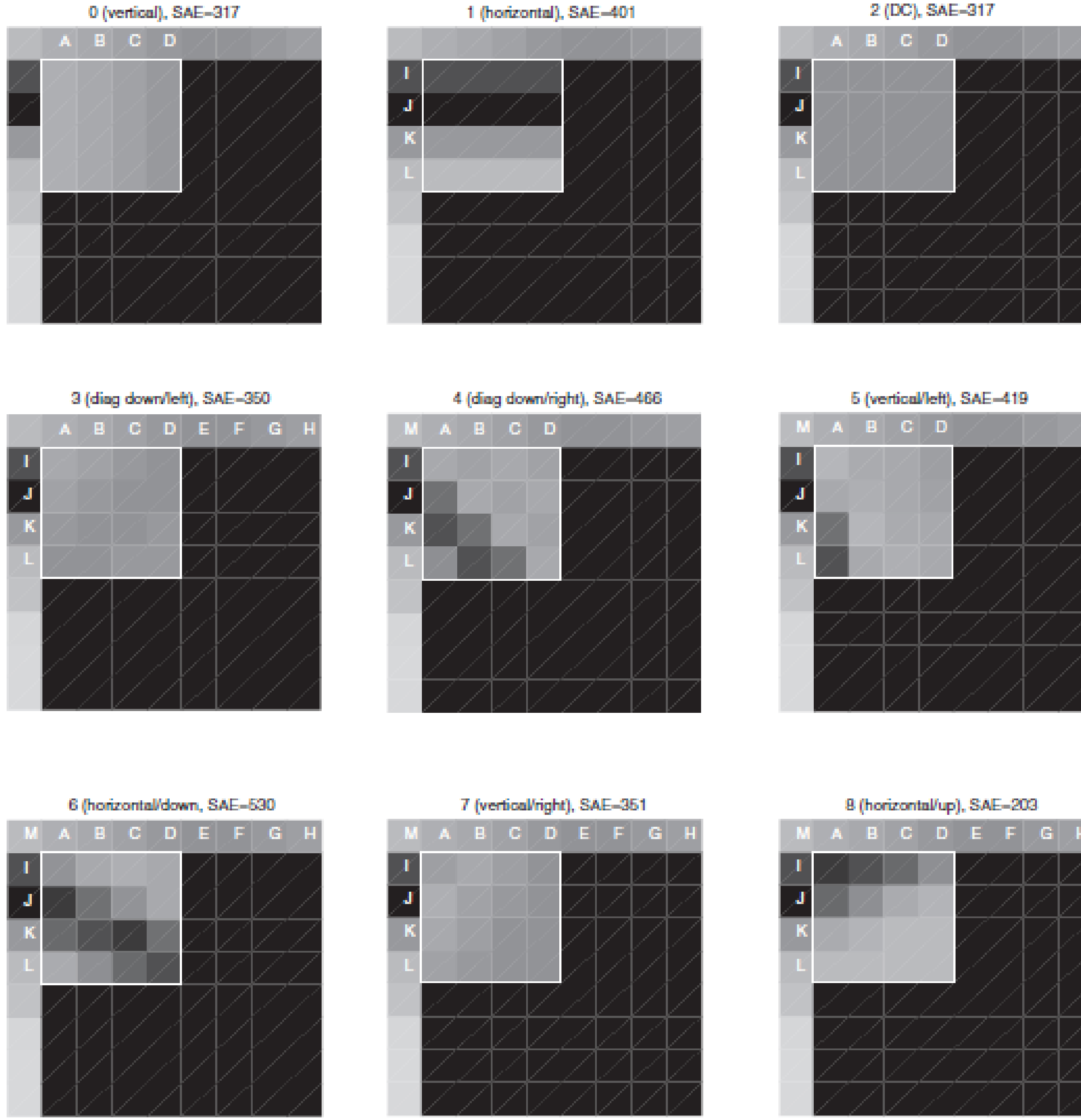




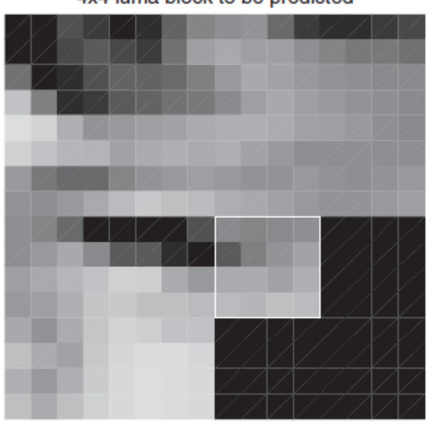
4×4 Intra Blocks Intra Prediction Modes



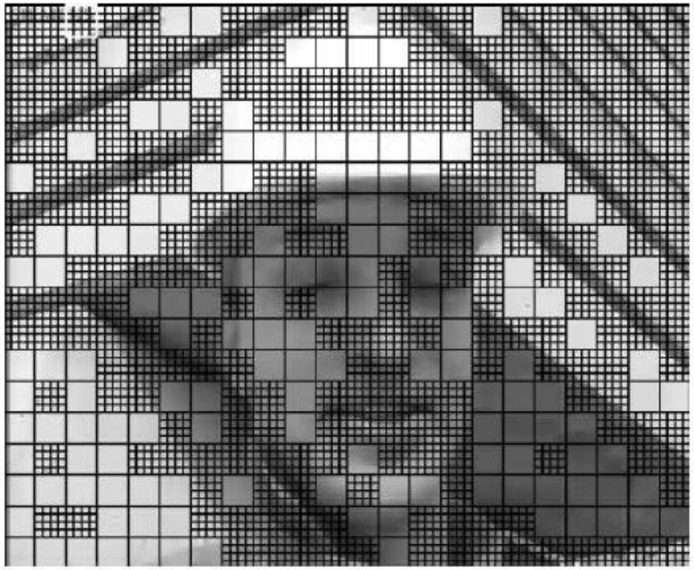
4×4 Intra Prediction Modes at Work



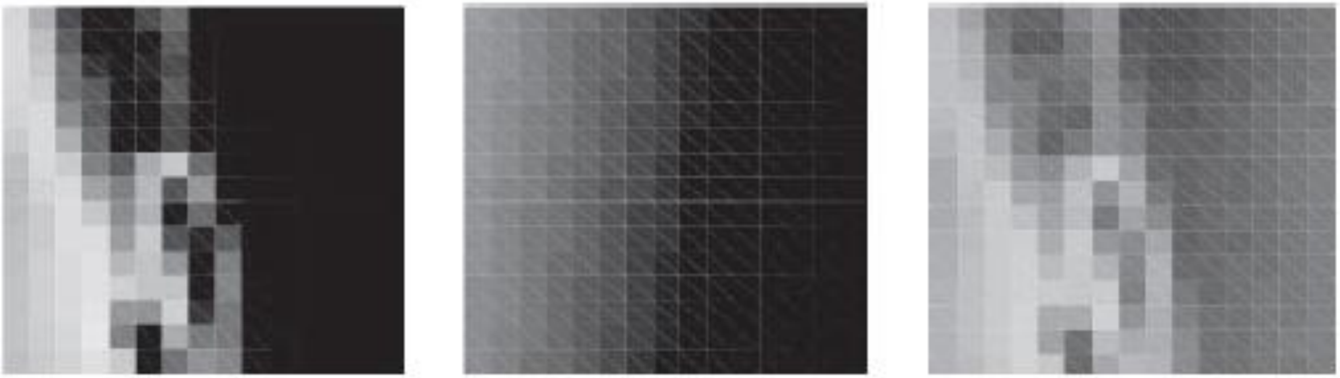
4x4 luma block to be predicted



Intra Prediction Examples



4×4 and 16×16 Intra prediction modes distribution



Original MB Intra Prediction Residual MB

Intra versus Inter Predictions ...

(a) Original frame.

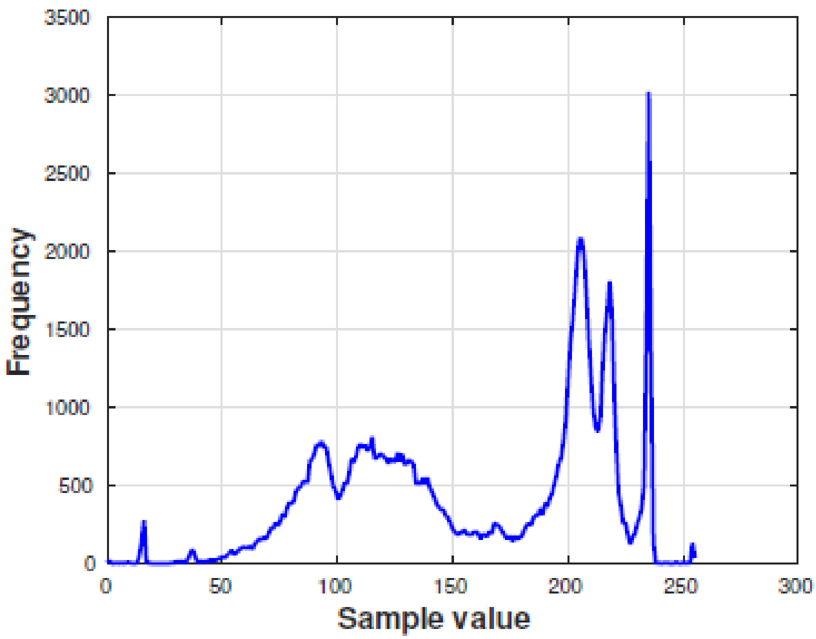
(b) Intra predicted frame.

(c) Frame difference between the original and the intra predicted frame.

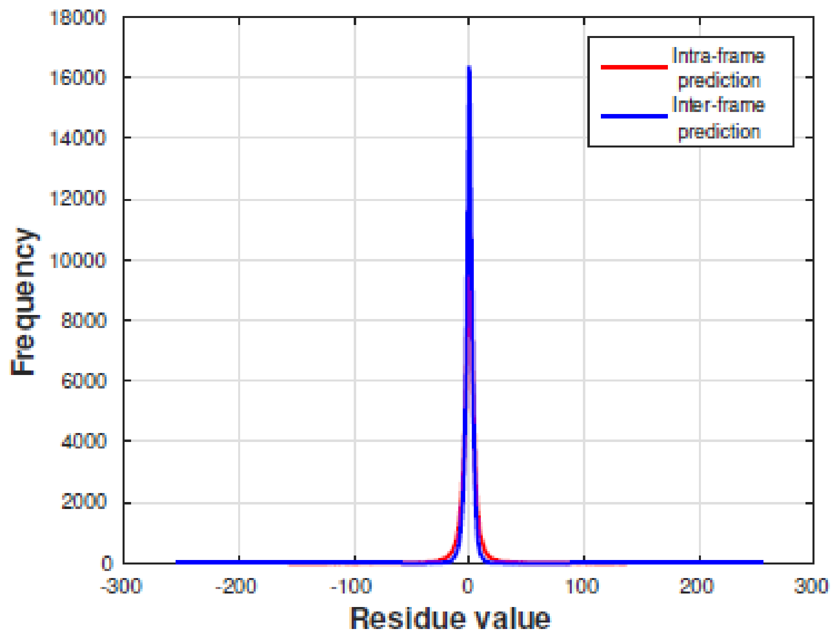
(d) Inter predicted frame.

(e) Frame difference between the original and the inter predicted frame.

Absolute versus Residual Energy ...



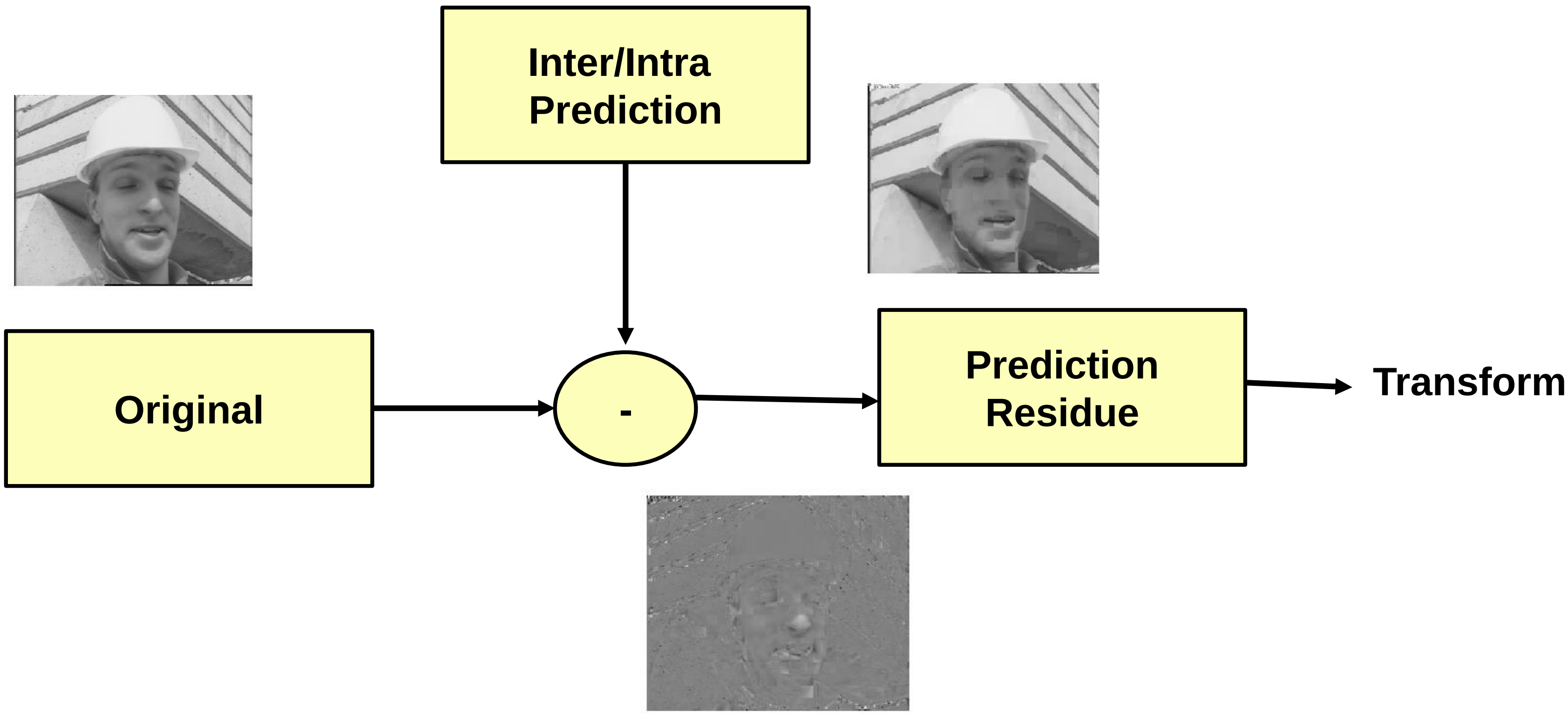
(a) Histogram of the original sample amplitude.



(b) Histogram of the residue amplitude.

Transform and Quantization

Creating the Prediction Residue ...



- **The better the prediction, the smaller the prediction residue**
- **In general, the smaller the prediction residue, the smaller the rate**

Multiple Transforms

Until now, only 8×8 DCT transforms !

The H.264/AVC standard uses 3 transforms to exploit the spatial redundancy in the Intra and Inter prediction residues, depending on the type of prediction residue to code:

1. 4×4 Integer Transform (iDCT) based on the DCT for all the other coefficients
2. 4×4 Hadamard Transform for the luminance DC coefficients in MBs coded with the Intra16×16 mode
3. 2×2 Hadamard Transform for the chrominance DC coefficients in any MB

Later, H.264/AVC 2nd version has reintroduced the 8×8 transform size, with a 8×8 iDCT transform.

The Old 1D-DCT …

$$c(k) = \alpha(k) \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi(2n+1)k}{2N}\right]$$ **DCT**

$$x(n) = \sum_{k=0}^{N-1} c(k)\alpha(k) \cos\left[\frac{\pi(2n+1)k}{2N}\right]$$ **Inverse DCT**

onde $\alpha(0) = \sqrt{\frac{1}{N}},\ \alpha(k) = \sqrt{\frac{2}{N}},\ 1 \le k \le N-1$

$$\{\mathbf{C}_N\}_{kn} = \alpha(k) \cos\left[\frac{\pi(n+\frac{1}{2})k}{N}\right]$$

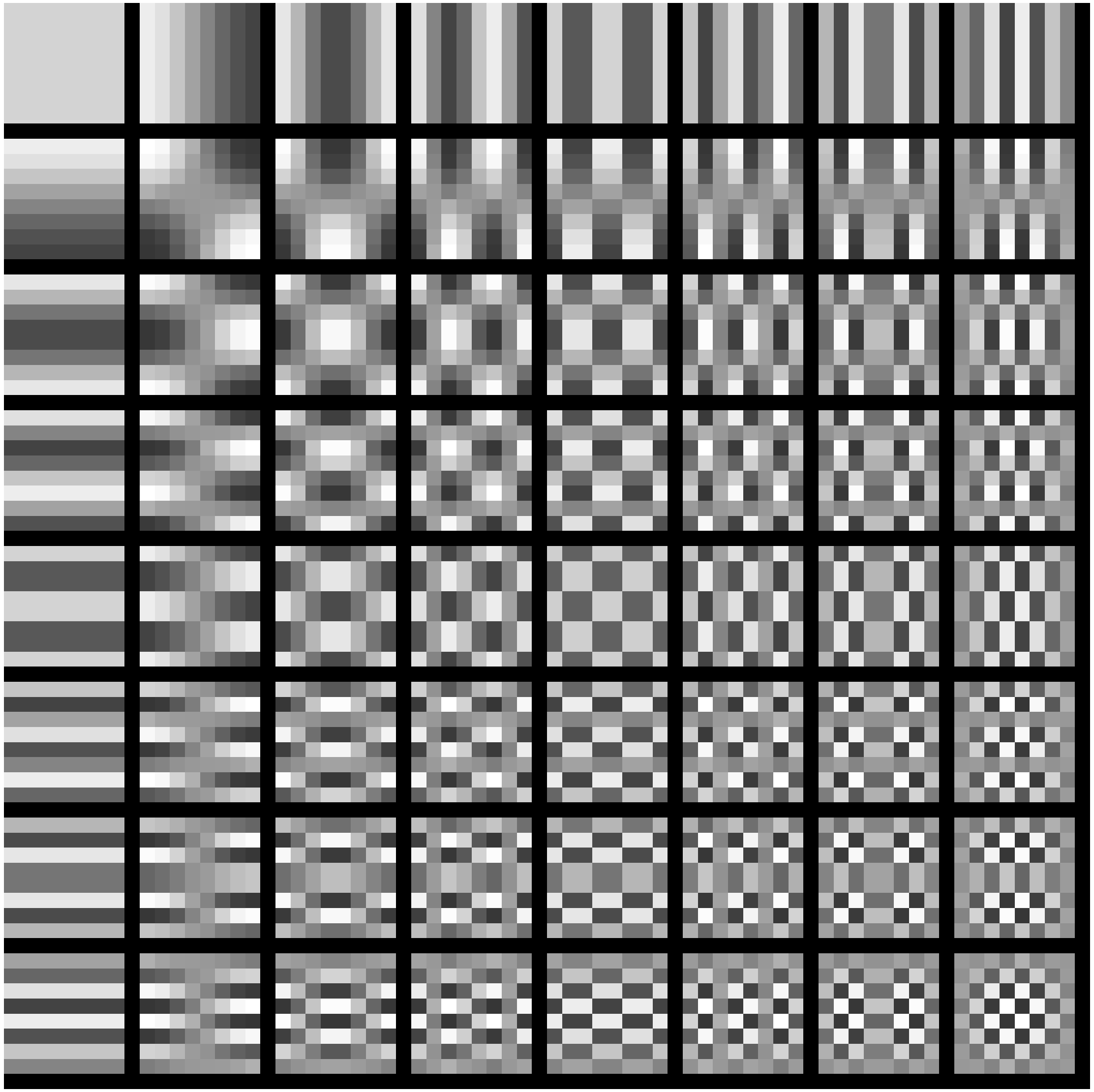
$$\mathbf{c} = \mathbf{C}_N \mathbf{x}$$
$$\mathbf{x} = \mathbf{C}_N^T \mathbf{c}$$

$$\mathbf{C}_N \mathbf{C}_N^T = \mathbf{I}$$

As the DCT basis functions use irrational numbers, the DCT coefficients may also be irrational numbers, thus creating the so-called *decoding mismatches* depending on the way rounding and truncation are implemented in each decoder.

No lossless coding is possible without the full specification of the transform process, which is undesirable.

2D DCT Basis Functions



Integer DCT Transform

The H.264/AVC standard uses an integer transform to code the prediction residue.

- **The transform is applied to 4×4 (later also 8×8) image blocks, $B_{4\times4}$, using a separable transform with properties similar to a 4×4 DCT**

$$C_{4x4} = T_v \cdot B_{4x4} \cdot T_h^T$$

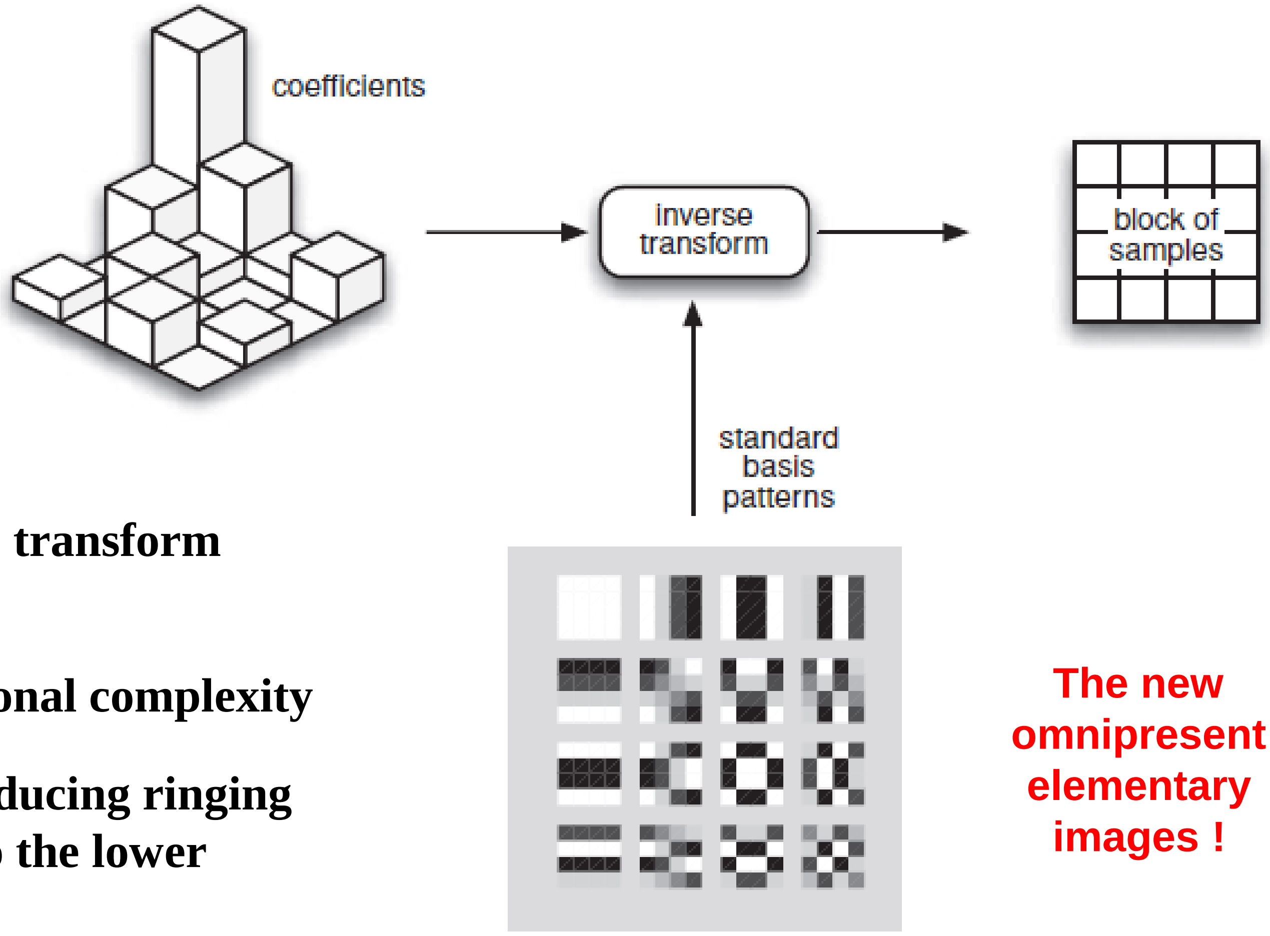
- **T_v, T_h: vertical and horizontal transform matrixes**

$$T_v = T_h = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

- **4×4 Integer DCT Transform**
  - Easier to implement (only sums and shifts, no multiplications)
  - No mismatch in the inverse transform

- **Later, a similar 8×8 integer DCT transform has been also adopted.**

The 4×4 DCT Basis Functions

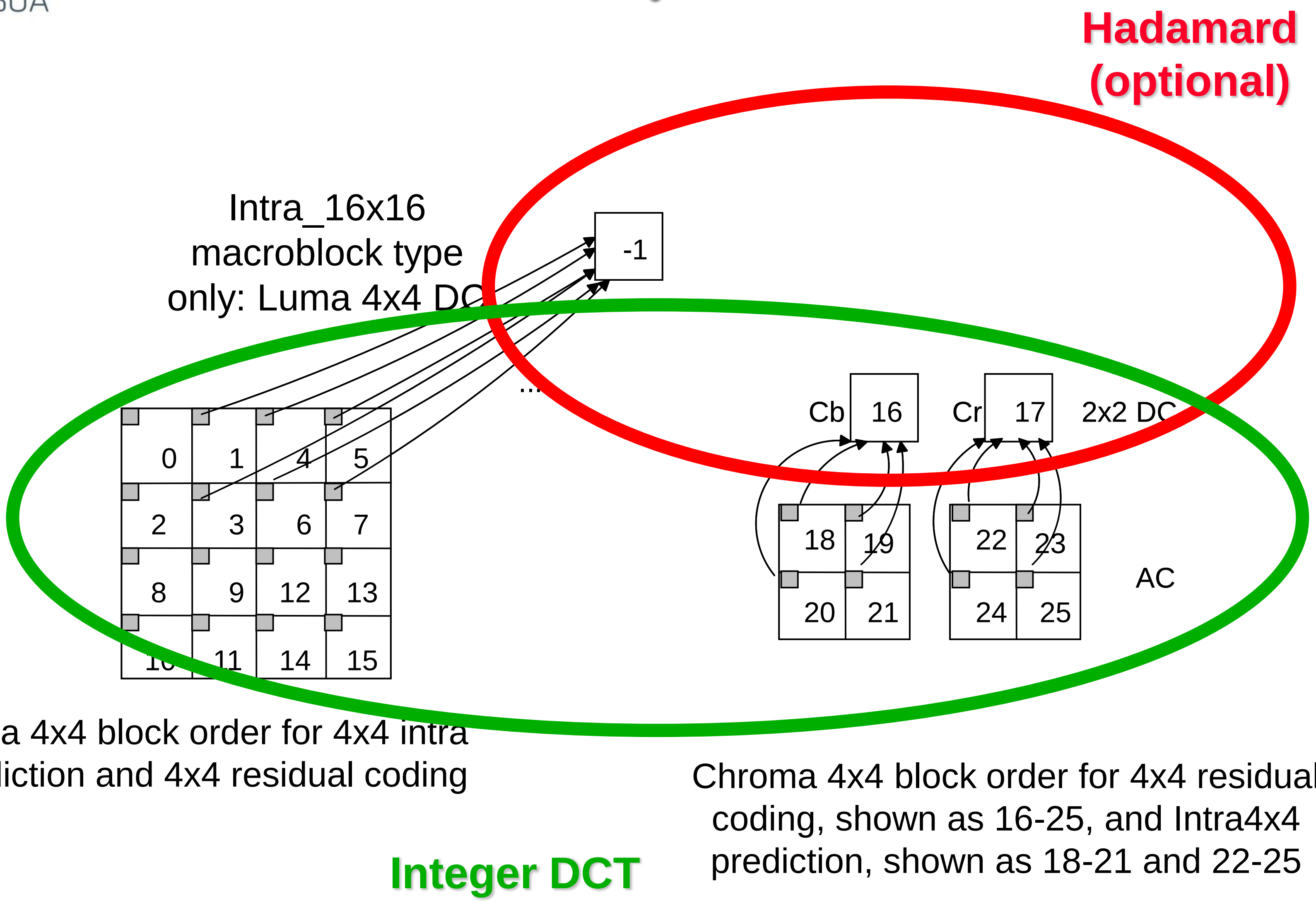


The 4×4 block-size transform allows

- **Low computational complexity**
- **Significantly reducing ringing artifacts (due to the lower spatial support)**

The new omnipresent elementary images !

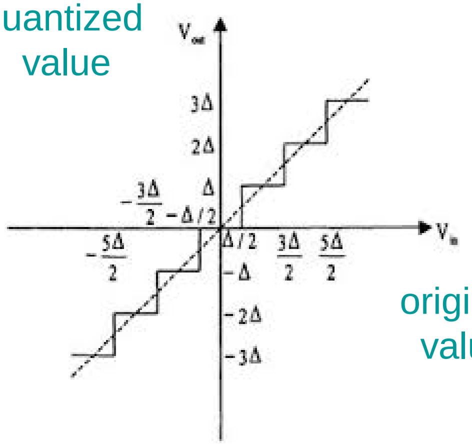
Hierarchical Two-Layer Transform



Luma 4x4 block order for 4x4 intra prediction and 4x4 residual coding

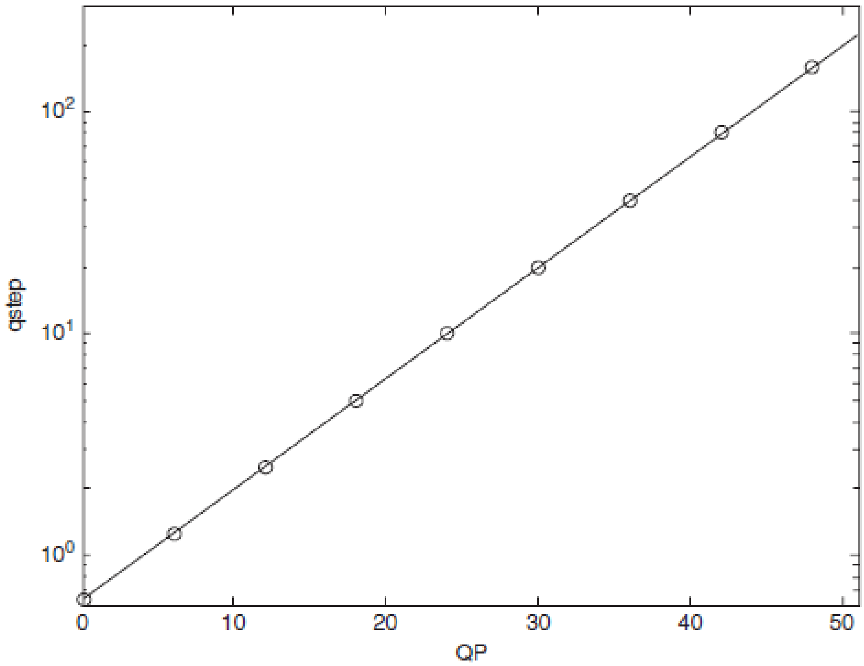
Chroma 4x4 block order for 4x4 residual coding, shown as 16-25, and Intra4x4 prediction, shown as 18-21 and 22-25

Quantization …



- **Quantization <u>removes irrelevant information</u> to obtain a rather substantial bitrate reduction.**
- **There is always a quantization error involved !**
- **Quantization also <u>allows to code at a certain target bitrate</u> by controlling the quantization error.**
- **Before quantization, some coefficients may be zeroed using a Just Noticeable Difference (JND) model which may have spatial and temporal components.**
- **<u>Quantization</u> basically corresponds to the division, performed at the encoder, of each transform coefficient by a <u>quantization factor</u>, thus attributing a <u>quantization level</u> to each coefficient.**
- **On the contrary, <u>inverse quantization</u> (reconstruction) corresponds to the multiplication of the received quantization level by the same quantization factor (performed both at the encoder and decoder) and eventually adding some offset.**

Q_{step} *versus* Q_p



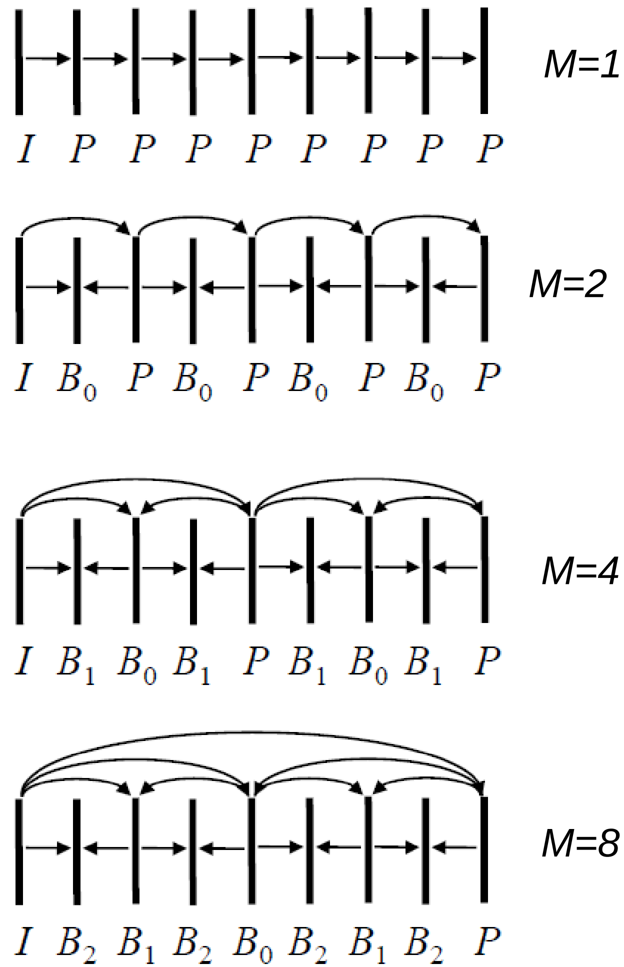
Quantization parameter, Q_p, **is an integer in the range 0-51.**

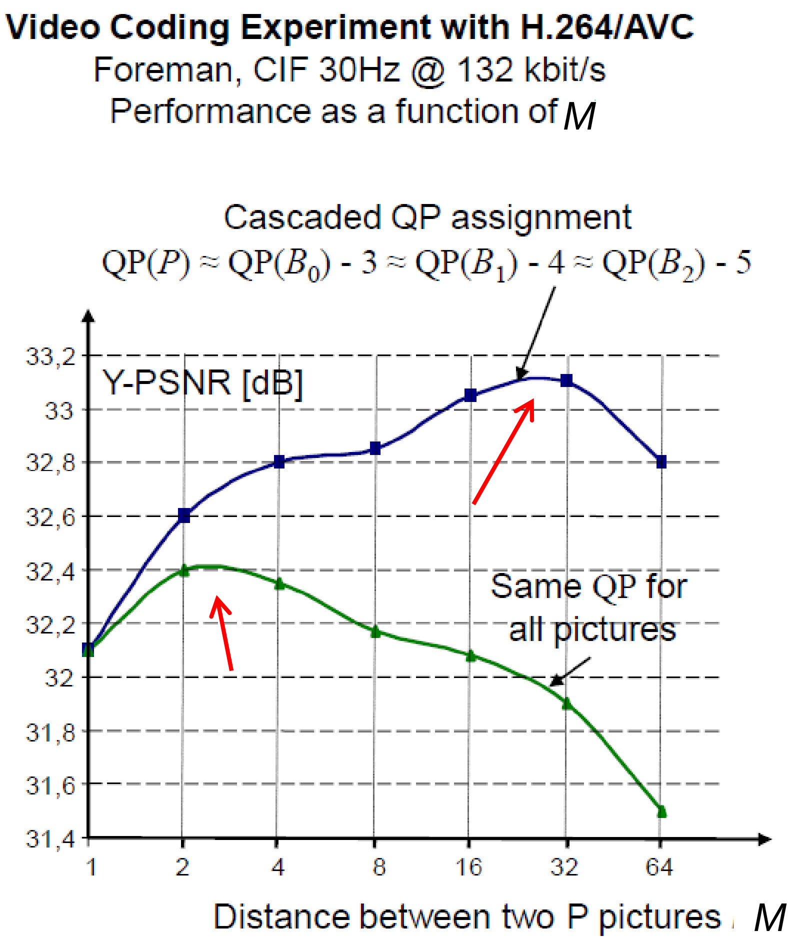
The higher the Q_p, the lower the rate/quality.

Q_p is just an index; Q_{step} is the real quantization step value !

- **An increase by 6 in Q_p doubles the quantization factor size (Q_{step}), such that the mapping of Q_p values to quantization step sizes (Q_{step}) is approximately logarithmic.**
- **The Q_p to Q_{step} mapping has been defined to have a reduction of about 12.5% in the bitrate for an increment of '1' in Q_p.**

Cascading Quantization Impacts

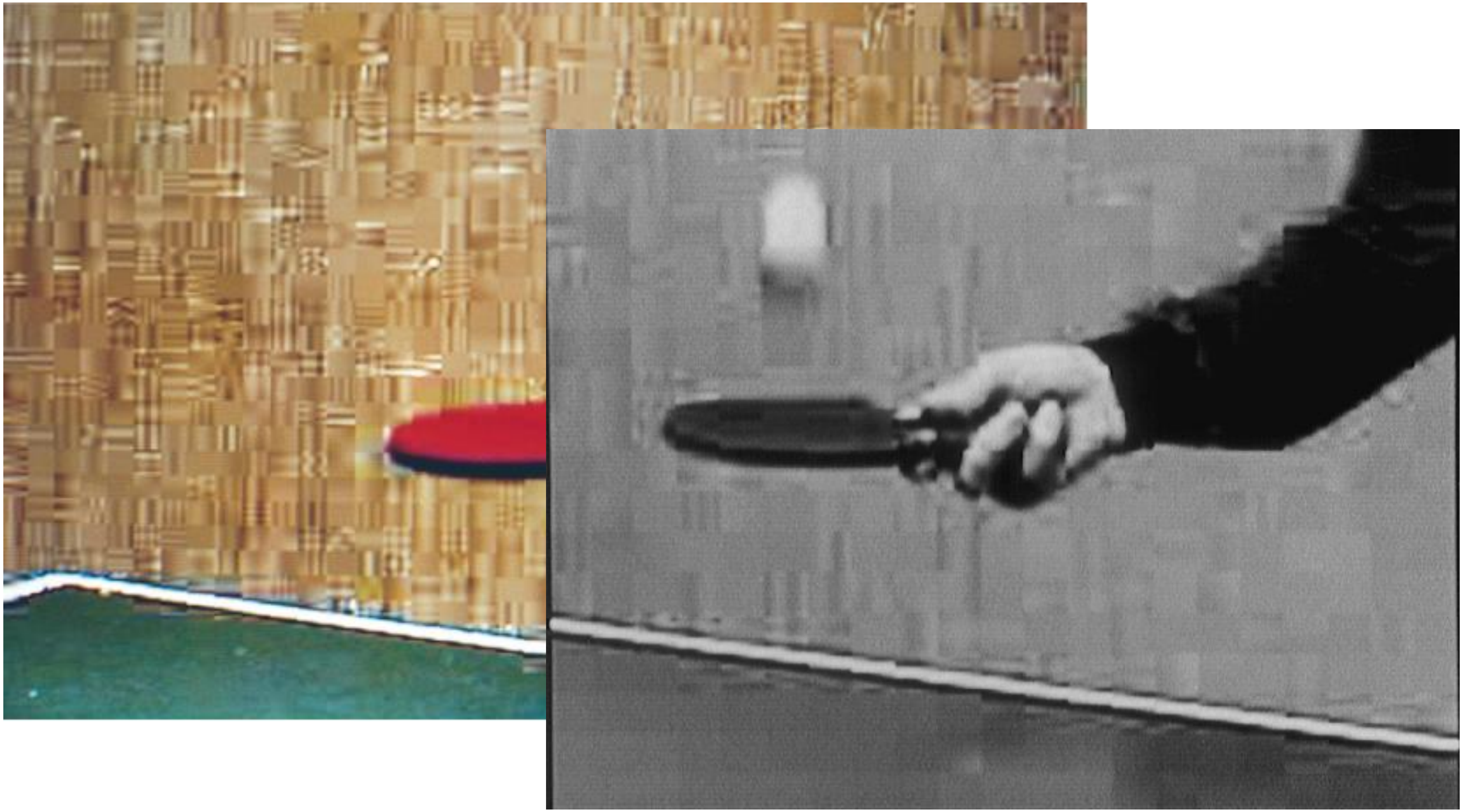




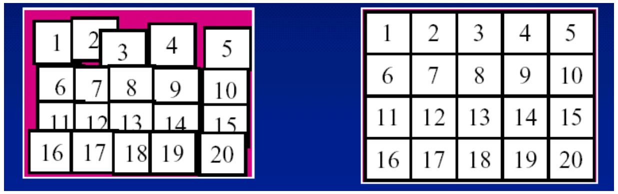
There is a hierarchy within the B frames !

Deblocking Filter

The Blocking Effect …



Blocking Effect: the Origin



There are two main tools in the H.264/AVC architecture that can be a source of blocking artifacts:

1. **IDCT with quantization** - The most significant factor is the block-based iDCT in Intra and Inter prediction error coding. Coarse quantization of the iDCT coefficients can cause visually disturbing discontinuities at the block boundaries.

2. **Motion compensated prediction** - Motion compensated blocks are generated by copying interpolated samples from displaced locations of possibly different reference frames. Since there is almost never a perfect fit for this data, discontinuities on the edges of the copied blocks typically arise. Additionally, in the copying process, existing edge discontinuities in the reference frames are carried into the interior of the block to be compensated.

A deblocking filter is many times an advantageous tool to maximize the subjective quality and the compression performance.

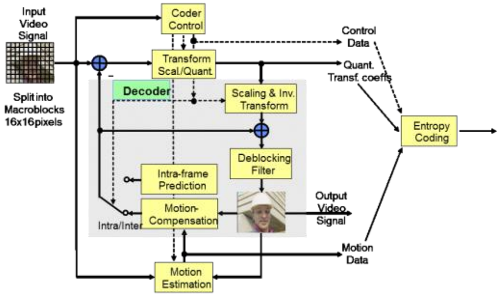
Deblocking Filter Approaches

H.261 already included a simple in-loop filter ! But not MPEG-1 and MPEG-2 Video …

There are two main approaches to integrate deblocking filters into video codecs:

- **POST-LOOP FILTERS -** Only operate on the display buffer outside the coding loop; thus, they are not normative in the standardization process. Because their use is optional, post-loop filters offer maximum freedom for decoder implementations**.**

- **IN-LOOP FILTERS -** Operate within the coding loop where the filtered frames are used as reference frames for motion compensation of subsequent coded frames. This forces all standard compliant decoders to perform identical filtering to stay in synchronization with the encoder. Naturally, a decoder can still perform post-loop filtering in addition to in-loop filtering, if found necessary in a specific application.
  - Guarantee a certain level of quality
  - No need for extra frame buffer in the decoder
  - Improve objective and subjective quality with reduced decoding complexity



H.264/AVC Deblocking: Adaptive, In-Loop Approach

The H.264/AVC standard specifies the use of an adaptive, in-loop deblocking filter operating at the block edges with the target to increase the final subjective and objective qualities.

- **The filter performs simple operations to detect and analyze artifacts on coded block boundaries and attenuates those by applying a selected filter.**
- **The same filter needs to be present at the encoder and decoder (thus normative) since the filtered blocks are after used for motion estimation (in-loop filter). This filter has a performance superior to a post-loop filter (not in-loop and thus not normative).**
- **The H.264/AVC filter has the following main advantages:**
  - **Improved subjective quality** - Blocks edges are smoothed without making the image blurred, improving the subjective quality.
  - **Improved compression efficiency** - The filtered blocks are used for motion compensation resulting in smaller residues after prediction, this means reducing the bitrate for the same target quality.

H.264/AVC Deblocking: Basics

- **Target -** In deblocking filtering, it is essential to be able to distinguish between *true* edges in the image and those created by quantization and motion compensation. To preserve image sharpness, the *true* edges should be left unfiltered as much as possible while filtering the *artificial* edges to reduce their visibility.

- **Basic idea -** A big difference between samples at the edges of two blocks should only be filtered if it can be attributed to quantization or motion compensation; otherwise, that difference must come from the image itself and, thus, **should not be filtered**.

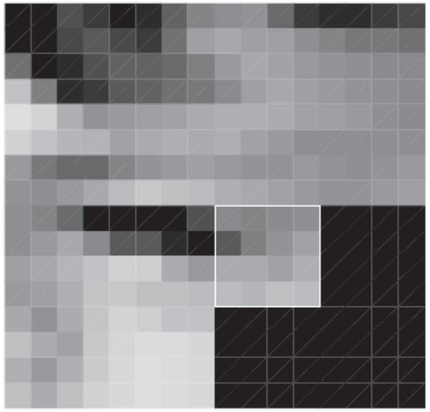
- **Method -** The filter is applied to the vertical and horizontal edges of all 4×4 blocks in a MB; it is adaptive to the content, essentially removing the block effect without necessarily smoothing the image:
  - **At slice level,** the filter strength may be adjusted to the characteristics of the video sequence.
  - **At the edge block level**, the filter strength is adjusted depending on the type of coding (Intra or Inter), the motion and the coded residues.
  - **At the sample level**, sample values and quantizer-dependent thresholds can turn off filtering for each individual sample.

H.264/AVC Deblocking: Adaptability Control

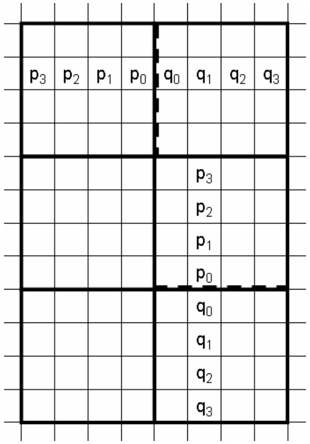
- The adaptive filter is controlled through a *Boundary-Strength* (*Bs*) parameter which is defined, at the decoder, for every edge between two 4×4 luminance sample blocks to define the filter strength.

- The value depends on the modes and coding conditions of the two adjacent (horizontal or vertical) blocks.

Block modes and conditions	Bs
One of the blocks is Intra *and* the edge is a macroblock edge	4
One of the blocks is Intra	3
One of the blocks has coded residuals	2
Difference of block motion ≥ 1 luma sample distance	1
Motion compensation from different reference frames	1
Else	0



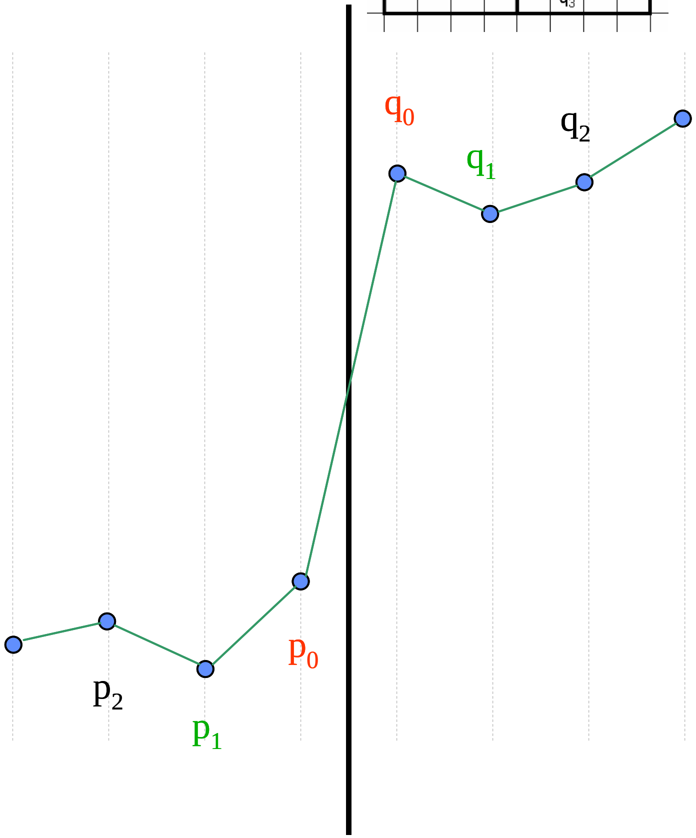
H.264/AVC Deblocking: Sample Level



- Up to three samples for luminance and one for chrominance on each side of the edge may be modified by the filtering process.

- Filtering on a line of samples only takes place if these three conditions all hold

$$|p_0 - q_0| < \alpha(\text{Index}_A)$$
$$|p_1 - p_0| < \beta(\text{Index}_B)$$
$$|q_1 - q_0| < \beta(\text{Index}_B)$$



- In these conditions, both table-derived thresholds are dependent on the average quantization parameter (QP) employed over the edge, as well as encoder selected offset values that can be used to control the properties of the deblocking filter at the slice level.

- The dependency of *α* and *β* on QP links the strength of the filtering to the general quality of the reconstructed picture prior to filtering.

H.264/AVC Deblocking: Subjective Result for Intra Coding at 0.28 bit/sample

1) Without filter

2) With H.264/AVC deblocking

H.264/AVC Deblocking: Subjective Result for Strong Inter Coding

1) Without Filter

2) With H.264/AVC deblocking

Exploiting Statistical Redundancy

Entropy Coding: Past Defficiencies

- **Entropy coding such as in MPEG-2 Video, H.263, and MPEG-4 Visual is based on fixed VLC tables.**
- **Due to VLCs, coding events with probability > 0.5 (this means $\log_2(1/p) < 1$ bit/symbol) cannot be efficiently represented (at least without code extensions which are not appropriate here).**
- **The usage of fixed VLC tables does not allow any efficient adaptation to the actual symbol statistics.**
- **If there is a fixed assignment of VLC tables and syntax elements, existing inter-symbol redundancies – *the context* - cannot be exploited.**

Huffman (VLC) Coding

Symbol A:
$\log_2 (1/0.7) = 0.5145 < 1$ bit/symbol

Símbolo	Probabilidade redução 0	Palav. Código redução 0	Probabilidade redução 1	Palav. Código redução 1
A	0.7	0	0.7	0
B	0.2	1 0	0.3	1
C	0.1	1 1		

Huffman coding allows obtaining a code with an average number of bits per symbol as close as desired to the source entropy.

But this requires knowledge on the source statistics, i.e., symbol probabilities.

Entropy = 1.157 bit/symbol
($H = \Sigma\, p_i \log_2 (1/p_i)$ bit/symbol)

Average code length = 1.3 bit/symbol

Efficiency = 1.157/1.3 = 89%

Entropy Coding

1 0 1 0 0 1 1 0 0 1 0 …

There are two types of H.264/AVC entropy coding:

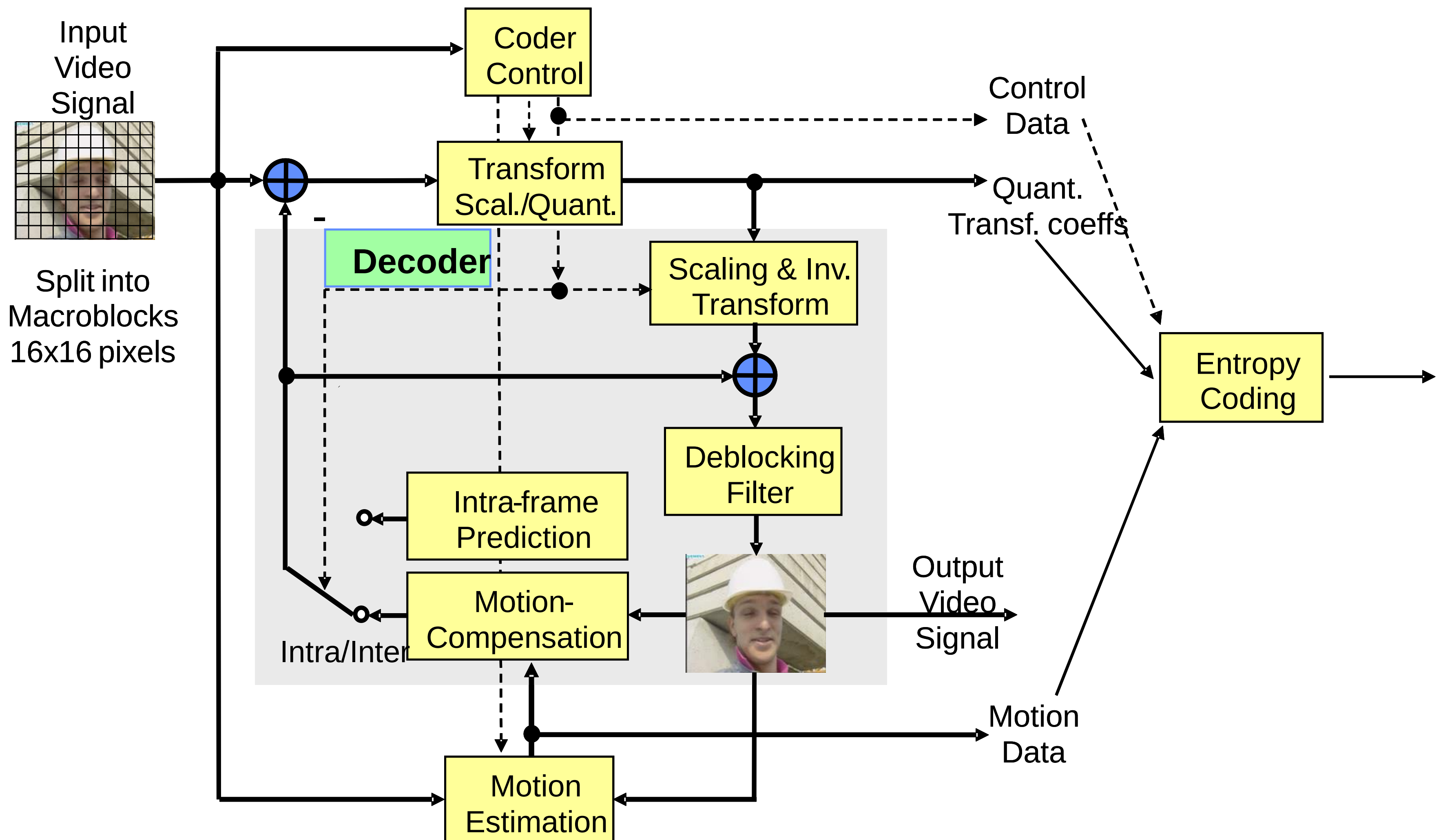
- Context Adaptive Variable Lenght Coding (CAVLC)
- Context-Based Adaptive Binary Arithmetic Coding (CABAC)

- **In the higher syntactic hierarchical levels, e.g. sequence, picture, the syntactic elements are coded using fixed or variable length codes.**
- **For the lower syntactic levels, e.g. slice, MB, and block, the syntactic elements are coded using CAVLC or CABAC.**
- **CABAC provides better compression, notably 5-15% less bitrate than CAVLC, at the cost of some additional complexity.**

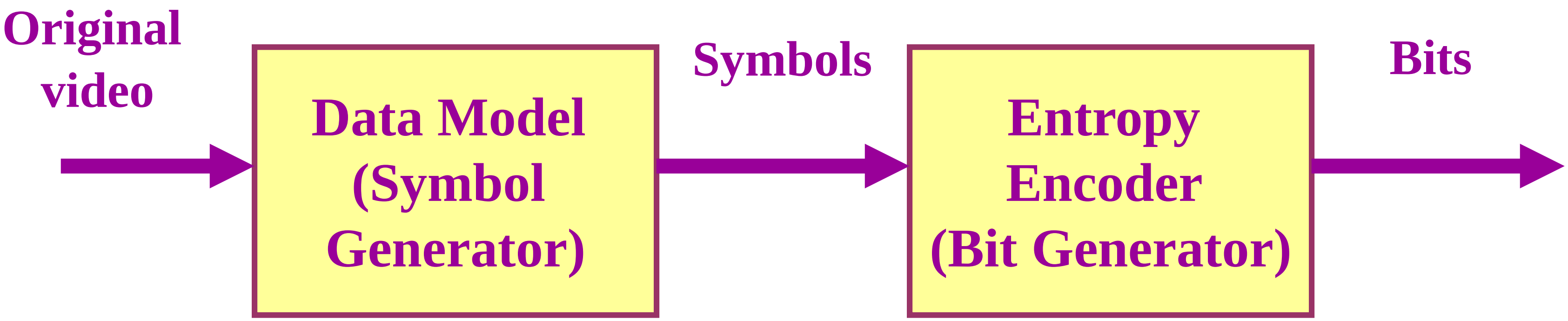
TÉCNICO LISBOA

Overall Codec

H.264/AVC Basic Encoding Architecture: Again Predictive Coding



The H.264/AVC Video Symbolic Model



A video sequence is represented as a succession of pictures (interlaced or progressive) constituted by I, P and B slices, structured in macroblocks, each of them represented using motion vectors regarding multiple reference frames and/or quantized transform coefficients, following the constraints imposed by the slice coding type.

Trading-off Compression Efficiency and Error Resilience

Error Resilience Tools Summary

Typically, source coding-level error resilience tools reduce the compression efficiency to 'buy' error robustness. Ideally, compression and error robustness should be separately treated but sometimes practice is different …

- **Slice Structured Coding**
- **Flexible Macroblock Ordering (FMO)**
- **Arbitrary Slice Ordering (ASO)**
- **Slice Data Partitioning**
- **Intra Coding**
- **Redundant Slices**
- **Flexible Reference Frame Concept**
- **Switching Pictures**

This is NOT about error correction but rather about creating source coding streams less sensitive (thus more robust) to errors !

Working for Error Robustness

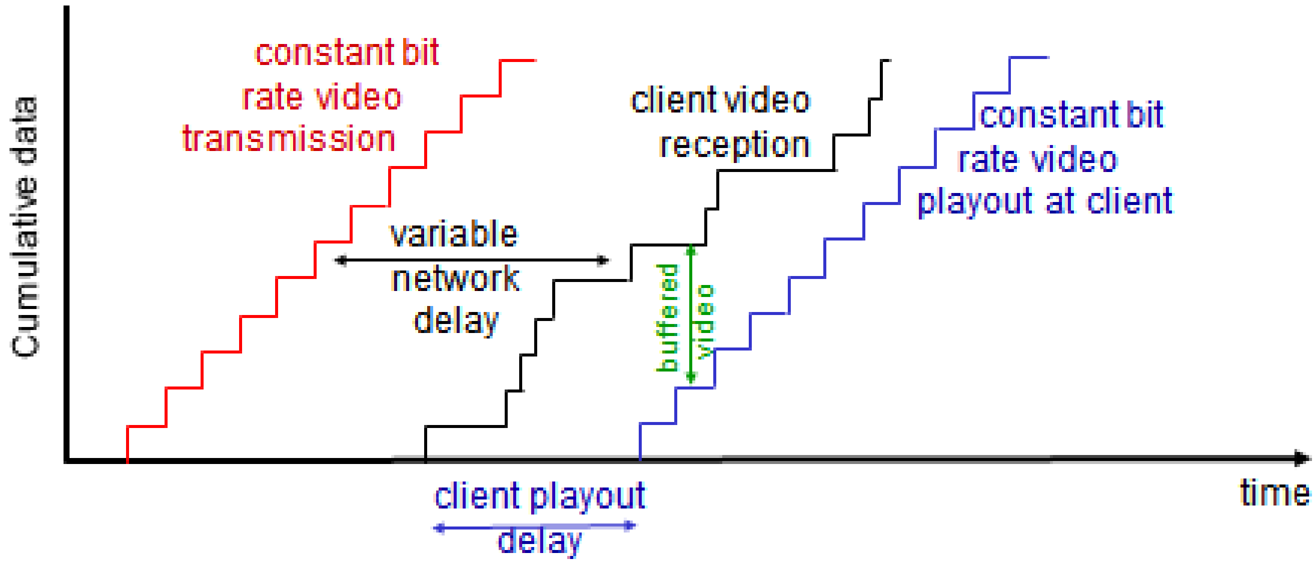
- **Since errors cannot be completely avoided, the following system design principles are relevant:**
  - **Error correction before the (de)coding layer** - Minimize the amount of losses in the channel without completely sacrificing the video bitrate (e.g. FEC and retransmission).
  - **Error detection** - If errors are unavoidable, detect and localize erroneous video data as efficiently as possible.
  - **Prioritization methods** - If losses are unavoidable, at least minimize the loss rates for very important data (e.g. data partitioning).
  - **Error recovery and concealment** - In case of losses, minimize the visual impact of losses on the actual distorted image (e.g. slice resynchronization, SEI messages by encoder).
  - **Encoder-decoder mismatch avoidance** - Limit or completely avoid encoder and decoder mismatches resulting in annoying error propagation (e.g. IDR refresh, flexible references).

Bitrate Adaptivity

Bit rate adaptivity is important to react to the rate dynamics due to statistical traffic, variable receiving conditions, as well as handovers and random user activity.

- **Dynamic QP changing -** For online encoding, if the encoder has sufficient feedback on the expected bitrate, rate control can be applied mainly by *changing QPs dynamically*, but also by changing the temporal resolution.
- **Playout buffering -** When channel bitrate fluctuations are not *a priori* known at the transmitter, or there is no sufficient means or necessity to change the bitrate frequently, *playout buffering* at the receiver can compensate for bitrate fluctuations to some extent at the cost of adding some delay. The decoder buffers some amount of data before start playing, thus buying QoE with initial delay.
- **Dropping less important data -** If not enough, rate adaptation has to be performed, notably by *dropping less important data* using temporal scalability, data partitioning (coarse form of SNR scalability), Flexible Macroblock Ordering (FMO), …
- **Stream switching -** Longer term adaptation may require stream switching using IDR frames or SI/SP frames.

Client-Side Buffering and Playout Delay



Client-Side Buffering and Playout Delay compensate for (varying) network-added delay, and delay jitter.

Video 'stalls', this means 'freezings', are very negative from the subjective point of view.

Once again, delay and memory 'buy' a better quality of experience …

Controling the Encoder

Controlling Coding Performance

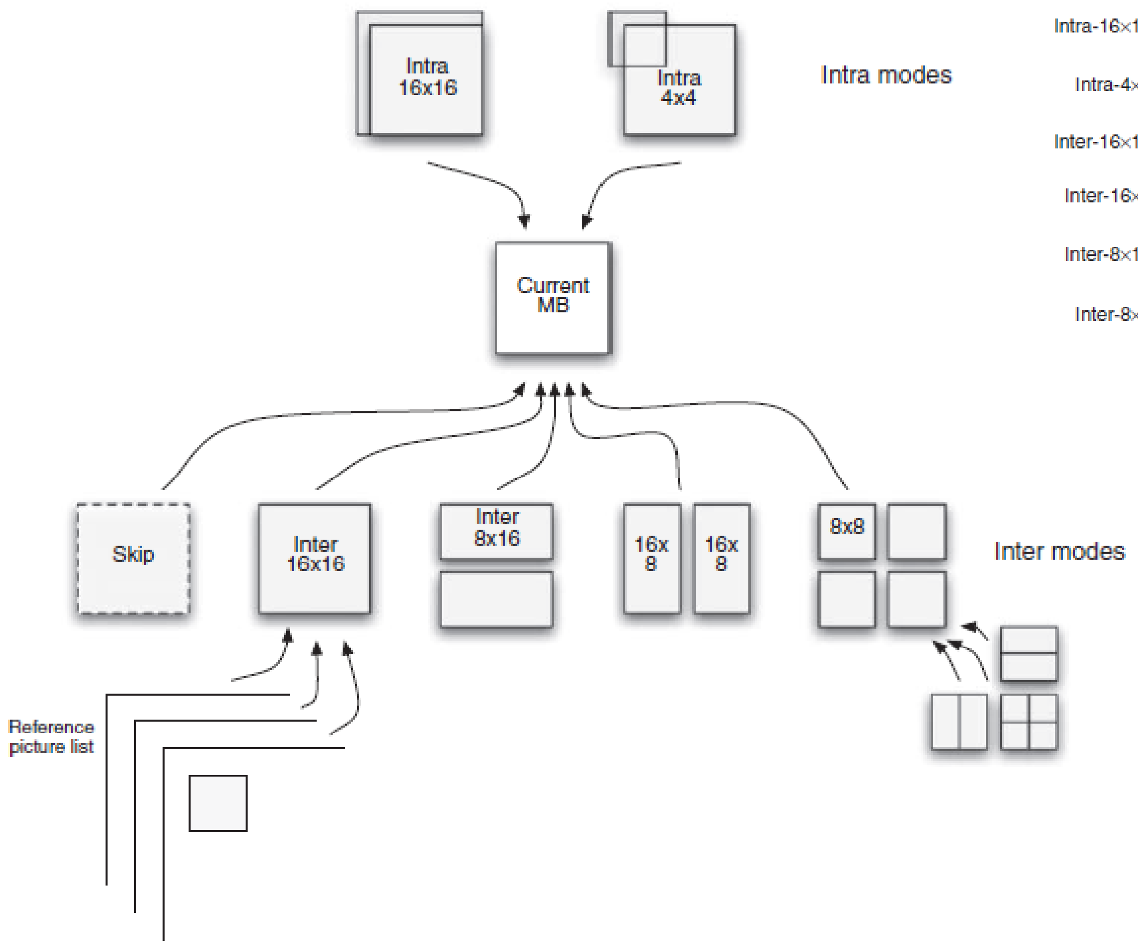
When using a standard with a specified decoder, parameters in the non-specified encoder should be selected such that good RD performance is achieved:

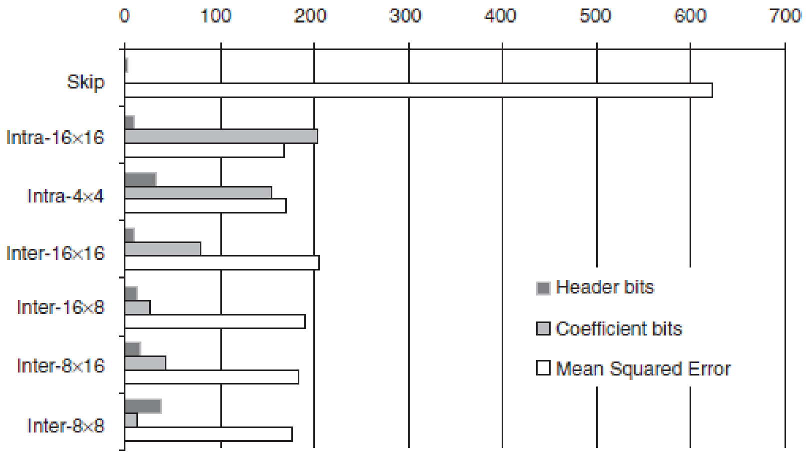
- **Global parameter selection -** Selects the appropriate temporal and spatial resolution of the video based on application, profile, and level constraints. Also, packetization modes, like slice sizes, are usually fixed for the entire session. The parameters are mainly determined by general application constraints.

- **Encoder control -** Performs local decisions, such as the selection of macroblock coding modes, reference frames, and motion vectors, on the macroblock level and below most appropriately based on a rate-distortion optimized (RDO) mode selection applying Lagrangian techniques.

- **Rate control -** Controls the timing and bitrate constraints of the application by adjusting the Quantization Parameter or Lagrange parameter and is usually applied to achieve a constant bit rate (CBR) encoded video suitable for transmission over CBR channels.

Available MB Prediction Modes: What is the Best Trade-off ?





If the encoder cannot select well the coding modes and parameters, the RD compression gains will not show up!

Encoder Control

- **Encoder control is a non-normative part of H.264/AVC; however, it is essential to determine the final RD compression performance.**
  - *The best RD performance <u>is NOT necessarily</u> associated to always choosing <u>the mode with the minimum error</u> but rather the mode with the best rate-distortion trade-off as there may exist modes which have almost the minimum error but a much lower rate.*

- **Its target is to choose the coding modes and parameters at encoder side, thus determining what part of the video signal should be coded using what method and parameter settings.**

- **This encoder freedom implies there is no guaranteed performance and thus there may exist 'good' and 'bad' encoders, always producing compliant streams.**

Encoder Control: Lagrangian Formulation

- **In mathematical optimization, the method of Lagrangian multipliers is a strategy for finding the local maxima and minima of a function subject to equality constraints.**
- **Constrained formulation:**

$$\min_{\mathrm{p}} D(\mathrm{p}) \quad \text{s.t.} \quad R(\mathrm{p}) \leq R_T$$

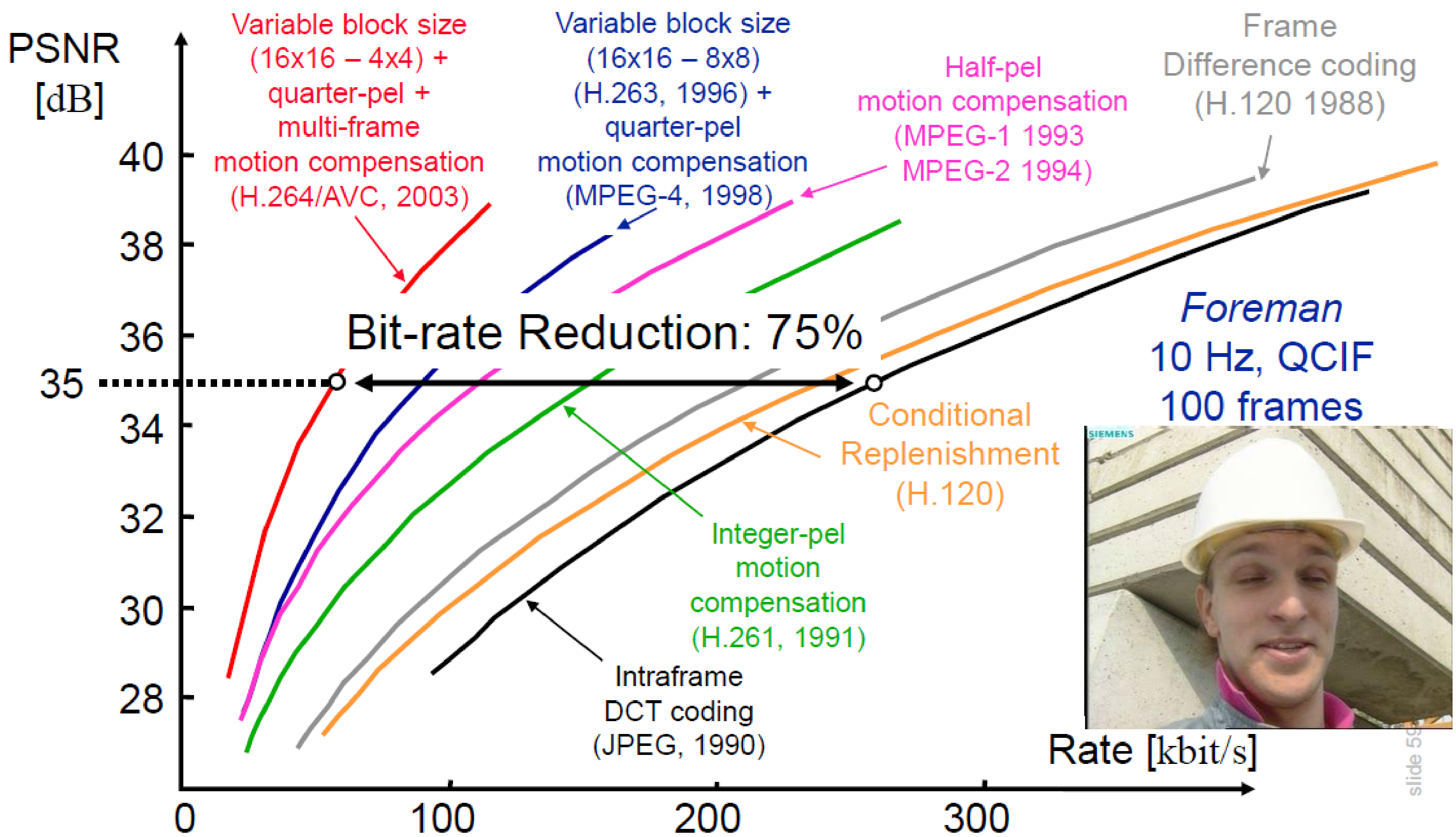
where D is Distortion, R is used Rate, R_T is Target rate and p the Parameters vector, e.g. coding mode, motion vectors, quantization step, etc.

- **Unconstrained Lagrangian formulation (the constraint goes in the cost):**

$$\mathrm{p}_{opt} = \arg\min_{\mathrm{p}} \left\{ D(\mathrm{p}) + \lambda \cdot R(\mathrm{p}) \right\}$$

with λ controlling the RD trade-off, i.e. small λ corresponding to high rate/quality and vice-versa.

Motion Estimation is King …



Complexity

Real-Time Coding: Fast Decisions are Essential !

For real-time applications, it is essential to adopt fast algorithms to select

- **The coding mode, e.g. Intra, Inter**
- **For Intra coding, the specific Intra prediction mode**
- **For Inter coding, the specific motion partition**
- **The (multiple) reference frame**
- **…**

Adding Complexity to Buy Quality

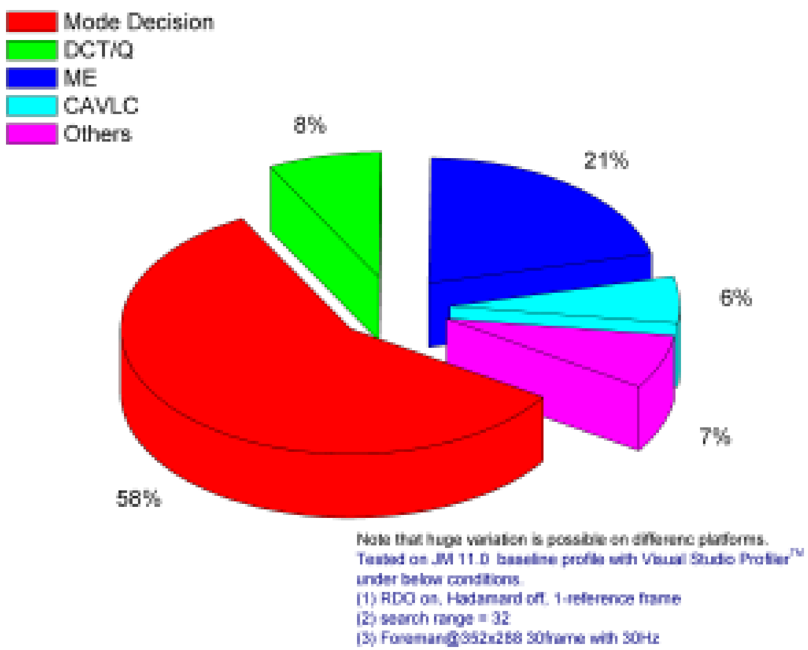
Complexity (memory and computation) typically increases 4× at the encoder and 3× at the decoder regarding MPEG-2 Video, Main profile.

It is essential to remind that the encoder complexity may be tens or hundreds of times higher than the decoder complexity !

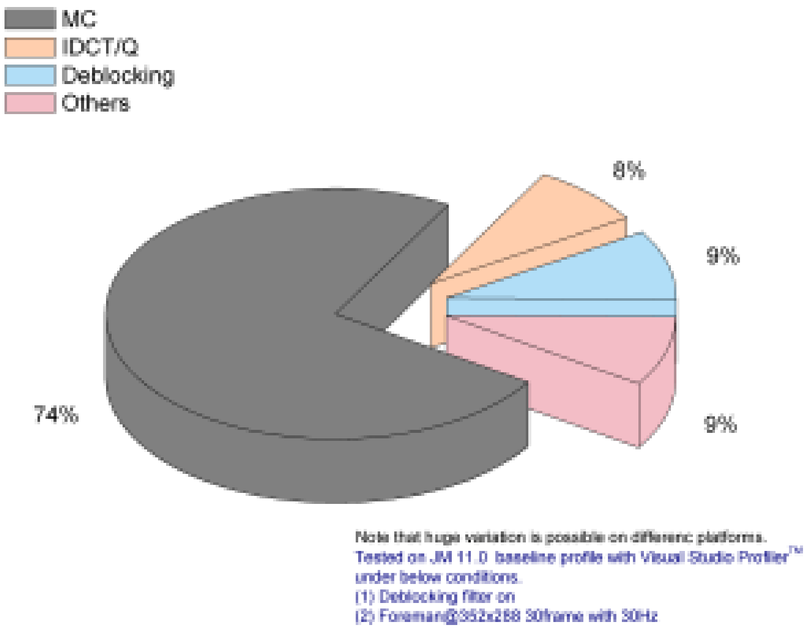
Problematic aspects:

- **Motion compensation with smaller block sizes (memory access)**
- **More complex (longer) filters for the ¼ pel motion compensation (memory access)**
- **Multiframe motion compensation (memory and computation)**
- **Many MB partitioning modes available (encoder computation)**
- **Intra prediction modes (encoder computation)**
- **More complex entropy coding (computation)**

Example Encoder and Decoder Complexity …



(a) H.264/AVC encoder computational costs; mode decision and ME occupy 79%

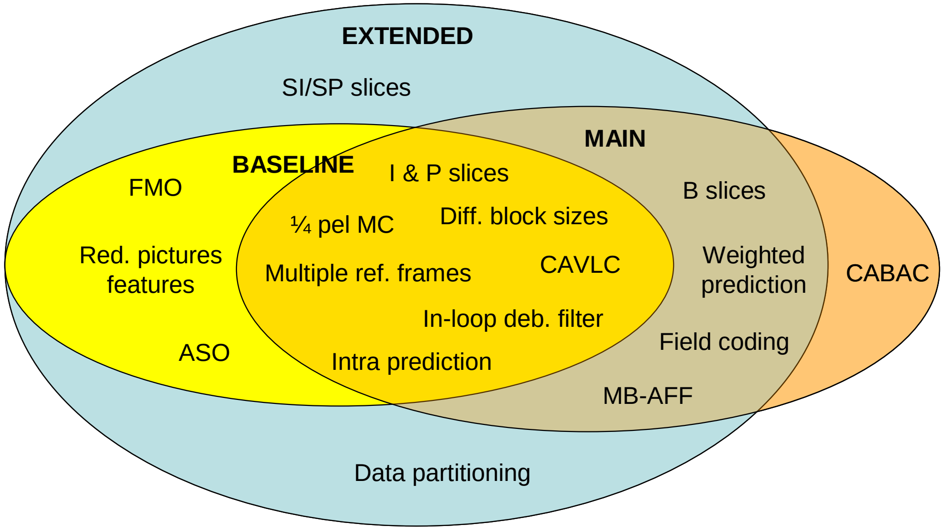


(b) H.264/AVC decoder computational costs; MC is the most computationally demanding block

ME – motion estimation
MC – motion compensation
Q - quantization

Profiles and Levels

First H.264/MPEG-4 AVC Profiles …



- **Baseline Profile is targeted towards real-time encoding and decoding for CE devices. Supports progressive video, uses I and P slices, CAVLC entropy coding.**
- **Main Profile is targeted mainly towards the broadcast market. Supports both interlaced and progressive video with macroblock or picture level field/frame mode selection. Uses I, P, B slices, weighted prediction, both CAVLC and CABAC for entropy coding.**
- **Extended Profile is targeted towards error prone channels (such as mobile communication). Uses I, P, B, SP, SI slices, supports both interlaced and progressive video, allows CAVLC coding only.**

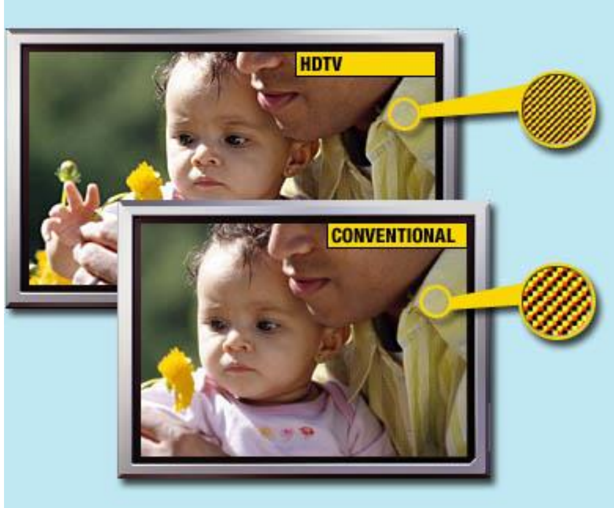
The Fidelity Range Extensions (FRExt) Profiles

- **High Profile** extends functionality of Main profile for more effective coding of high definition content; uses <u>adaptive 8×8 or 4×4 transform</u>, enables perceptual quantization matrices. This is the most efficient H.264/AVC profile.

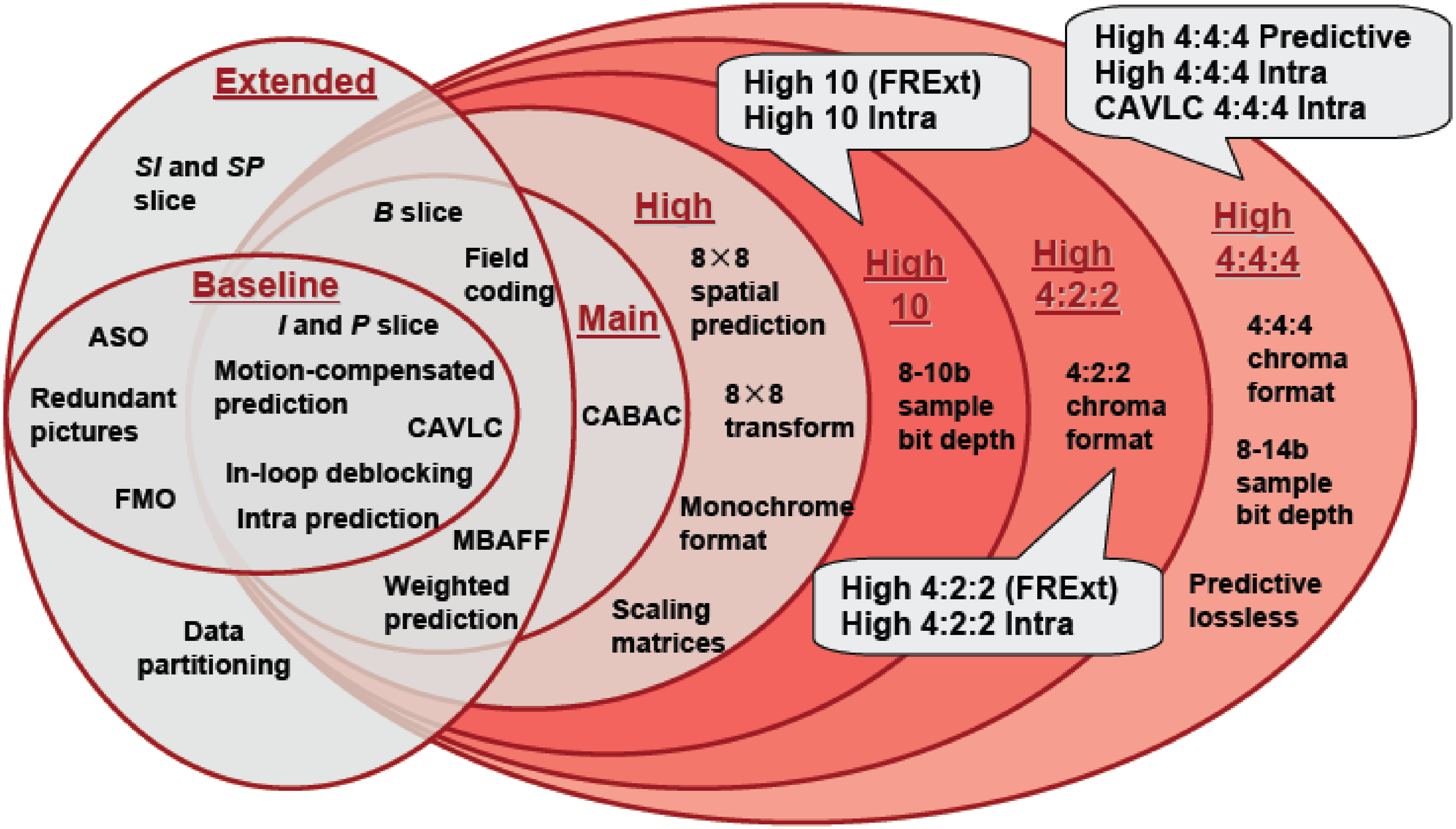
- **High 10 Profile** is an extension of High profile for 10 bit component resolution.

- **High 4:2:2 Profile** supports 4:2:2 chroma format and up to 10 bit component resolution; suitable for video production and editing.

- **High 4:4:4 Profile** supports 4:4:4 chroma format and up to 12 bit component resolution; in addition, it enables lossless mode of operation and direct coding of RGB signal; it targets professional production and graphics.



H.264/AVC Profiles



H.264/AVC Profiles: Technical Summary

Feature support in particular profiles

Feature	CBP	BP	XP	MP	HiP	Hi10P	Hi422P	Hi444PP
Chroma formats	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0/4:2:2	4:2:0/4:2:2/4:4:4
Sample depths (bits)	8	8	8	8	8	8 to 10	8 to 10	8 to 14
Flexible macroblock ordering (FMO)	No	Yes	Yes	No	No	No	No	No
Arbitrary slice ordering (ASO)	No	Yes	Yes	No	No	No	No	No
Redundant slices (RS)	No	Yes	Yes	No	No	No	No	No
Data Partitioning	No	No	Yes	No	No	No	No	No
SI and SP slices	No	No	Yes	No	No	No	No	No
B slices	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Interlaced coding (PicAFF, MBAFF)	No	No	Yes	Yes	Yes	Yes	Yes	Yes
CABAC entropy coding	No	No	No	Yes	Yes	Yes	Yes	Yes
8×8 vs. 4×4 transform adaptivity	No	No	No	No	Yes	Yes	Yes	Yes
Quantization scaling matrices	No	No	No	No	Yes	Yes	Yes	Yes
Separate C_b and C_r QP control	No	No	No	No	Yes	Yes	Yes	Yes
Monochrome (4:0:0)	No	No	No	No	Yes	Yes	Yes	Yes
Separate color plane coding	No	No	No	No	No	No	No	Yes
Predictive lossless coding	No	No	No	No	No	No	No	Yes

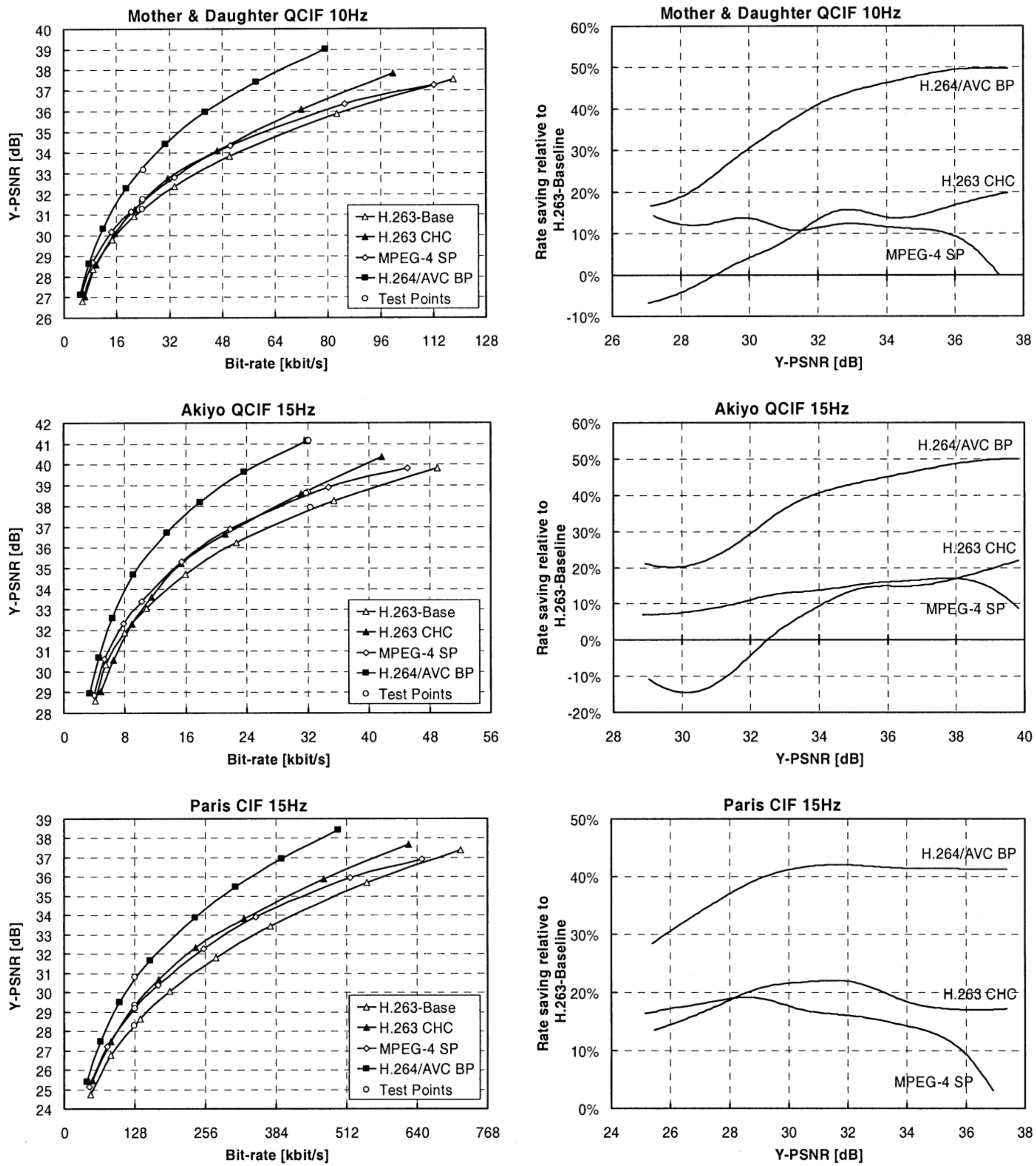
H.264/AVC Levels

Level Number	Max Macroblock processing Rate (Macroblock /second)	Max frame size (Macroblocks)	Max video bit rate (Bit/second) kbit/s	Max Coded Picture Buffer size (Bit)	Vertical Motion Vector Component Range (luma frame samples)	Min Compression Ratio	Max Number of Motion Vectors per Two Consecutive Macroblocks
10	1485	99	64	175	[-64,+63.75]	2	-
11	3000	396	192	500	[-128,+127.75]	2	-
12	6000	396	384	1000	[-128,+127.75]	2	-
13	11880	396	768	2000	[-128,+127.75]	2	-
20	11800	396	2000	2000	[-128,+127.75]	2	-
21	19800	792	4000	4000	[-256,+255.75]	2	-
22	20250	1620	4000	4000	[-256,+255.75]	2	-
30	40500	1620	10000	10000	[-256,+255.75]	2	32
31	108000	3600	14000	14000	[-512,+511.75]	4	16
32	216000	5120	20000	20000	[-512,+511.75]	4	16
40	245760	8192	20000	25000	[-512,+511.75]	4	16
41	245760	8192	50000	62500	[-512,+511.75]	2	16
42	491520	8192	50000	62500	[-512,+511.75]	2	16
50	589824	22080	135000	135000	[-512,+511.75]	2	16
51	983040	36864	240000	240000	[-512,+511.75]	2	16

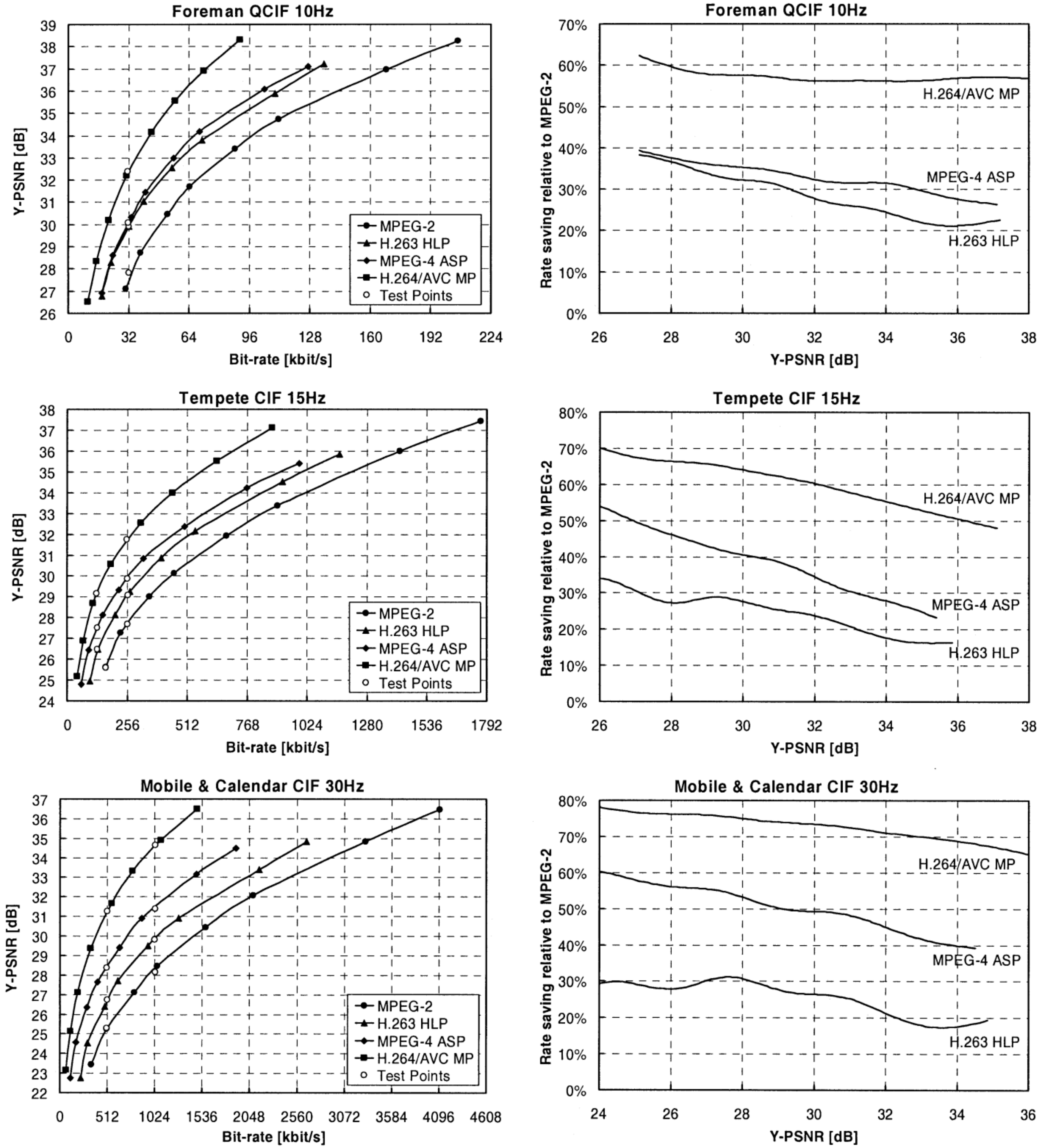
Some information on H.264/AVC's levels common for all profiles.

Performance

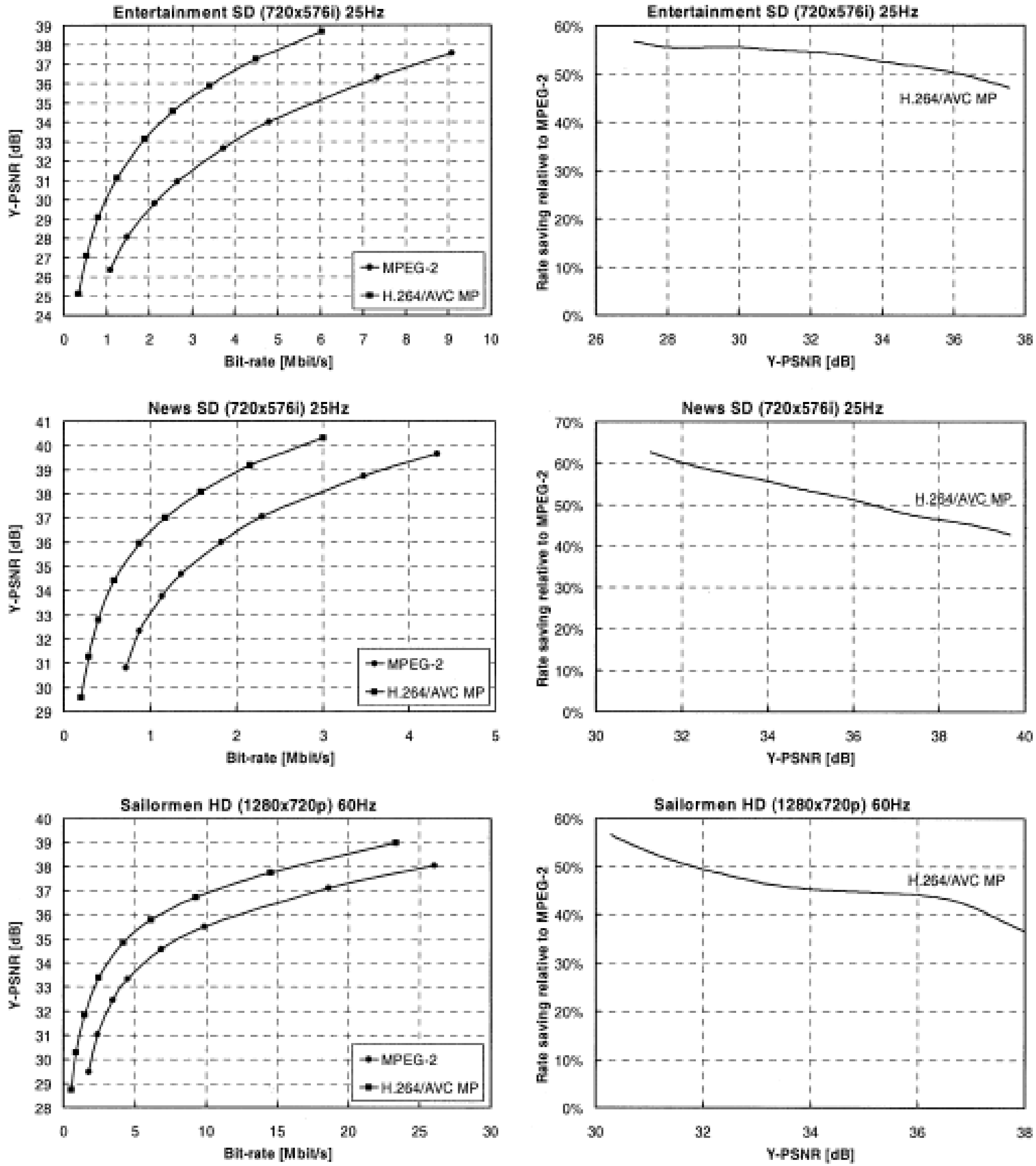
Performance for Videoconferencing Applications



Performance for Video Streaming Applications



Performance for Entertainment-Quality Applications



H.264/AVC: a Success Story …

- 3GPP (recommended in rel 6)
- 3GPP2 (optional for streaming service)
- ARIB (Japan mobile segment broadcast)
- ATSC (preliminary adoption for robust-mode back-up channel)
- Blu-ray Disc Association (mandatory for Video BD-ROM players)
- DLNA (optional in first version)
- DMB (Korea - mandatory)
- DVB (specified in TS 102 005 and one of two in TS 101 154)
- DVD Forum (mandatory for HD DVD players)
- IETF AVT (RTP payload spec approved as RFC 3984)
- ISMA (mandatory specified in near-final rel 2.0)
- SCTE (under consideration)
- US DoD MISB (US government preferred codec up to 1080p)
- (and, of course, MPEG and the ITU-T)

H.264/AVC Patent Licensing

- **As with MPEG-2 Parts and MPEG-4 Part 2 among others, the vendors of H.264/AVC products and services are expected to pay patent licensing royalties for the patented technology that their products use.**

- **The primary source of licenses for patents applying to this standard is a private organization known as MPEG LA (which is not affiliated in any way with the MPEG standardization organization); MPEG LA also administers patent pools for MPEG-2 Part 1 Systems, MPEG-2 Part 2 Video, MPEG-4 Part 2 Video, and other technologies.**

MPEG LA®

Decoder-Encoder Royalties

- **Royalties to be paid by end product manufacturers for an encoder, a decoder or both ("unit") begin at US $0.20 per unit after the first 100,000 units each year. There are no royalties on the first 100,000 units each year. Above 5 million units per year, the royalty is US $0.10 per unit.**
- **The maximum royalty for these rights payable by an Enterprise (company and greater than 50% owned subsidiaries) is $3.5 million per year in 2005-2006, $4.25 million per year in 2007-08 and $5 million per year in 2009-10.**
- **In addition, in recognition of existing distribution channels, under certain circumstances an Enterprise selling decoders or encoders both (i) as end products under its own brand name to end users for use in personal computers and (ii) for incorporation under its brand name into personal computers sold to end users by other licensees, also may pay royalties on behalf of the other licensees for the decoder and encoder products incorporated in (ii) limited to $10.5 million per year in 2005-2006, $11 million per year in 2007-2008 and $11.5 million per year in 2009-2010.**
- **The initial term of the license is through December 31, 2010. To encourage early market adoption and start-up, the License will provide a grace period in which no royalties will be payable on decoders and encoders sold before January 1, 2005.**

Participation Fees (1)

- **TITLE-BY-TITLE – For AVC video (either on physical media or ordered and paid for on title-by-title basis, e.g., PPV, VOD, or digital download, where viewer determines titles to be viewed or number of viewable titles are otherwise limited), there are no royalties up to 12 minutes in length. For AVC video greater than 12 minutes in length, royalties are the lower of (a) 2% of the price paid to the licensee from licensee's first arms length sale or (b) $0.02 per title. Categories of licensees include (i) replicators of physical media, and (ii) service/content providers (e.g., cable, satellite, video DSL, internet and mobile) of VOD, PPV and electronic downloads to end users.**

- **SUBSCRIPTION – For AVC video provided on a subscription basis (not ordered title-by-title), no royalties are payable by a system (satellite, internet, local mobile or local cable franchise) consisting of 100,000 or fewer subscribers in a year. For systems with greater than 100,000 AVC video subscribers, the annual participation fee is $25,000 per year up to 250,000 subscribers, $50,000 per year for greater than 250,000 AVC video subscribers up to 500,000 subscribers, $75,000 per year for greater than 500,000 AVC video subscribers up to 1,000,000 subscribers, and $100,000 per year for greater than 1,000,000 AVC video subscribers.**

TÉCNICO LISBOA

Participation Fees (2)

- **Over-the-air free broadcast – There are no royalties for over-the-air free broadcast AVC video to markets of 100,000 or fewer households. For over-the-air free broadcast AVC video to markets of greater than 100,000 households, royalties are $10,000 per year per local market service (by a transmitter or transmitter simultaneously with repeaters, e.g., multiple transmitters serving one station).**

- **Internet broadcast (non-subscription, not title-by-title) – Since this market is still developing, no royalties will be payable for internet broadcast services (non-subscription, not title-by-title) during the initial term of the license (which runs through December 31, 2010) and then shall not exceed the over-the-air free broadcast TV encoding fee during the renewal term.**

- **The maximum royalty for Participation rights payable by an Enterprise (company and greater than 50% owned subsidiaries) is $3.5 million per year in 2006-2007, $4.25 million in 2008-09 and $5 million in 2010.**

- **As noted above, the initial term of the license is through December 31, 2010. To encourage early marketplace adoption and start-up, the License will provide for a grace period in which no Participation Fees will be payable for products or services sold before January 1, 2006.**

The H.264/AVC Family …

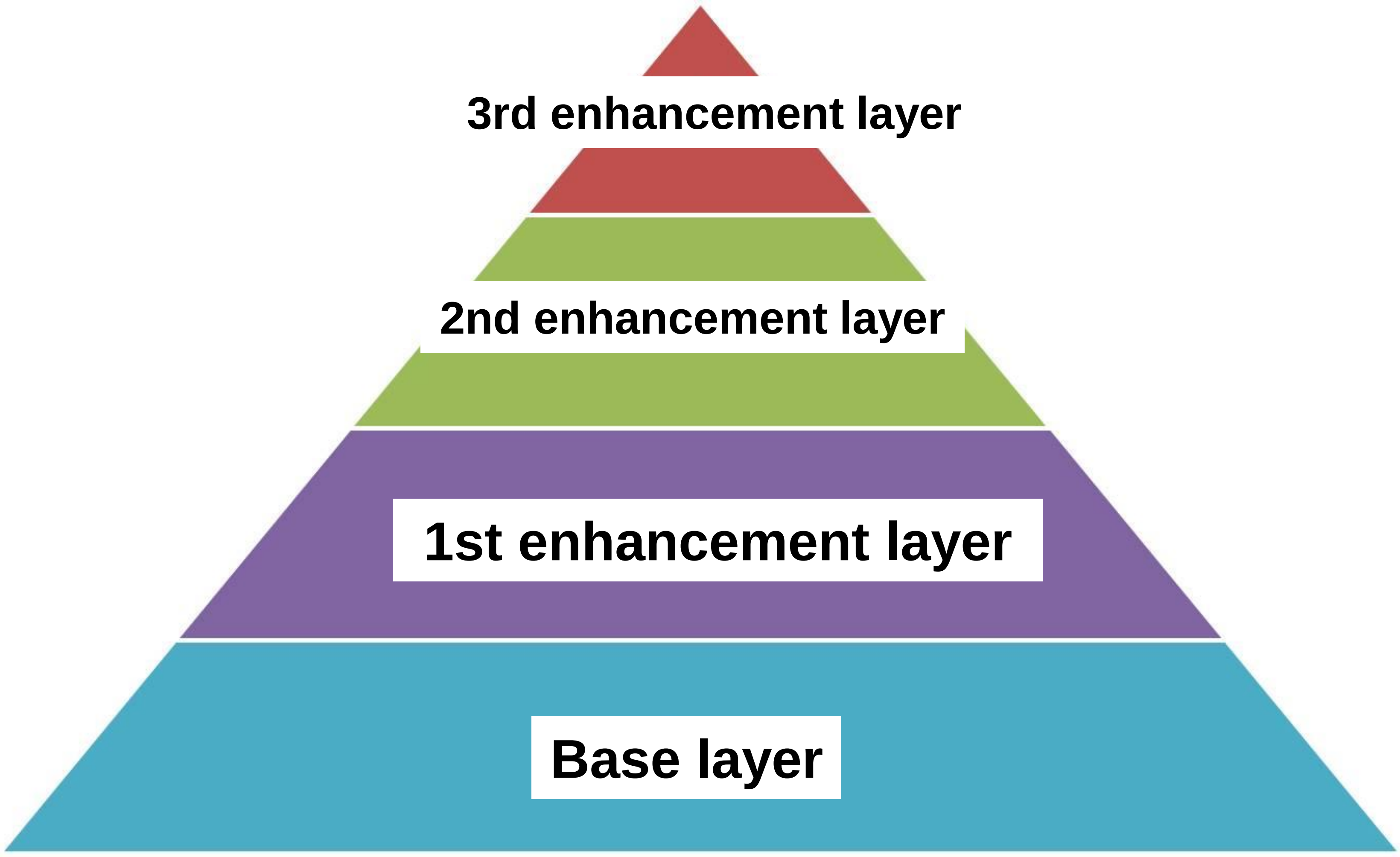
AVC, Advanced Video Coding

MVC, Multiview Video Coding

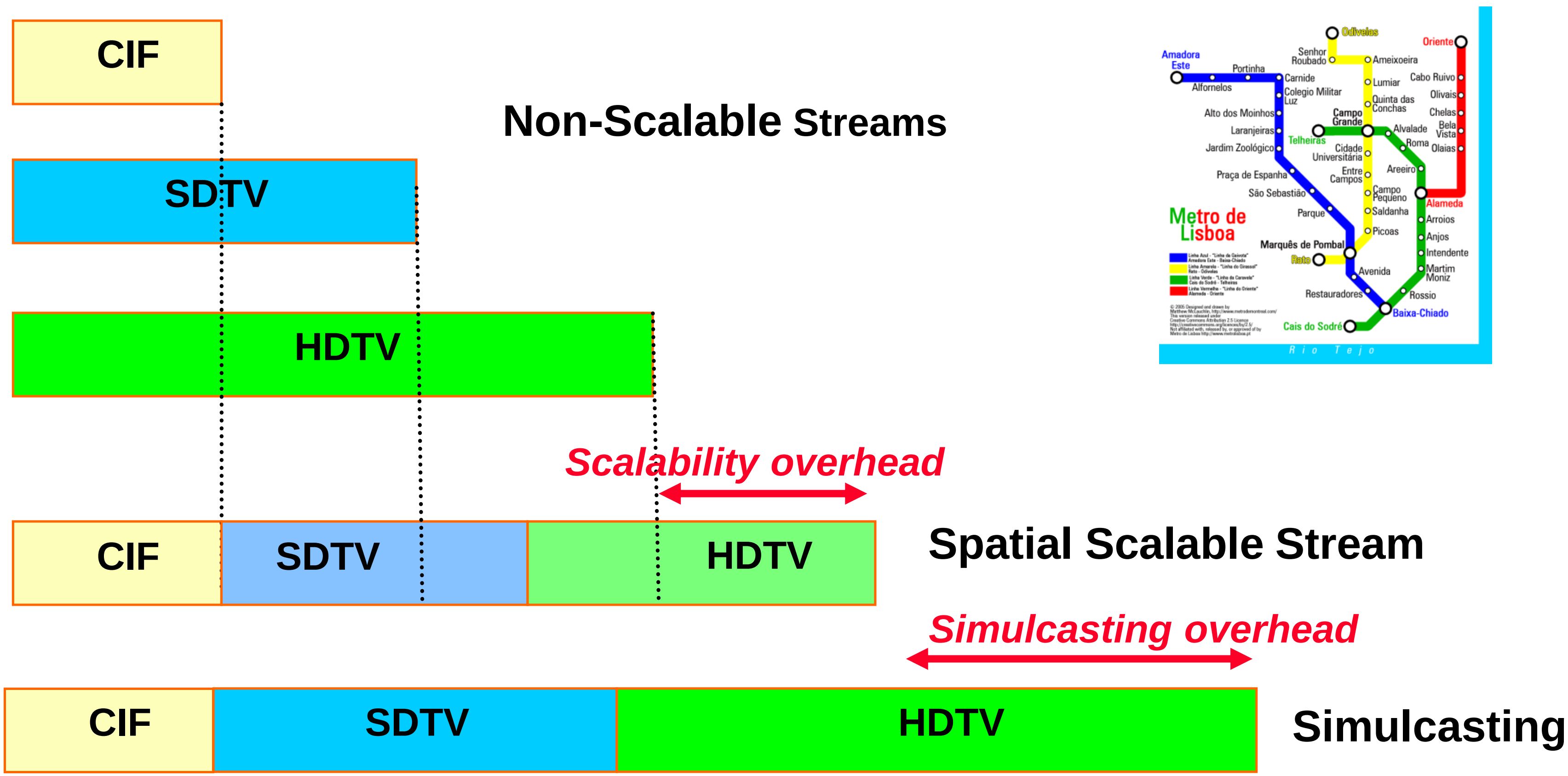
SVC, Scalable Video Coding

An Heterogeneous World …

Scalable Hierarchical Coding

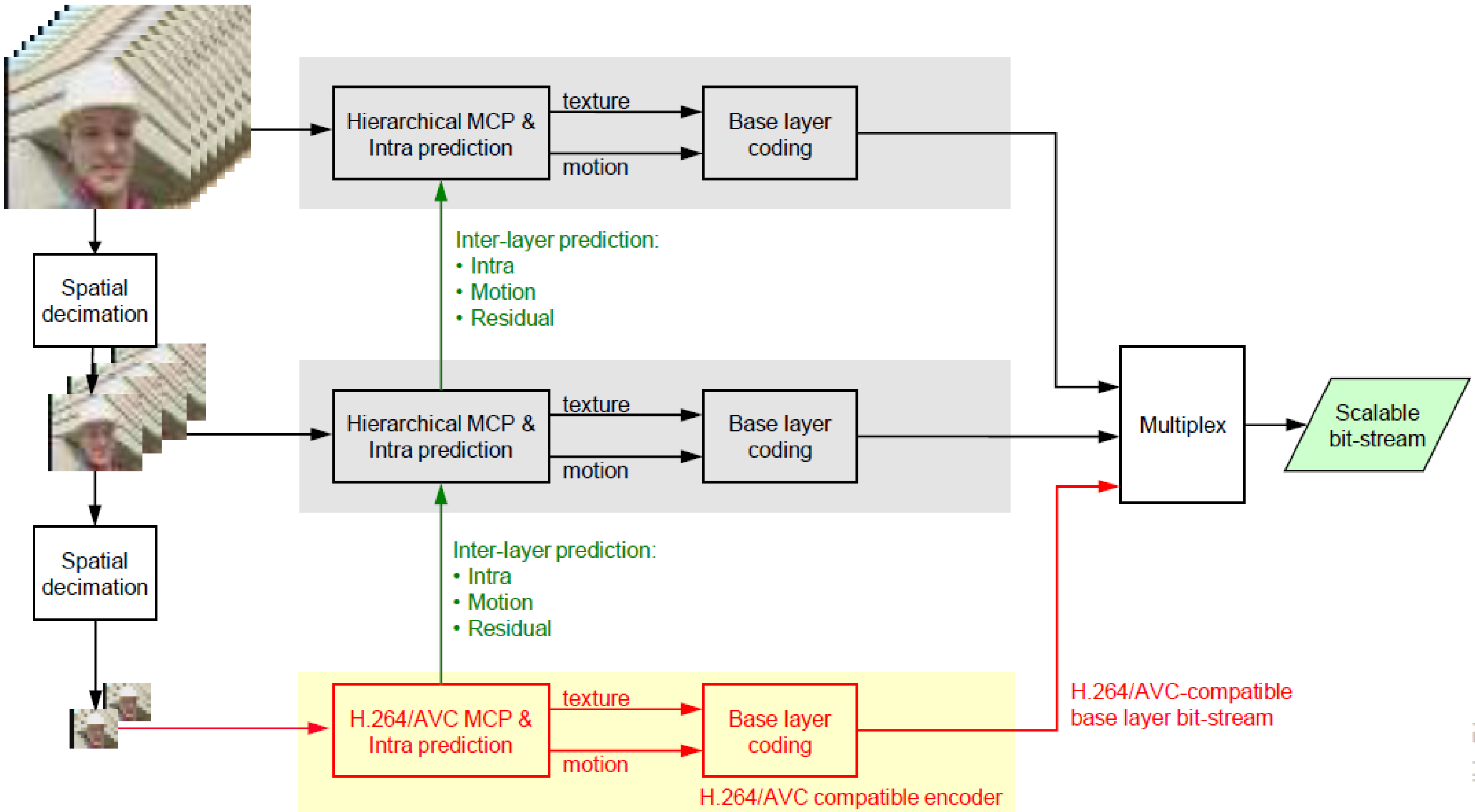


Scalability: Rate Benefits and Penalties



For each spatial resolution (except the lowest), the scalable stream asks for a bitrate overhead regarding the corresponding alternative non-scalable stream, although the total bitrate is lower than the total simulcasting bitrate.

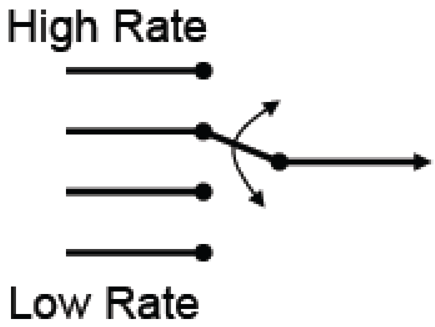
SVC Encoder: Compressed Domain Spatial Scalability



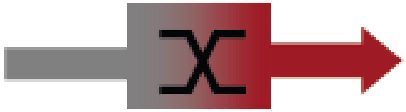
SVC Alternatives for Heterogeneous Environments

- **Simulcast**
  - Simplest solution
  - Each layer as an independent stream
  - Incurs significant increase of rate

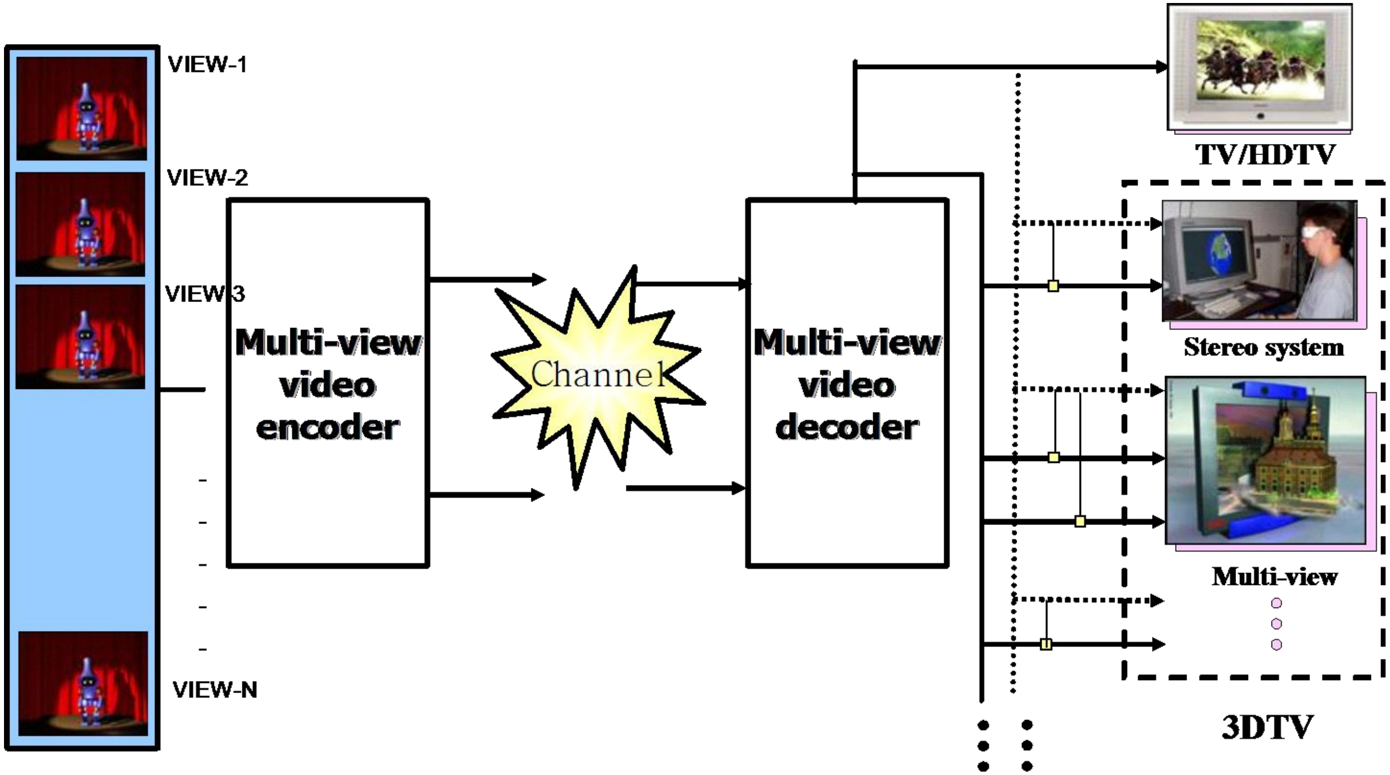
- **Stream Switching**
  - Viable for some application scenarios
  - Lacks flexibility within the network
  - Incurs increase of server storage/complexity



- **Transcoding** *(direct digital-to-digital conversion of one encoding to another, e.g. with different rate, resolution, format)*
  - Low cost, designed for specific application needs
  - Already deployed in many application domains
  - Incurs increase of computation



Multiview Video Coding



Multiview video (MVV) refers to a set of N temporally synchronized video streams coming from cameras capturing the same real scenery from different viewpoints.

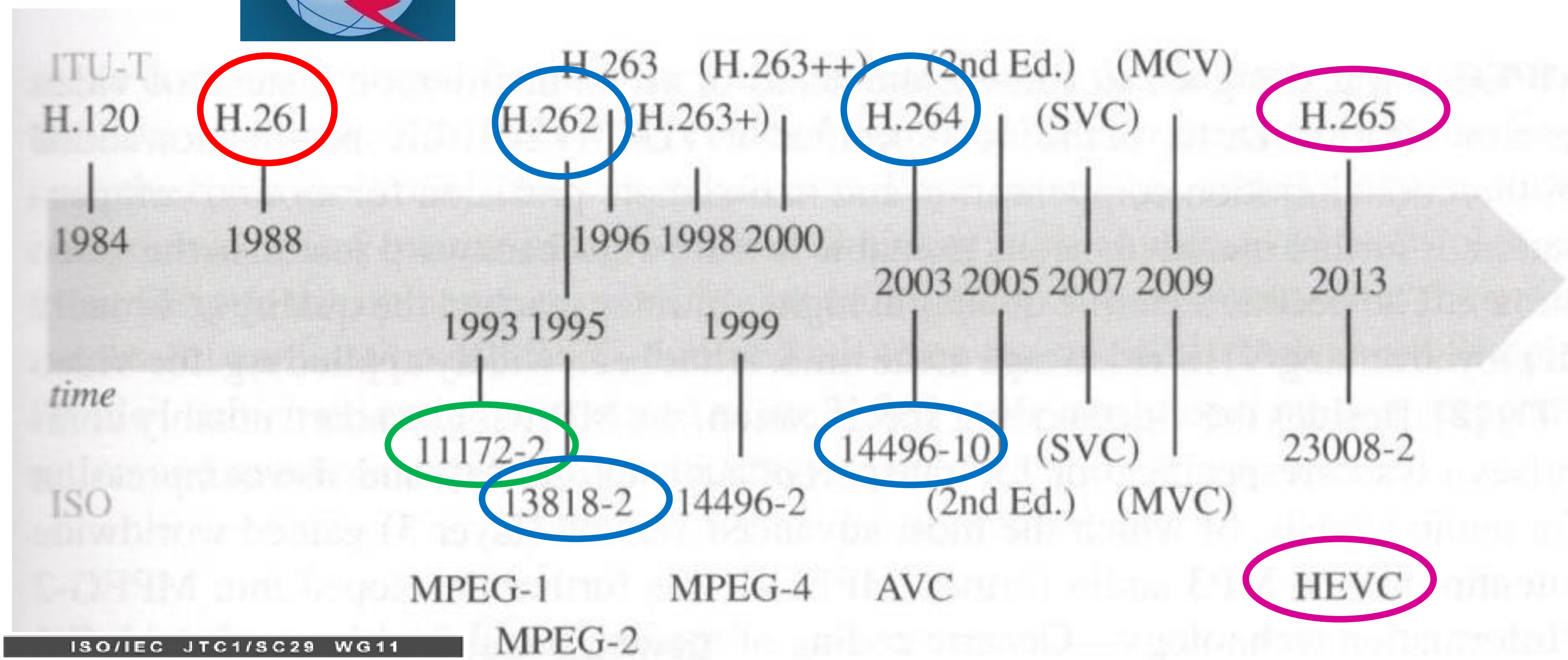
TÉCNICO LISBOA

KEEP CALM

BECAUSE

THIS IS NOT THE END

Standards Over Time …



High Efficiency Video Coding (HEVC) Standard: Why ?

~2014

- **Video is continuously increasing in resolution and views**
  - HD existing, Ultra HD (4K×2K, 8K×4K) appearing
  - Mobile services going towards HD
  - Stereo, multi-view emerging
- **Devices available to record and display Ultra HD resolutions**
  - Becoming affordable for home and mobile consumers
- **Video has multiple dimensions to grow the data rate**
  - Spatial resolution, temporal resolution
  - Color resolution, bit depth
  - Multi-view
- **Necessary video data rate grows faster than feasible network transport capacities**
  - Better video compression (than current H.264/AVC) needed in next decade

TÉCNICO LISBOA

Sensors and Displays Leading the Process

Digital Television Format Comparison

	RESOLUTION	ASPECT RATIO	FRAME RATE*
UHDTV	7680 x 4320	16:9	60p
HDTV	1920 x 1080	16:9	24p, 30p, 60i
	1280 x 720	16:9	24p, 30p, 60i
SDTV	704 x 480	16:9	24p, 30p, 60i, 60p
	704 x 480	4:3	24p, 30p, 60i, 60p
	640 x 480	4:3	24p, 30p, 60i, 60p

Main HEVC Requirements

50% gain

- **Compression - Substantially greater bitrate reduction over the H.264/AVC High profile** is required for the target application(s); at no point of the entire bitrate range shall HEVC be worse than existing standard(s). Subjective visually lossless compression shall be supported.

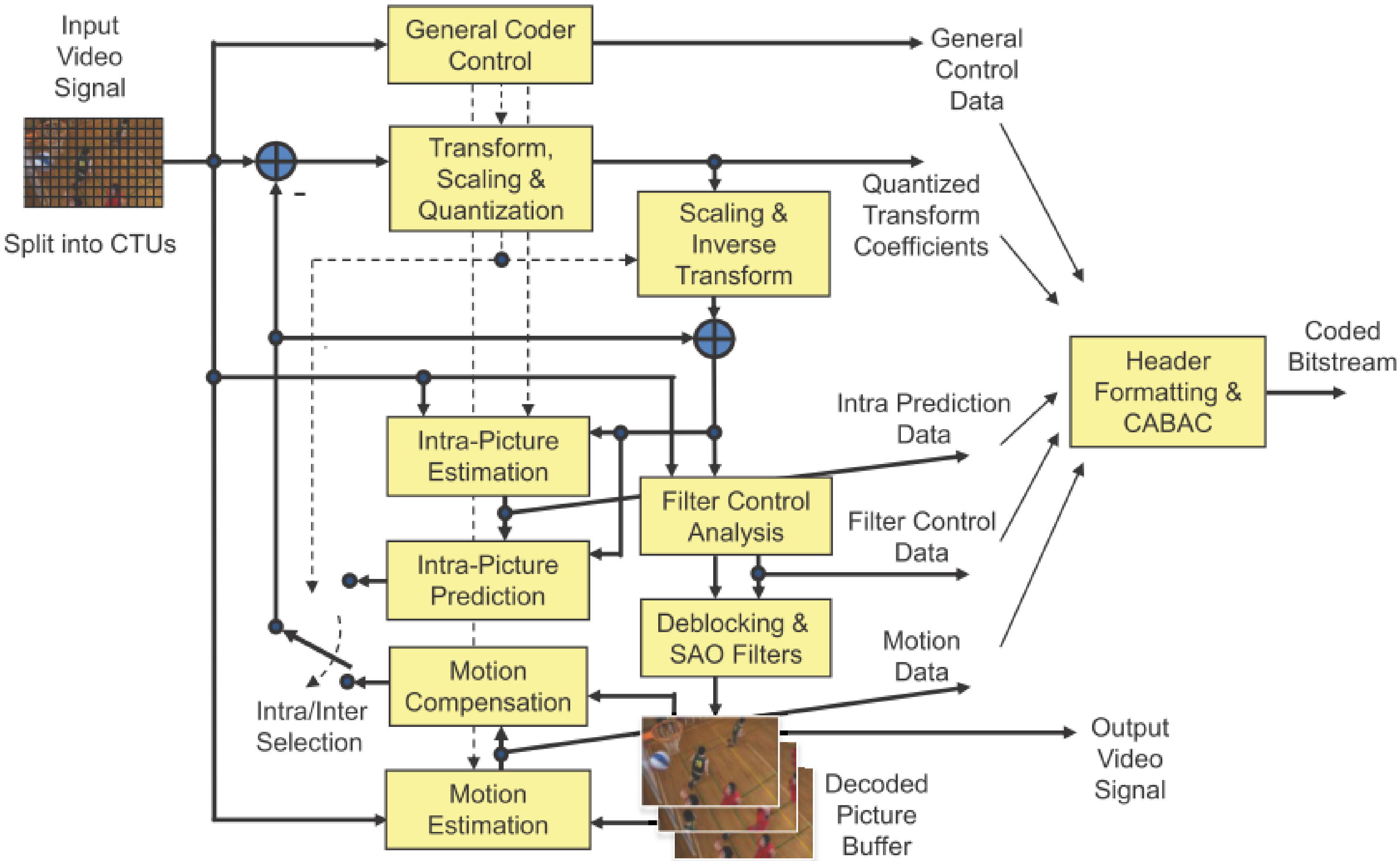
- **Complexity -** Shall allow for feasible implementation within the constraints of the available technology at the expected time of usage. **HEVC should be capable of trading-off complexity and compression efficiency** by having: i) an operating point with significant decrease in complexity compared to H.264/AVC but with better compression efficiency than H.264/AVC; ii) an operating point with increase complexity and commensurate increase in compression performance.

- **Picture Formats -** Focus on a set of rectangular picture formats that will include all commonly used picture formats, ranging at least from VGA to 4K×2K, and potentially extending to QVGA and **8K×4K**.

- **Color Spaces and Color Sampling -** a) The YCbCr color space 4:2:0, 8 bits per component shall be supported; b) YCbCr/RGB **4:4:4** should be supported; c) Higher bit depth **up to 14 bits** per component should be supported.

TÉCNICO LISBOA

HEVC: Still a Quiet Evolution …



Bibliography

- I. Richardson, **H.264 and MPEG-4 Video Compression**, John Wiley & Sons, 2003
- T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, **Overview of the H.264/AVC Video Coding Standard**, IEEE Transactions on Circuits and Systems for Video Technology, July 2003
- J. Ostermann, et al., **Video coding with H.264/AVC: Tools, Performance, and Complexity**, IEEE Circuits and Systems Magazine, 2004
- T. Wiegand, et al., **Rate-Constrained Coder Control and Comparison of Video Coding Standards**, IEEE Transactions on Circuits and Systems for Video Technology, July 2003
- G. J. Sullivan, P. Topiwala, and A. Luthra, **The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions**, SPIE Conference on Applications of Digital Image Processing XXVII, August 2004